

IBM WebSphere Commerce



# 관리 안내서

버전 5.5



IBM WebSphere Commerce



# 관리 안내서

버전 5.5

**주!**

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 반드시 199 페이지의 『주의사항』의 정보를 읽으십시오.

**초판(2003년 7월)**

이 개정판은 새 개정판에서 별도로 명시하지 않는 한 IBM WebSphere Commerce 버전 5.5(제품 번호 5724-A18) 및 모든 후속 릴리스와 수정에 적용됩니다. 제품 레벨에 올바른 개정판을 사용하고 있는지 확인하십시오.

책에 대한 주문은 한국 IBM 담당자 또는 해당 지역의 IBM 지방 사무소로 문의하십시오.

IBM은 귀하의 의견을 환영합니다. 다음 URL에서 사용 가능한 온라인 IBM WebSphere Commerce 문서 피드백 양식을 사용하여 의견을 보낼 수 있습니다.

<http://www.ibm.com/software/webservers/commerce/rcf.html>

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

© Copyright International Business Machines Corporation 2003. All rights reserved.

# 목차

이 책에 대하여 . . . . .	vii	LDAP -- Lightweight Directory Access Protocol . . . . .	31
정보 위치 . . . . .	vii	스태이징 서버 . . . . .	39
WebSphere Commerce 서적 . . . . .	vii	연결 서비스 . . . . .	66
WebSphere Commerce 온라인 도움말 . . . . .	viii	구성 태스크 . . . . .	69
웹의 추가 정보 . . . . .	viii	메시지 전달 서비스 구성 . . . . .	69
+ 변경사항 요약 . . . . .	viii	스케줄 서비스 구성 . . . . .	70
시작하기 전에 . . . . .	ix	로그 작성 구성 . . . . .	70
정보 요구사항 . . . . .	x	로그 파일 보기 . . . . .	73
이 책에 사용된 규칙 . . . . .	x	LDAP 구성 . . . . .	75
경로 변수 . . . . .	xi	규칙 서비스 구성 . . . . .	75
<b>제 1 장 WebSphere Commerce 관리 소개 . . . . .</b>	<b>1</b>	스태이징 서버 구성 . . . . .	75
관리 도구 . . . . .	2	연결 구성 . . . . .	82
관리 콘솔 . . . . .	2	WebSphere Commerce Payments 구성 . . . . .	82
구성 관리자 . . . . .	3	<b>제 4 장 호스팅된 상점 관리 . . . . .</b>	<b>85</b>
WebSphere Commerce Payments . . . . .	3	재판매자 관리 . . . . .	85
조직 관리 콘솔 . . . . .	11	배급업자 관리 . . . . .	86
WebSphere Commerce 액셀러레이터 . . . . .	11	사이트 관리 . . . . .	86
WebSphere Application Server . . . . .	12	<b>제 5 장 동적 캐시 작성 . . . . .</b>	<b>87</b>
데이터베이스 . . . . .	12	WebSphere Commerce Servlet 또는 JSP 결과 캐시 작성 . . . . .	87
<b>제 2 장 기본 관리 태스크 . . . . .</b>	<b>13</b>	WebSphere Application Server 동적 캐시 . . . . .	87
1회 수행 관리 태스크 . . . . .	13	동적 캐시 . . . . .	88
설치 및 구성 . . . . .	13	동적 캐시 서비스 및 Servlet 캐시 작성 사용 . . . . .	88
정기적인 관리 태스크 . . . . .	13	동적 캐시 모니터 사용 . . . . .	88
WebSphere Commerce의 시작 및 중지 . . . . .	13	WebSphere 동적 캐시 모니터에 대한 종속성 ID의 트리 보기 . . . . .	89
WebSphere Commerce 구성 관리자 실행 . . . . .	14	캐시 가능 오브젝트 구성 . . . . .	89
인스턴스 작성, 구성 및 삭제 . . . . .	17	캐시 작성 기본값 . . . . .	91
관리 콘솔 열기 . . . . .	19	전체 페이지 및 단편 캐시 작성 . . . . .	91
조직 관리 콘솔 열기 . . . . .	20	전체 페이지 캐시 작성 . . . . .	92
WebSphere Commerce 구성요소의 사용 및 사용 안함 . . . . .	21	단편 캐시 . . . . .	93
WebSphere Commerce Payments 시작 및 중지 . . . . .	22	캐시 Servlet 필터링 . . . . .	93
설치 안내서의 관리 태스크 . . . . .	24	상점 페이지 캐시 전략 . . . . .	95
<b>제 3 장 구성 . . . . .</b>	<b>25</b>	WebSphere Commerce의 캐시 무효화 메커니즘 . . . . .	96
구성 서비스 . . . . .	25	cachespec.xml에 무효화 정책 정의 . . . . .	96
구조적 구성 . . . . .	25	Cache Monitor를 통한 캐시 항목 제거 . . . . .	99
메시지 전달 서비스 . . . . .	25	동적 캐시 무효화 API 호출 . . . . .	100
스케줄 서비스 . . . . .	26	명령 기반 무효화를 위한 명령 쓰기 . . . . .	100
로그 작성 서비스 . . . . .	26	WebSphere Commerce에서 명령 무효화 사용의 예 . . . . .	100
WebSphere Commerce Payments 문제점 판별 . . . . .	30		
규칙 서비스 . . . . .	30		

캐시 무효화 예. . . . .	101
캐시 무효화 API(DynaCacheInvalidation) . . .	105
CACHEIVL 테이블 . . . . .	105
외부 캐시에서 무효화 . . . . .	105
웹 서버 플러그인 캐시 . . . . .	105
단순 파일 Servlet. . . . .	106
문제점 판별. . . . .	108
<b>제 6 장 시스템 유지보수 . . . . .</b>	<b>111</b>
데이터베이스 정리 유틸리티 . . . . .	111
데이터베이스 정리 유틸리티 오브젝트 . . . . .	112
데이터베이스 정리 명령 . . . . .	118
데이터베이스 정리. . . . .	122
WebSphere Commerce 유틸리티에 대한 PATH	
환경 변수 설정. . . . .	123
오브젝트 삭제 . . . . .	124
계정 오브젝트 삭제 . . . . .	124
주소 오브젝트 삭제 . . . . .	125
ATP 재고(Available to promise inventory) 오	
브젝트 삭제. . . . .	125
첨부 오브젝트 삭제 . . . . .	126
경매 오브젝트 삭제 . . . . .	127
경매 로그 오브젝트 삭제 . . . . .	127
autobid 로그 오브젝트 삭제 . . . . .	128
기본 항목 오브젝트 삭제 . . . . .	129
입찰 로그 오브젝트 삭제 . . . . .	129
계산 코드 오브젝트 삭제 . . . . .	130
캠페인 오브젝트 삭제 . . . . .	130
캠페인 통계 오브젝트 삭제. . . . .	131
카탈로그 항목 오브젝트 삭제 . . . . .	132
카탈로그 그룹 오브젝트 삭제 . . . . .	132
장기 구매 계약 오브젝트 삭제. . . . .	133
쿠폰 오브젝트 삭제 . . . . .	134
예상 재고 레코드 오브젝트 삭제 . . . . .	134
예상 재고 레코드 오브젝트 정보 삭제 . . . . .	135
포럼 메시지 오브젝트 삭제. . . . .	136
서비스 센터 오브젝트 삭제. . . . .	136
재고 코드 조정 오브젝트 삭제. . . . .	137
재고 조정 오브젝트 삭제 . . . . .	138
지정된 항목 정보 오브젝트 삭제 . . . . .	138
구성원 메시지 관계 오브젝트 삭제 . . . . .	139
메시지 오브젝트 삭제 . . . . .	140
주문 오브젝트 삭제 . . . . .	140
조직 오브젝트 삭제 . . . . .	141
상품 어드바이저 통계 오브젝트 삭제 . . . . .	142
상품 비교 통계 오브젝트 삭제. . . . .	142
상품 탐색기 통계 오브젝트 삭제 . . . . .	143

정책 오브젝트 삭제 . . . . .	144
상품 세트 오브젝트 삭제 . . . . .	144
RFQ 오브젝트 삭제 . . . . .	145
반품 항목 오브젝트 삭제 . . . . .	145
반품 이유 오브젝트 삭제 . . . . .	146
판매 보조 기능 통계 오브젝트 삭제. . . . .	147
스테이징된 오브젝트 삭제 . . . . .	148
상점 오브젝트 삭제 . . . . .	148
사용자 오브젝트 삭제 . . . . .	149
사용자 트래픽 로그 오브젝트 삭제 . . . . .	150
공급업체 오브젝트 삭제 . . . . .	150
데이터베이스 정리 유틸리티에 새 구성 추가하는 법	151
오브젝트 삭제 예 . . . . .	152
<b>제 7 장 성능 . . . . .</b>	<b>155</b>
WebSphere Commerce PMI 모듈을 사용한 성능	
모니터링 . . . . .	156
PMI 모듈 설정. . . . .	157
Tivoli Performance Viewer 시작 . . . . .	158
WebSphere Commerce PMI 모듈 보고서 값	159
기타 성능 분석 도구. . . . .	160
스케줄러 유지보수. . . . .	160
LDAP에 대한 복제 . . . . .	161
스테이징 서버에서 사이트 테스트. . . . .	161
스테이징 서버 구성 . . . . .	162
사용자 정의 테이블의 트리거 작성 . . . . .	162
원격 데이터베이스 구성 . . . . .	162
스테이징 데이터베이스로 데이터 복사 . . . . .	162
Stage Check 명령 실행. . . . .	163
프로덕션 데이터베이스로 데이터 전달 . . . . .	164
프로덕션 서버로 파일 복사. . . . .	165
스테이징된 오브젝트 삭제 . . . . .	165
WebSphere Commerce Payments 성능 조정 매개	
변수 . . . . .	166
iSeries에서 Payments 인스턴스 데이터베이스 관리	168
<b>제 8 장 문제점 해결. . . . .</b>	<b>169</b>
Adapter for CrossWorlds® . . . . .	169
WebSphere Commerce Payments . . . . .	170
고성능 환경을 위한 성능 조정. . . . .	171
<b>부록 A. LDAP 시나리오 . . . . .</b>	<b>173</b>
LDAP 시나리오: LDAP 서버를 구성원 저장소로	
사용 . . . . .	173
<b>부록 B. LDAP 파일. . . . .</b>	<b>175</b>
ldapmap.dtd . . . . .	175
ldapentry.xml . . . . .	176

<b>부록 C. WebSphere Commerce Payments 학습</b>	
서 . . . . .	181
1단계: WebSphere Commerce Payments 사용자 인터페이스 액세스. . . . .	181
2단계: WebSphere Commerce Payments 판매자 작성 및 카세트 권한 부여 . . . . .	182
3단계: WebSphere Commerce Payments 사용자 정의 . . . . .	183
4단계: 사용자 역할 지정 . . . . .	184
판매자 관리자로서 로그인 . . . . .	185
5단계: 계정 작성 . . . . .	185
계정 설정 지정. . . . .	186
6단계: 브랜드 작성 . . . . .	187
7단계: Sample Checkout을 사용하여 주문 작성	188

8단계: 주문 승인 . . . . .	189
주문 페이지에서 주문 승인. . . . .	189
판매 가능으로 주문 승인 . . . . .	190
9단계: 지불 예치 . . . . .	191
10단계: 결산 일괄처리 . . . . .	191
11단계: 대변 발행. . . . .	193
12단계: 일괄처리 총계 보기 . . . . .	194

<b>부록 D. WebSphere Commerce Payments에 대</b>	
<b>한 자국어 지원(NLS) 정보.</b> . . . . .	197
NLS 힌트와 추가정보 . . . . .	197
지원되는 언어 및 로케일 . . . . .	198
주의사항 . . . . .	199





---

## 이 책에 대하여

이 책은 관리자를 위한 것으로 관리 콘솔 및 기타 관리 도구를 사용하여 WebSphere® Commerce 구성요소를 관리하는 방법에 대해 설명합니다.

---


## 정보 위치


WebSphere Commerce는 전자상거래 솔루션에 대해 설명하는 온라인 및 하드카피 정보를 제공합니다. WebSphere Commerce에서 제공되는 소프트웨어 제품은 소프트웨어의 특정 기능에 대해 설명하는 추가 정보를 제공합니다. 이 절에서는 여러 유형의 정보를 찾을 수 있는 위치에 대한 빠른 개요를 제공합니다.

### WebSphere Commerce 서적

- *WebSphere Commerce 기본 정보*, 버전 5.5
- *WebSphere Commerce 프로그래밍 안내서 및 학습서*, 버전 5.5
- *WebSphere Commerce 설치 안내서*, 버전 5.5
- *WebSphere Commerce 빠른 시작*, 버전 5.5
- *WebSphere Commerce 추가 소프트웨어 안내서*, 버전 5.5
- *WebSphere Commerce 이주 안내서*, 버전 5.5
- *WebSphere Commerce 상점 개발 안내서*, 버전 5.5
- *WebSphere Commerce 견본 상점 안내서*, 버전 5.5
- *WebSphere Commerce 관리 안내서*, 버전 5.5
- *WebSphere Commerce 보안 안내서*, 버전 5.5
- *WebSphere Commerce Payments 프로그래밍 안내서 및 참조서*, 버전 5.5
- *WebSphere Commerce Payments OfflineCard Cassette 보충 설명서*, 버전 5.5
- *WebSphere Commerce Payments CustomOffline Cassette 보충 설명서*, 버전 5.5

이러한 서적의 갱신사항은 다음 웹 주소를 참조하십시오.

 Business Edition([http://www.ibm.com/software/webservers/commerce/wc\\_be/](http://www.ibm.com/software/webservers/commerce/wc_be/))

 Professional Edition([http://www.ibm.com/software/webservers/commerce/wc\\_pe/](http://www.ibm.com/software/webservers/commerce/wc_pe/))

## WebSphere Commerce 온라인 도움말

WebSphere Commerce 온라인 도움말은 웹 브라우저를 사용하여 볼 수 있는 온라인 정보로 구성됩니다.

온라인 도움말은 Internet Explorer, 버전 5.5 이상에서 실행되는 웹 브라우저에서 다음 주소를 사용하여 액세스할 수 있습니다.

`https://host_name:8000/wchelp`

여기서 `host_name`은 WebSphere Commerce 시스템의 이름입니다.

## 웹의 추가 정보

### WebSphere Commerce 지원

뉴스그룹, FAQ, 기술 정보, 문제점 해결 정보 및 다운로드를 포함한 지원 정보를 찾으려면 다음 웹 주소를 방문하십시오.

WebSphere Commerce 지원 웹 페이지(<http://www.ibm.com/software/commerce/support/>)

WebSphere Commerce 기술 라이브러리 웹 페이지(<http://www.ibm.com/software/commerce/library/>)

### 소프트웨어 파트너

많은 소프트웨어 파트너가 WebSphere Commerce를 향상시키기 위한 제품 및 서비스를 제공합니다. 이러한 파트너에 대한 정보를 보려면 Commerce Zone (<http://www.ibm.com/software/wsdd/zones/commerce/>)에 방문하여 Software Developers 링크를 누르십시오.

### Redbooks™

추가 고급 기술 정보를 찾으려면 Redbooks 웹 사이트(<http://www.ibm.com/redbooks>)를 방문하여 WebSphere Commerce를 검색하십시오.

---

## + 변경사항 요약

- + 이 안내서 및 이 안내서의 갱신된 버전은 WebSphere Commerce 기술 라이브러리 웹
- + 페이지(<http://www.ibm.com/software/commerce/library/>)에서 사용 가능합니다.
- + WebSphere Commerce Edition에 대한 추가 정보는 개요 페이지를 참조하십시오.
- +
  - Business Edition
  - + ([http://www.ibm.com/software/webservers/commerce/wc\\_be/](http://www.ibm.com/software/webservers/commerce/wc_be/))
  - Professional Edition
  - + ([http://www.ibm.com/software/webservers/commerce/wc\\_pe/](http://www.ibm.com/software/webservers/commerce/wc_pe/))

+ 추가 지원 정보는 WebSphere Commerce 지원 사이트(<http://www.ibm.com/software/commerce/support/>)를 참조하십시오.

+ 제품의 최종 변경사항에 대해 알려면, 갱신된 제품에 포함된 README 파일(또한 위  
+ 의 웹 사이트에서 사용할 수 있는)을 참조하십시오. WebSphere Commerce 설치 방법  
+ 및 지원 제품에 대한 지시사항은 *WebSphere Commerce 설치 안내서*를 참조하십시오.

+ 이 책의 마지막 버전으로부터의 갱신사항은 여백에 표시되는 개정 문자로 구분됩니다.  
+ 이 책에서는 다음과 같은 규칙을 개정 문자에 사용합니다.

- + • "+"(플러스) 문자는 이 책의 현재 버전만 해당하는 갱신사항을 나타냅니다.

+ 다음 테이블은 이 책에 작성한 기본 변경사항을 보여줍니다.

변경	영향받는 장 또는 페이지
iSeries™ 플랫폼에 대한 수정된 dbclean 예제.	111 페이지의 제 6 장 『시스템 유지보수』
iSeries 플랫폼에 대한 수정된 dbclean과 stagingcopy, stagingcheck 및 stagingprop 예제.	155 페이지의 제 7 장 『성능』 및 39 페이지의 『스태징 서버』
"독립형 도구에 대한 로그 작성" 절은 더 이상 사용되지 않으므로 제거되었습니다.	70 페이지의 『로그 작성 구성』
동적 캐시에 관한 정보의 설명 및 기타 갱신사항.	89 페이지의 『캐시 가능 오브젝트 구성』

## 시작하기 전에

IBM® WebSphere Commerce 관리 안내서는 사이트 운영자 또는 시스템 관리자가 WebSphere Commerce 사이트에서 수행해야 하는 태스크에 대한 정보를 제공합니다. 이 안내서에서는 두 역할을 사이트 운영자 역할이라고 합니다.

### 중요

이 책에는 플랫폼별 정보가 나와 있습니다. 반드시 플랫폼에 해당하는 지시사항을 따르십시오.

이 책은 다음 장으로 구성됩니다.

### WebSphere Commerce 관리 소개

기본 관리 태스크

구성

호스팅된 상점 관리

동적 캐시 작성

시스템 유지보수

성능

문제점 해결

부록 A: LDAP 시나리오

부록 B: LDAP 파일

부록 C: WebSphere Commerce Payments 학습

부록 D: WebSphere Commerce Payments에 대한 자국어 지원(NLS) 정보

## 정보 요구사항

이 책은 WebSphere Commerce 사이트의 설치 및 유지보수하는 방법을 알아야 하는 사이트 운영자 또는 시스템 관리자를 위한 것입니다. 운영자는 다음 영역에 대한 지식이 있어야 합니다.

- 전자상거래 솔루션 아키텍처
- 방화벽, DFS™, DNS, 라우터
- 데이터베이스 기술
- XML
- 상점 비즈니스 프로시저

## 이 책에 사용된 규칙

이 책에서는 다음 규칙이 사용됩니다.

굵은체는 필드, 버튼 또는 메뉴 선택사항과 같은 GUI(Graphical User Interface) 제어를 나타냅니다.

모노체는 디렉토리 경로 및 표시된 것과 똑같이 입력하는 텍스트의 예를 나타냅니다.

기울임꼴은 사용자가 입력하는 값으로 대체되는 변수 및 강조에 사용됩니다.



는 태스크를 완료하는 데 도움이 되는 추가 정보를 나타냅니다.

**400** 은 IBM Eserver iSeries 400®(이전에는 AS/400®이라고 함)용 WebSphere Commerce에 고유하게 적용되는 정보를 표시합니다.

**AIX** 는 AIX®용 WebSphere Commerce에 고유하게 적용되는 정보를 표시합니다.

**Linux** Linux®용 WebSphere Commerce에 고유하게 적용되는 정보를 나타냅니다.

▶ **Solaris** Solaris Operating Environment software용 WebSphere Commerce에 고유하게 적용되는 정보를 나타냅니다.

▶ **Windows**

Windows®용 WebSphere Commerce에 고유하게 적용되는 정보를 나타냅니다.

▶ **Professional** WebSphere Commerce Professional Edition에 고유하게 적용되는 정보를 나타냅니다.

▶ **Business** WebSphere Commerce Business Edition에 고유하게 적용되는 정보를 나타냅니다.

## 경로 변수

이 책은 다음의 변수를 사용하여 디렉토리 경로를 나타냅니다.

### *WC\_installdir*

WebSphere Commerce의 설치 디렉토리입니다. 다음은 다양한 운영체제에서의 WebSphere Commerce의 기본 설치 디렉토리입니다.

▶ <b>AIX</b>	/usr/WebSphere/CommerceServer55
▶ <b>400</b>	/QIBM/ProdData/CommerceServer55
▶ <b>Linux</b>	/opt/WebSphere/CommerceServer55
▶ <b>Solaris</b>	/opt/WebSphere/CommerceServer55
▶ <b>Windows</b>	C:\Program Files\WebSphere\CommerceServer55

### ▶ **400** *WC\_userdir*

WebSphere Commerce에서 사용하는 모든 데이터에 대한 디렉토리로 사용자가 수정하거나 구성해야 합니다. 이러한 데이터의 예는 WebSphere Commerce 인스턴스 정보입니다. 이 디렉토리는 OS/400®의 경우 고유합니다.

*WC\_userdir* 변수는 다음 디렉토리를 나타냅니다.

/QIBM/UserData/CommerceServer55

### *WAS\_installdir*

WebSphere Application Server의 설치 디렉토리입니다. 다음은 다양한 운영체제에서의 WebSphere Application Server의 기본 설치 디렉토리입니다.

▶ <b>AIX</b>	/usr/WebSphere/AppServer
▶ <b>400</b>	/QIBM/ProdData/WebAS5
▶ <b>Linux</b>	/opt/WebSphere/AppServer
▶ <b>Solaris</b>	/opt/WebSphere/AppServer
▶ <b>Windows</b>	C:\Program Files\WebSphere\AppServer

### *WAS\_userdir*

WebSphere Application Server에서 사용하는 모든 데이터에 대한 디렉토리로

사용자가 수정할 수 있거나 구성해야 합니다.

▶ 400 /QIBM/UserData/WebAS5/Base/WAS\_instance\_name

---

## 제 1 장 WebSphere Commerce 관리 소개

이 책에서는 사이트 운영자가 공통으로 수행하는 태스크를 위주로 설명합니다. 이 책에서 설명되는 사이트 운영자 역할은 많은 회사의 시스템 관리자 역할과 같고 지금부터는 사이트 운영자라고 언급합니다.

사이트 운영자는 WebSphere Commerce에 제공되는 정의된 기본 역할 중 하나입니다. 일반적으로 사이트 운영자는 WebSphere Commerce 및 연관된 소프트웨어와 하드웨어를 설치, 구성 및 유지보수합니다. 사이트 운영자는 시스템 경고, 경보 및 오류에 응답하고 시스템 문제점을 진단하여 해결합니다. 이 역할은 일반적으로 액세스 및 권한(구성원을 작성하여 해당 역할에 지정하는 권한)을 제어하며 웹 사이트를 관리하고 성능을 모니터링하며 로드 밸런싱 태스크를 관리합니다. 또한 사이트 운영자는 테스트, 스테이징 및 프로덕션과 같은 서로 다른 단계의 개발에 대해 여러 서버 구성을 설정하고 유지보수합니다. 이 역할은 또한 중요한 시스템 백업을 처리하고 성능 문제를 해결합니다.

기타 WebSphere Commerce 역할에 대한 추가 정보는 WebSphere Commerce Production 온라인 도움말을 참조하십시오. 사이트 운영자는 다음 영역의 태스크를 수행합니다.

### 액세스 관리

- 사용자
- 조직
- 역할
- 액세스 그룹
- 정책
- 자원 그룹
- 조치 그룹

### 보안

- 계정 정책
- 암호 정책
- 계정 잠금 정책
- 보안 확인 프로그램

성능 시스템 성능 모니터

### 구성

- 전송
- 메시지 유형

- 메시지 로그 작성
- 추적
- 구성요소 구성
- 스케줄러
- 레지스트리
- 제품 정보

### WebSphere Commerce Payments

- 사용자
- 판매자 설정
- WebSphere Commerce Payments 설정
- 카세트
- 추적
- 액세스 관리
- 사이트의 메시지 전달 시스템 정의
- 프로덕션 서버로 규칙 서비스 전달
- WebSphere Commerce Payments 구성
- 캐시 구성 및 유지보수
- 스케줄러 구성
- 로그 작성 구성
- 데이터베이스 유지보수
- WebSphere Application Server 관리
- 웹 서버 관리
- 시스템 성능 모니터
- 사이트의 보안 유지
- 문제점 해결

---

## 관리 도구

사이트 운영자의 태스크를 더 쉽게 수행할 수 있도록 하는 여러 가지 도구가 있습니다. 기본 도구는 이 절에서 설명됩니다. 이후의 장에서는 기타 도구 및 이러한 도구를 사용하여 수행할 수 있는 태스크에 대해 설명합니다.

### 관리 콘솔

관리 콘솔을 사용하면 사이트 운영자가 사이트 구성 및 상점 구성과 관련된 태스크를 수행할 수 있습니다. 관리 콘솔 사용에 대한 자세한 내용은 19 페이지의 『관리 콘솔 열기』를 참조하십시오. 사이트 운영자가 관리 콘솔을 사용하여 수행하는 태스크는 다음과 같습니다.

- 사용자, 조직, 역할 및 구성원 그룹 관리
- 액세스 제어 관리



- 사이트의 전송 및 메시지 유형 정의
- WebSphere Commerce Payments 설정 지정
- WebSphere Commerce 구성요소의 사용 및 사용 안함
- 사이트에서 실행할 작업 스케줄
- 레지스트리 구성요소 갱신
- 상점 아카이브 공개

### 상점 공개 유틸리티

상점 공개 관리 유틸리티는 관리 콘솔에 들어 있으며 이를 사용하여 WebSphere Commerce에서 제공하는 샘플을 기초로 상점 아카이브를 신속하게 작성할 수 있습니다. 또한 이 유틸리티를 사용하여 상점에서 특정 기능(예: 협업)을 사용하거나 사용하지 않음으로써 공개된 상점을 구성할 수 있습니다. 상점 공개에 관한 추가 정보는 WebSphere Commerce 온라인 도움말의 "상점 아카이브 공개"를 참조하십시오.

### 구성 관리자

구성 관리자는 WebSphere Commerce를 전개하고 실행하는 데 필요한 인프라를 작성하고 변경합니다. 인프라의 설정에는 데이터베이스, 웹 서버 및 WebSphere Commerce Payments의 위치에 관한 정보와 이러한 응용프로그램을 WebSphere Commerce에서 사용하도록 구성하는 데 필요한 정보를 포함합니다. 또한 시스템 설정 정보 및 WebSphere Commerce 응용프로그램 자체를 초기화하는 데 필요한 정보의 판별도 포함됩니다. 구성 관리자 사용에 대한 자세한 내용은 14 페이지의 『WebSphere Commerce 구성 관리자 실행』 또는 *WebSphere Commerce 설치 안내서*를 참조하십시오.

구성 관리자를 사용하면 사이트 운영자가 구문이 중요한 구성 파일로 작업하지 않고도 관리 태스크 및 구성 태스크를 수행할 수 있습니다. 사이트 운영자가 수행할 수 있는 일부 기능은 다음과 같습니다.

- WebSphere Commerce 인스턴스의 작성 또는 삭제
- WebSphere Commerce 인스턴스의 구성 설정 변경
- 웹 서버 구성
- 데이터베이스가 스테이징 서버로 수행되도록 구성
- WebSphere Commerce Payments 매개변수 입력

### WebSphere Commerce Payments

WebSphere Commerce Payments는 인터넷 판매자에게 안전한 전자 지불 처리를 제공하는 WebSphere Commerce의 구성요소입니다. 개방형 표준 기반 기술을 바탕으로 WebSphere Commerce Payments는 지불 카세트에 대해 작업하여 다음을 포함한 여러 지불 프로토콜을 지원합니다.

- VisaNet은 지불을 승인하고 결산하는 기능을 갖는 전세계적인 원격 통신 데이터 및 지불 처리의 제공업체입니다.
- BankServACH는 ACH(Automated Clearing House Network)와 인터페이스하여 온라인 전자 수표 지불을 지원하는 지불 게이트웨이입니다.
- Paymentech는 Visa 및 MasterCard에 대한 직접 링크로 온라인 신용 카드 지불의 판매자 프로세서입니다.
- WebSphere Commerce Payments용으로 작성된 타사 지불 카세트입니다.

간단히 말하면 WebSphere Commerce Payments는 WebSphere Commerce와 통합되어 지불 처리 관리를 위한 현금 등록기와 같은 기능을 제공합니다. WebSphere Commerce Payments 사이트가 인터넷 판매자 상점 첫화면 뒤에 위치하여 지불을 수신하고 해당 지불을 은행 및 기타 금융 기관과 함께 처리하기 때문에 고객은 절대로 WebSphere Commerce Payments와 직접 상호작용하지 않습니다.

주: IBM WebSphere Commerce Payments(이하에서는 WebSphere Commerce Payments)를 이전에는 Payment Manager라고 했습니다. 버전 3.1.3에서 지불 응용프로그램이 WebSphere Commerce Payments로 이름이 변경되었으며 제품에 대한 참조가 이 문서 전체에서 변경되었습니다.

## 복수 지불 방법 수용

WebSphere Commerce Payments는 복수 지불 방법을 승인해야 하는 판매자를 수용하기 위한 유연하고 확장 가능한 방법을 제공하는 다중 지불 프레임워크 아키텍처를 구현합니다. 다중 지불 프레임워크는 지불 관리인 프레임워크를 특정 지불 방법인 소프트웨어 카세트에서 분리하기 때문에 각각이 독립적으로 발전할 수 있습니다.

WebSphere Commerce Payments는 각 지불 방법에 대한 소프트웨어 카세트가 지불 프레임워크에 접속되는 플러그인 아키텍처를 제공합니다. 프레임워크는 임의의 지불 방법을 사용한 지불 작성 및 수신에 필요한 일반 인프라 기능을 제공합니다.

지불 카세트는 WebSphere Commerce Payments 프레임워크의 데이터 플로우 및 제어 규칙을 준수하는 소프트웨어 응용프로그램입니다. 각 지불 카세트에는 특정 지불 방법 및 프로토콜의 구현이 들어있습니다.

카세트는 IBM 또는 타사의 지불 시스템 구현자가 작성할 수 있습니다. IBM은 카세트 개발을 지원하며 고유한 지불 카세트 작성에 관심있는 개발자에게 상세한 지시사항을 제공합니다. 카세트 개발에 대한 추가 정보는 <http://www.software.ibm.com/commerce/payments/download.html>의 *WebSphere Commerce Payments Cassette Kit Programming Guide* 및 다운로드를 참조하십시오.

## WebSphere Commerce Payments 사용자 인터페이스

WebSphere Commerce UI 또는 독립형 UI를 통해 Payments 기능에 액세스할 수 있습니다. 지불 관리자 및 호스팅된 WebSphere Commerce Payments를 사용하는 개별 판매자는 다음을 수행하는 데 사용할 수 있습니다.

- WebSphere Commerce Payments 구성
- 지불 중립적 방법으로 다음과 같은 일상적인 지불 처리 태스크 수행
  - 주문 승인
  - 지불 예치
  - 결산 일괄처리
  - 대변 발행
  - 매일의 결산 일괄처리 총계 보기

WebSphere Commerce Payments 사용자 인터페이스는 브라우저 기반이며 웹 브라우저의 SSL 기능을 사용하여 원격으로 안전하게 액세스할 수 있습니다.

## WebSphere Commerce Payments 역할

WebSphere Commerce Payments는 각 사용자에게 예를 들어, 지불 관리자 대 판매자 관리자의 시각에서 해당 사용자의 역할을 기초로 상이한 뷰가 제공되도록 역할을 제공합니다. 판매자 조직 안에서 WebSphere Commerce Payments는 다른 역할의 개념을 사용하므로 판매자가 자신의 사용자를 모니터링할 수 있습니다. 예를 들어, 점원은 주문 승인 같은 작업으로 제한되는 반면, 판매자 또는 지불 관리자는 금융 기관과의 관계를 수정할 수 있습니다.

Payments 프레임워크에는 WebSphere Commerce에서 지정된 역할에 대한 상대적인 매핑을 갖는 네 가지 역할이 있습니다. 이것은 자신이 사용자를 작성하고 역할을 지정하는 운영자인지를 아는 데 유용합니다. WebSphere Commerce 조직 관리 콘솔에서 사용자를 작성하는 경우, 우선 사용자에게 아래 표시된 WebSphere Commerce 역할을 지정해야 합니다. 그러면 사용자에게 대응하는 Payments 역할을 지정할 수 있는 Payments UI에 해당 사용자들이 표시됩니다. 다음 테이블은 Payments 사용자 역할을 대응하는 WebSphere Commerce 역할에 매핑합니다.

표 1. 역할 매핑

Payments 역할	WebSphere Commerce 역할
지불 관리자	사이트 운영자
판매자 관리자	사이트 운영자
감독자	작업 또는 판매 관리자
점원	고객 서비스 대표

지불 관리자와 판매 운영자 모두 WebSphere Commerce Payments를 관리할 수 있습니다. 감독자와 점원은 재무 역할입니다. WebSphere Commerce Payments를 관리하

지는 않더라도 지불 처리 기능을 관리합니다. 다음 테이블은 각 Payments 역할에 대한 책임을 설명합니다.

표 2. 역할 책임

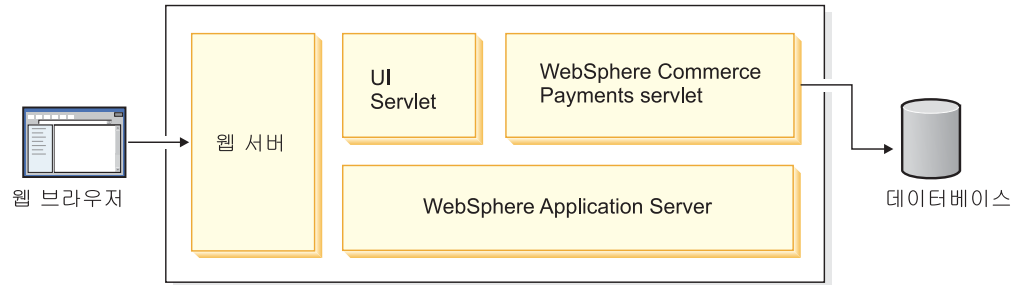
역할	책임
지불 관리자	<ul style="list-style-type: none"> <li>• 판매자 관리자, 감독자 및 점원 정의</li> <li>• 판매자 및 해당 카세트 구성</li> <li>• Payments 호스트 이름 및 상대 식별</li> <li>• 설치된 모든 카세트 구성</li> <li>• 이벤트 리스너 추가, 삭제 및 갱신</li> <li>• 지불 결산</li> <li>• 주문 승인 또는 판매</li> <li>• 대변 발행 및 대변 철회</li> <li>• 주문 예치</li> <li>• 주문 및 일괄처리 검색</li> <li>• 매일의 결산 일괄처리 총계 보기</li> </ul>
판매자 관리자	<ul style="list-style-type: none"> <li>• 판매자 관리자, 감독자 및 점원 정의</li> <li>• 판매자 및 해당 카세트 구성</li> <li>• 이벤트 리스너 추가, 삭제 및 갱신</li> </ul>
감독자	<ul style="list-style-type: none"> <li>• 지불 결산</li> <li>• 주문 승인 또는 판매</li> <li>• 대변 발행 및 대변 철회</li> <li>• 주문 예치</li> <li>• 주문 및 일괄처리 검색</li> <li>• 매일의 결산 일괄처리 총계 보기</li> </ul>
점원	<ul style="list-style-type: none"> <li>• 지불 결산</li> <li>• 주문 승인 또는 판매</li> <li>• 주문 예치</li> <li>• 주문 및 일괄처리 검색</li> <li>• 매일의 결산 일괄처리 총계 보기</li> </ul>

## WebSphere Commerce Payments 구성요소

WebSphere Commerce Payments를 이해하려면 다음 WebSphere Commerce Payments 구성요소를 참조하십시오.

- WebSphere Application Server로 구성된 웹 서버
- WebSphere Commerce Payments Servlets
- 사용자 인터페이스 Servlet
- 데이터베이스

다음 그림은 WebSphere Commerce Payments의 내부 구성요소를 보여줍니다.



WebSphere Commerce Payments Servlet은 WebSphere Commerce Payments의 주요 구성요소입니다. Servlet은 모든 플랫폼에 대해 일반 Servlet 환경을 제공하는 제품인 WebSphere Application Server와 함께 작동합니다.

이 절의 나머지는 WebSphere Commerce Payments Servlet 뿐 아니라 그 부수적인 지원 제품을 추가로 정의합니다.

- 웹 서버
- WebSphere Application Server
- 데이터베이스

**웹 서버:** 왜 웹 서버입니까? 이 장의 앞에서 언급한 것처럼, WebSphere Commerce Payments는 수신되는 HTTP 요청을 처리합니다. HTTP 요청은 먼저 WebSphere Application Server로 구성된 웹 서버에 의해 처리됩니다. 그런 다음 이들 요청이 다른 WebSphere Commerce Payments 구성요소로 중계됩니다.

**IBM WebSphere Application Server:** 웹 서버를 WebSphere Application Server와 함께 구성하는 이유는 무엇입니까? 두 가지 이유가 있습니다.

1. WebSphere Application Server는 복수 플랫폼에서 다양한 서버와 함께 작동하는 제품입니다.
2. WebSphere Application Server는 WebSphere Commerce Payments Servlet이 실행될 수 있는 환경을 제공합니다.

**WebSphere Commerce Payments Servlet:** Payment Servlet이라고도 부르는 WebSphere Commerce Payments Servlet이 WebSphere Commerce Payments의 핵심입니다. 사용자 인터페이스나 기타 판매자 응용프로그램에서 요청을 수신하고 보안 확인을 수행한 후 금융 네트워크와의 통신 같은 모든 필요한 지불 처리를 처리하는 지불 카세트를 호출하고 마지막으로 응답을 호출 응용프로그램으로 보냅니다. 이 구성요소가 WebSphere Commerce Payments 데이터베이스에 연결하고 데이터베이스에 저장된 보안 데이터에 액세스할 수 있는 유일한 구성요소입니다.

## WebSphere Commerce Payments 및 전자상거래 엔티티

이 절에는 WebSphere Commerce Payments가 판매자의 구매 소프트웨어를 지원하는 방법에 대해 알게 됩니다. 세 개의 시나리오가 WebSphere Commerce Payments가 판매자 소프트웨어 및 일반 지불 프로토콜과 결합하여 다음 전자상거래 엔티티와 상호작용하는 방법을 보여줍니다.

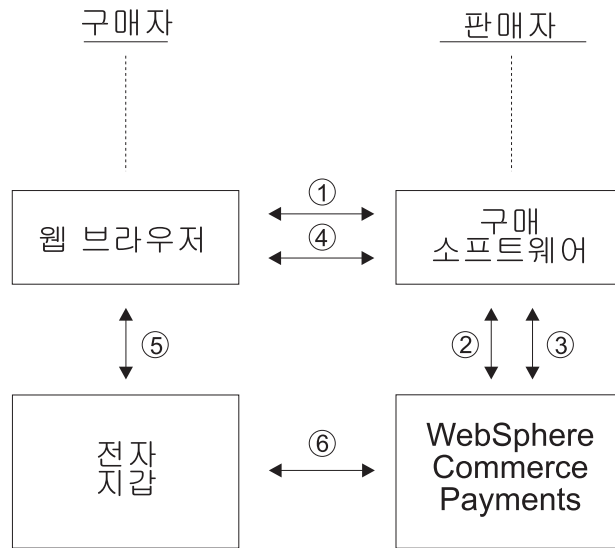
- 판매자
- 구매자
- 금융 기관

아래의 시나리오는 WebSphere Commerce Payments를 사용한 전자상거래를 보여줍니다. 이들 시나리오는 일반적이며 특정 지불 프로토콜에 고유한 어떤 상호작용도 의미하지 않음을 기억하는 것이 중요합니다. 오히려 많은 전자상거래 비즈니스 시나리오에 공통된 기본 트랜잭션을 탐색합니다.

**시나리오 1: 현금과 전자 지갑:** 첫 번째 시나리오는 가상 현금 프로토콜을 사용하고 구매자와 판매자의 두 엔티티를 포함합니다. 자금이 구매자에서 판매자로 전송될 것입니다. 이 트랜잭션에서 구매자는 프로토콜에 고유한 전자 지갑 응용프로그램을 사용합니다. 트랜잭션 단계가 다음과 같이 시나리오 도표에 표시됩니다.

1. 구매자와 판매자 구매 소프트웨어 간의 상호작용이 구매자가 판매자에게 구매 요청을 발행하는 것으로 종결됩니다.
2. 구매 요청에 대한 응답으로 판매자 소프트웨어가 지불 개시 명령을 보냅니다. 즉, WebSphere Commerce Payments에 지불을 수신합니다.
3. WebSphere Commerce Payments가 구매자의 지갑 프로그램에 대한 프로토콜 특정 정보를 포함하는 지불 개시 메시지로 응답합니다.
4. 판매자 소프트웨어가 이 지불 개시 메시지를 구매자의 브라우저로 전송합니다.
5. 구매자의 브라우저가 수신한 지불 개시 메시지가 구매자의 전자 지갑 프로그램을 시작합니다.
6. 구매자는 WebSphere Commerce Payments 및 전자 지갑 소프트웨어와의 상호작용으로 트랜잭션을 완료합니다.

**시나리오 1: 현금 및 전자 지갑과 상호작용하는 WebSphere Commerce Payments:**



이 시나리오가 아래에 나오는 두 시나리오와 다른 점은 다음 두 엔티티만을 포함하는 전자상거래 트랜잭션을 보여준다는 점입니다.

- 전자 지갑으로 표시되는 구매자
  - WebSphere Commerce Payments를 사용하는 판매자
- 금융 기관은 관련되지 않습니다.

**시나리오 2: 신용 카드와 전자 지갑:** 두 번째 시나리오는 가상 신용 카드 프로토콜을 사용한 구매를 보여주며 다음 세 엔티티가 관련됩니다.

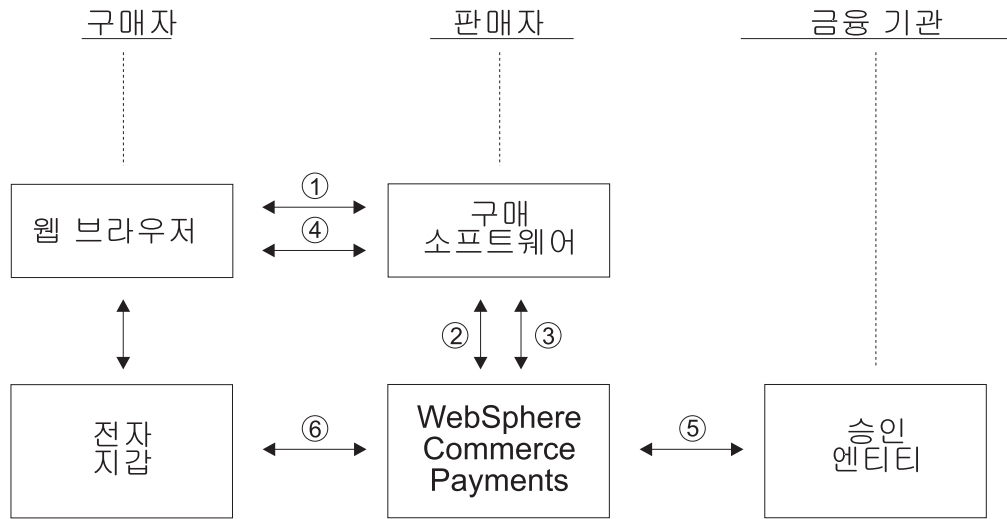
- 판매자
- 다시 전자 지갑을 사용하는 구매자
- 금융 기관

이 시나리오는 다음 차이점을 갖고 시나리오 1과 유사하게 진행됩니다.

- 구매자의 전자 지갑이 구매 요청을 완료할 때 WebSphere Commerce Payments가 충분한 카드 소유자 한도를 확인하기 위해 금융 기관의 승인 엔티티에 승인 요청을 발행합니다.
- 트랜잭션은 WebSphere Commerce Payments가 구매자의 전자 지갑에 승인 상태를 표시하는 확인 응답을 보내는 것으로 종결됩니다.

아래 시나리오 도표는 5와 6단계에서 이 차이점을 보여줍니다.

**시나리오 2: 신용 카드 및 전자 지갑과 상호작용하는 WebSphere Commerce Payments:**



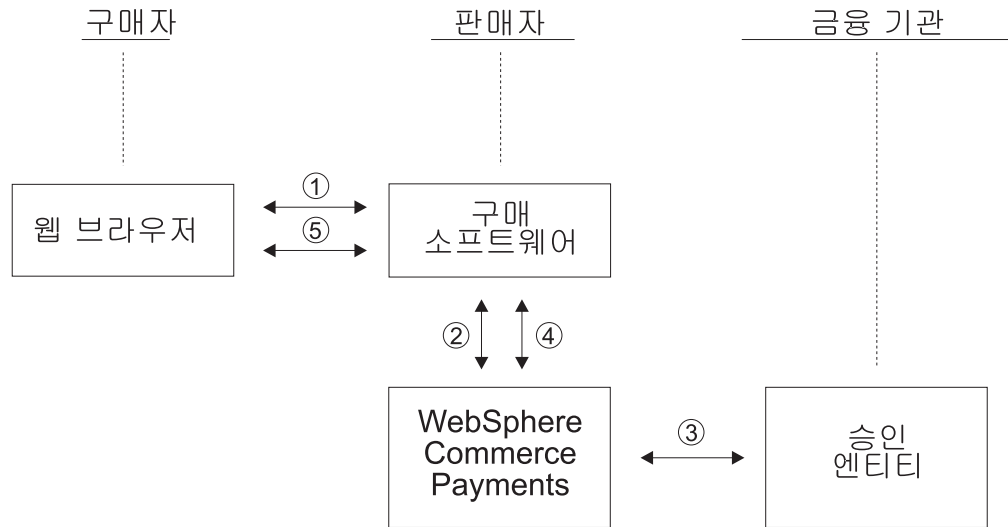
고객의 전자 지갑 프로그램이 시작되고 트랜잭션이 완료될 때, 금융 기관의 승인 엔티티가 충분한 카드 소유자 한도를 확인하는 반면 WebSphere Commerce Payments는 구매자에게 한도 예치금이 발생했음을 확인시킵니다.

**시나리오 3: 신용 카드와 SSL:** 마지막 시나리오는 구매자에게 전자 지갑이나 소비자 특정 소프트웨어가 없는 신용 카드 프로토콜을 설명합니다. 모든 소비자 정보가 안전한 방식, 예를 들어 SSL 소켓으로 판매자 소프트웨어에 전달됩니다. 트랜잭션 단계가 다음과 같이 시나리오 도표에 표시됩니다.

1. 구매자와 판매자 구매 소프트웨어 간의 상호작용이 구매자가 판매자에게 구매 요청을 발행하는 것으로 종결됩니다.
2. 구매 요청에 대한 응답으로 판매자 소프트웨어가 WebSphere Commerce Payments에 지불 개시 명령을 보냅니다.
3. WebSphere Commerce Payments가 승인 엔티티의 금융 기관을 통해 충분한 카드 소유자 한도를 확인하여 트랜잭션을 승인하려 시도합니다.
4. WebSphere Commerce Payments가 판매자 소프트웨어에 지불 예치가 발생했다는 결과를 리턴합니다.
5. 마지막으로 구매자에게 구매 요청의 성공 또는 실패가 통지됩니다.



### 시나리오 3: 신용 카드 및 SSL과 상호작용하는 WebSphere Commerce Payments



앞의 도표는 가능한 전자상거래 시나리오를 설명하고 해당 시나리오 안에서 WebSphere Commerce Payments의 역할을 보여줍니다. 다른 많은 비즈니스 플로우가 가능하며 WebSphere Commerce Payments는 해당 상호작용에서 비슷한 지불 기능을 가정합니다.

### 조직 관리 콘솔

조직 관리 콘솔을 사용하면 사이트 운영자가 사용자의 사이트나 상점에 액세스하는 조직을 제어할 수 있습니다. 수행할 수 있는 태스크는 여러 메뉴를 통해 조직 관리 콘솔 홈페이지에 표시됩니다. 이 태스크는 사용자 역할 및 권한 레벨에 기초합니다. 추가 정보에 대해서는 WebSphere Commerce 온라인 도움말의 조직 관리 콘솔 정보를 참조하십시오.

### WebSphere Commerce 액셀러레이터

WebSphere Commerce 액셀러레이터를 사용하면 상점의 외관 관리부터 주문 작성 및 유지보수, 상점 활동 추적까지 여러 상점 운영을 완료하여 온라인 상점을 유지보수할 수 있습니다. 각 역할로 수행할 수 있는 태스크는 WebSphere Commerce 액셀러레이터 홈페이지 메뉴에 표시됩니다. 이러한 태스크는 사이트 운영자가 관리 콘솔을 사용하여 정의하는 사용자 역할 및 권한 레벨에 기초합니다. 추가 정보는 WebSphere Commerce 온라인 도움말에 있는 WebSphere Commerce 액셀러레이터 정보를 참조하십시오.

주: 사이트 운영자가 조직 관리 콘솔 또는 WebSphere Commerce 액셀러레이터를 사용하여 수행하는 태스크는 온라인 도움말에 설명되어 있습니다.

## WebSphere Application Server

WebSphere Application Server는 인터넷 및 인트라넷 웹 응용프로그램을 빌드, 전개 및 관리하기 위한 Java 기반 응용프로그램 환경입니다. WebSphere Application Server는 또한 JSP(Java™ Server Page) 파일을 위한 런타임 지원을 제공합니다.

WebSphere Application Server의 관리 콘솔은 관리자가 다음을 수행할 수 있도록 도구 및 기능을 제공합니다.

- WebSphere Commerce 서버 중지 및 시작
- IBM WebSphere Commerce Payments 중지 및 시작
- 자원 설치 및 구성(예를 들어, servlets 및 엔터프라이즈 bean)
- 자원에 보안 지정
- 모든 응용프로그램이 사용 가능한지 확인
- 사용자 액세스 부여 또는 취소(예를 들어, 새 직원의 새 계정 작성 및 특정 사용자 그룹에 사용자 추가)
- Commerce Server 성능 모니터
- 성능 향상을 위해 응용프로그램 구성요소 복제
- 실행 중인 WebSphere 응용프로그램의 추적 및 디버깅 정보 제공

WebSphere Application Server 정보는 WebSphere Application Server 웹 사이트 (<http://www.ibm.com/software/webservers/appserv>)에서 볼 수 있습니다. WebSphere Application Server의 시스템 관리에 대한 자세한 정보는 다음 문서를 참조하십시오.

- 시스템 관리
- 조정 및 문제점 해결

## 데이터베이스

관리 데이터베이스 시스템으로 DB2® 또는 Oracle을 사용할 수 있습니다. 추가 정보는 각 시스템의 *WebSphere Commerce* 설치 안내서를 참조하십시오.

---

## 제 2 장 기본 관리 태스크

이 장에서는 정기적으로 또는 한 번만 수행해야 하는 기본 관리 태스크에 대해 설명합니다. 비즈니스 프로세스에 따라 각 태스크를 반드시 수행하지 않아도 됩니다.

---

### 1회 수행 관리 태스크

다음 태스크는 대개 한 번만 수행됩니다.

#### 설치 및 구성

사이트 운영자로 수행하는 태스크 중 하나는 WebSphere Commerce 사이트를 설치하고 구성하는 것입니다. 일반적으로 이것은 첫 번째로 수행해야 하는 단계입니다. 대개 설치에는 사용 중인 운영체제에 따른 WebSphere Commerce 소프트웨어의 설치가 포함됩니다. 설치 단계는 제품과 함께 제공되는 설치 안내서에서 찾을 수 있습니다. 설치 및 구성에 대한 추가 정보는 *WebSphere Commerce 설치 안내서*를 참조하십시오. 추가 구성 정보는 WebSphere Commerce 온라인 도움말에서 볼 수 있습니다.

---

### 정기적인 관리 태스크

다음 태스크는 정기적으로 수행됩니다.

- WebSphere Commerce의 시작 및 중지
- 구성 관리자 열기
- WebSphere Commerce 인스턴스의 작성, 구성 및 삭제
- 관리 콘솔 열기
- 조직 관리 콘솔 열기
- WebSphere Commerce 구성요소의 사용 및 사용 안함
- WebSphere Commerce Payments의 시작 및 중지

#### WebSphere Commerce의 시작 및 중지

WebSphere Commerce의 각 인스턴스에는 WebSphere Commerce Server가 있습니다. WebSphere Commerce Server는 WebSphere Commerce 시스템의 정보 플로우를 제어하는 데 사용되는 Java 기반의 Commerce Server입니다. WebSphere Commerce Server의 각 구성요소에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 "WebSphere Commerce Server"를 참조하십시오.


WebSphere Commerce 인스턴스는 WC\_instance\_name이라는 응용프로그램 서버에서 시작되고 중지되며 여기서 instance\_name은 시작 또는 중지하려는 WebSphere

Commerce 인스턴스의 이름입니다. 예를 들어 demo라는 WebSphere Commerce 인스턴스를 시작하려면 WC\_demo라는 응용프로그램 서버를 시작해야 합니다.

WebSphere Commerce 인스턴스를 시작 또는 중지하려면 다음을 수행하십시오.

1. 시작 또는 중지하려는 WebSphere Commerce 인스턴스를 결정하십시오.
2. 사용자 플랫폼에 대한 응용프로그램 서버 시작 및 중지에 대한 지시사항을 따르십시오. 이들 지시사항에 대한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말에 있는 "<사용자 플랫폼>에서 응용프로그램 서버 시작 및 중지"를 참조하십시오.

WebSphere Commerce Payments가 아직 시작되지 않은 경우, WebSphere Commerce를 시작한 후에 시작할 수 있습니다. WebSphere Commerce Payments 시작에 관한 지시사항은 22 페이지의 『WebSphere Commerce Payments 시작 및 중지』를 참조하십시오.

주:  WebSphere Commerce Studio에서 WebSphere Commerce 시작에 대한 지시사항은 WebSphere Commerce Production 온라인 도움말의 "WebSphere Commerce 시작 및 중지"를 참조하십시오.

## WebSphere Commerce 구성 관리자 실행

구성 관리자를 사용하면 WebSphere Commerce의 여러 구성요소를 사용하고 구성할 수 있습니다. 가능한 경우, 구성요소는 WebSphere Commerce 구성 파일을 수정하지 않고 구성 관리자를 통해 사용 가능해야 합니다. WebSphere Commerce 인스턴스를 작성하거나 구성하려면 구성 관리자를 사용해야 합니다.

### AIX, Linux 및 Solaris 소프트웨어에서 WebSphere Commerce 구성 관리자 실행

AIX, Linux 및 Solaris 소프트웨어에서 WebSphere Commerce 구성 관리자를 실행하려면 다음을 수행하십시오.

1. 다음을 수행하여 서버를 시작하십시오.
  - a. 단말기 창을 여십시오.
  - b. 설치 후 스크립트가 WebSphere Commerce 설치의 일부로 실행될 때 작성된 WebSphere Application Server 사용자로 로그인하십시오. 기본 WebSphere Application Server 사용자 이름은 *wasuser*입니다.
  - c. 다음 명령을 실행하십시오.

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /usr/WebSphere/CommerceServer55/bin
./config_server.sh
```

▶ Linux

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /opt/WebSphere/CommerceServer55/bin
./config_server.sh
```

여기서 *host\_name*은 구성 관리자에 액세스하려는 시스템의 완전한 호스트 이름입니다.

주:

- 1) `config_server.sh` 명령을 입력한 단말기 창을 닫지 마십시오. 그렇지 않으면 구성 관리자 서버가 중지됩니다.
  - 2) 백그라운드 프로세스로 구성 관리자 서버를 실행하지 마십시오. 이는 잠재적인 보안 위험입니다.
2. 다음을 수행하여 클라이언트를 시작하십시오.
- a. 다른 단말기 창을 여십시오.
  - b. 설치 후 스크립트가 WebSphere Commerce 설치의 일부로 실행될 때 작성된 WebSphere Application Server 사용자로 로그인하십시오. 기본 WebSphere Application Server 사용자 이름은 *wasuser*입니다.
  - c. 다음 명령을 실행하십시오.

▶ AIX    ▶ Solaris

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /usr/WebSphere/CommerceServer55/bin
./config_client.sh &
```

▶ Linux

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /opt/WebSphere/CommerceServer55/bin
./config_client.sh &
```

여기서 *host\_name*은 구성 관리자에 액세스하려는 시스템의 완전한 호스트 이름입니다.

## iSeries에서 WebSphere Commerce 구성 관리자 실행

iSeries에서 WebSphere Commerce 구성 관리자를 실행하려면 다음을 수행하십시오.

1. 다음을 수행하여 서버를 시작하십시오.
  - a. iSeries에 로그인하여 프로파일이 \*SECOFR 사용자 클래스를 가지고 있고 영어나, 사용자 인스턴스의 기본 언어로 선택할 언어 중 하나의 언어별 설정으로 설치되어 있는지 확인하십시오.

- b. 다음 명령을 입력하여 QShell 세션을 시작하십시오.

```
STRQSH
```

그리고 QShell 세션에서 다음을 수행하십시오.

- 1) 다음 명령을 발행하여 WebSphere Commerce 서버 bin 디렉토리로 전환하십시오.

```
cd /qibm/proddata/commerceserver55/bin
```

- 2) 다음 명령을 실행하여 구성 관리자 서버 프로그램을 시작하십시오.

```
config_server.sh -port server_port_number
```

여기서 *server\_port\_number*은 구성 관리자가 청구할 iSeries 서버의 포트 번호입니다. 이 매개변수는 선택적이며 기본값은 1099입니다. 이 값은 1024 - 65535 범위에 있어야 하며 현재 사용하지 않습니다.

주: 기본 언어가 사용자 인스턴스를 작성 중인 언어와 동일하지 않은 시스템을 사용 중인 경우, QSYSlanguage\_feature\_number 라이브러리를 사용자 프로파일의 라이브러리 목록에 추가해야 합니다. 그렇지 않으면 프로파일은 QSYS에서 이를 찾습니다. 언어 기능 라이브러리를 추가하려면, EDTLIBL 명령을 사용하십시오.

- c. 구성 관리자가 시스템에서 처음 실행하는 경우, 다음 메시지를 보게 됩니다.

```
Attaching Java program to /QIBM/ProdData/WebCommerce55/lib/wcsconfig.JAR.  
Attaching Java program /QIBM/ProdData/WebCommerce55/lib/wcsruntme.JAR.  
Attaching Java program to /QIBM/ProdData/WebCommerce55/wc.ear/lib  
/wcslogging.JAR.  
Attaching Java program to /QIBM/ProdData/WebCommerce55/lib/xml4j.JAR.  
Attaching Java program to /QIBM/ProdData/WebCommerce55/lib/sslite.ZIP.
```

- d. 다음 메시지가 나타나면 다음 단계를 시작하십시오.

레지스트리가 작성되었습니다. 레지스트리에서 CMServe가 바인드되었습니다.

2. 다음을 수행하여 클라이언트를 시작하십시오.

- a. 클라이언트 시스템의 명령 프롬프트를 사용하여 WCS400 디렉토리로 변경하십시오.

- b. 다음 명령을 실행하여 클라이언트를 구성하십시오.

```
config_client.bat iSeries_Host_name Server_port_number
```

여기서

*iSeries\_Host\_name*

서버의 완전한 호스트 이름입니다.

*Server\_port\_number*

구성 관리자가 청구 중인 iSeries 서버의 포트 번호입니다.

## Windows에서 WebSphere Commerce 구성 관리자 실행

Windows에서 WebSphere Commerce 구성 관리자를 실행하려면 다음을 수행하십시오.

1. 시작 → 설정 → 제어판 → 관리 도구 → 서비스를 선택하여 IBM WC 구성 관리자 서버 프로세스가 실행 중인지 확인하고 IBM WC 구성 관리자 서비스가 시작된 상태를 가지는지 확인하십시오.

### 중요

IBM WC 구성 관리자 서비스를 실행 중인 상태로 두면 보안 문제점이 발생할 수 있습니다. 구성 관리자를 사용하지 않은 경우에는 WC 구성 관리자 서버 서비스를 중지하십시오.

보안 문제점을 방지하려면 IBM WC 구성 관리자 서버가 자동이 아닌 수동 시작으로 설정되어 있는지 확인하십시오.

2. 시작 → IBM WebSphere Commerce → 구성을 선택하십시오.

## 인스턴스 작성, 구성 및 삭제

WebSphere Commerce 인스턴스를 작성, 구성 또는 삭제하려면 구성 관리자를 사용해야 합니다. 이 도구는 잘못된 값을 입력하지 않도록 하고 인스턴스를 구성할 수 있도록 설계되었습니다.

대부분의 WebSphere Commerce 구성 매개변수를 수정하려면 구성 관리자를 사용해야 합니다. 그러나 일부 매개변수는 WebSphere Commerce 구성 파일(instance\_name.xml)에서 변경해야 합니다. 구성 관리자에서 구성해야 하는지 또는 WebSphere Commerce 구성 파일(instance\_name.xml)에서 구성해야 하는지 판별하려면 각 구성요소의 문서를 참조하십시오.

### 인스턴스 작성 및 구성

인스턴스를 작성하려면 다음을 수행하십시오.

1. 구성 관리자를 여십시오. 구성 관리자를 여는 방법에 대해서는 14 페이지의 『WebSphere Commerce 구성 관리자 실행』을 참조하십시오.
2. 시스템 이름의 노드를 펼치고 인스턴스 목록을 선택한 후 조치 메뉴에서 인스턴스 작성을 선택하십시오.
3. 인스턴스 작성 마법사가 인스턴스 작성 프로세스를 안내해 줍니다. 패널에서 도움이 필요하면 도움말을 선택하십시오.


인스턴스를 구성하려면 다음을 수행하십시오.

1. 구성 관리자를 여십시오.

2. 시스템 이름의 노드를 펼치고 인스턴스 목록을 펼친 다음 수정하려는 인스턴스를 선택하십시오.
3. 수정할 구성요소를 선택하십시오.
4. 구성요소를 선택하면 해당 패널이 열립니다. 그러면 필요에 따라 구성요소를 수정할 수 있습니다.
5. 구성요소가 없으면 마우스 오른쪽 단추로 눌러 구성요소를 작성할 수 있습니다. 마법사가 구성요소 작성 프로세스를 안내해 줍니다.




## 인스턴스 삭제

WebSphere Commerce 인스턴스를 삭제하려면 다음을 수행하십시오.


1. WebSphere Commerce 인스턴스를 백업하십시오(권장됨).
2. `WC_installdir\instances\instance_name\xml`() `WC_userdir/instances/instance_name/xml` 디렉토리로 변경하여 `instance_name.xml` 파일을 백업 디렉토리에 복사하십시오.
3. WebSphere Application Server를 시작하십시오.
4. 다음을 수행하여 WebSphere Commerce Server를 제거하십시오.

-  QSHHELL 세션 명령행에서 다음을 입력하십시오.

```
WC_installdir/bin/rmCommerceServer.sh instance_name
```


-    명령 디렉토리 서비스에서 다음을 입력하십시오.

```
WC_userdir/bin
./rmCommerceServer.sh instance_name
```

-  명령 디렉토리 서비스에서 다음을 입력하십시오.

```
WC_installdir/bin
./rmCommerceServer instance_name
```

5. WebSphere Commerce 구성 관리자에서 인스턴스를 마우스 오른쪽 단추로 누르고 삭제를 선택하십시오.
6. WebSphere Commerce Business Edition 데이터베이스를 삭제하십시오.

-  DB2 데이터베이스를 사용하는 경우, 다음을 수행하십시오.
  - a. DB2 명령창을 열고 다음을 입력하십시오.

```
db2 drop db db_name
db2 uncatalog node node_name
```

여기서 `db_name`은 구성 관리자에서 입력한 WebSphere Commerce 데이터베이스의 이름이고 `node_name`은 이러한 데이터베이스에 대해 카탈로그를 작성한 노드의 이름입니다. 기본 노드 이름은 시스템 호스트 이름의 짧은(즉, 완전하지 않은) 버전입니다.

-  Oracle 데이터베이스를 사용하는 경우 다음을 수행하십시오.



- a. SQL\*Plus 세션을 시작하고 다음을 입력하십시오.

```
sqlplus system/manager@wc_SID
```

여기서 *wc\_SID*는 WebSphere Commerce 데이터베이스의 SID(Oracle System Identifier)입니다.

- b. SQL\*Plus 세션에서 다음 명령을 실행하십시오.

```
DROP TABLESPACE wc_tablespace_name
INCLUDING CONTENTS;
DROP USER wc_Oracle_user_ID
CASCADE;
EXIT;
```

주: 또한, 실제 테이블 공간 파일을 제거해야 합니다. 테이블 공간 파일은 *wc\_SID.ora*이고 여기서 *wc\_SID*는 WebSphere Commerce 데이터베이스의 SID(Oracle System Identifier)입니다.

▶ 400 다음의 SQL문을 실행하여 인스턴스 데이터베이스 라이브러리를 삭제하십시오. `drop schema instance_name`. 스키마 삭제를 시도하고 리시버가 첨부되어 있는 동안 리시버를 삭제할 수 없다는 메시지를 수신하면 다음의 명령을 시도해 보십시오. `ENDJRNPF FILE(*ALL) JRN(instance_name/QSQRN)`

7. WebSphere Commerce 인스턴스를 완전히 삭제하려면 다음 디렉토리를 삭제하십시오.

- ▶ AIX ▶ Solaris ▶ Linux ▶ Windows `WC_installdir/instances/instance_name`

- ▶ 400 `WC_userdir/instances/instance_name`

+

8. ▶ 400 DLTUSRPF 명령을 사용하여 인스턴스 사용자 프로파일을 삭제하십시오.

## 관리 콘솔 열기

관리 태스크를 수행할 때마다 대부분 관리 콘솔에서 수행해야 합니다. 관리 콘솔에서는 관리 조작 및 구성 태스크를 수행하여 사이트 또는 상점을 제어할 수 있습니다. 관리 콘솔에 대한 추가 정보는 2 페이지의 『관리 콘솔』을 참조하십시오.

1. 다음 중 하나를 수행하여 관리 콘솔을 시작하십시오.

- 브라우저에서 다음 URL에 액세스하십시오.

```
https://host_name:8002/adminconsole
```

여기서 *host\_name*은 WebSphere Commerce Server의 완전한 TCP/IP 이름입니다.

- ▶ Studio WebSphere Commerce Studio를 설치하고 이 시스템에서 관리 콘솔을 열려는 경우, 다음을 수행하십시오.

- a. WebSphere Commerce Studio에서 WebSphere Commerce를 시작했는지 확인하십시오. WebSphere Commerce Studio에서 WebSphere Commerce 시

작에 대한 지시사항은 WebSphere Commerce Production 온라인 도움말의 "WebSphere Commerce Studio에서 WebSphere Commerce 시작 및 중지"를 참조하십시오.

- b. 브라우저에서 다음 웹 주소에 액세스하십시오.

```
https://host_name/webapp/wcs/tools/servlet/ToolsLogon?
XMLFile=adminconsole.AdminConsoleLogon
```

여기서 *host\_name*은 WebSphere Commerce Studio 시스템의 완전한 HTTP 호스트 이름입니다.

2. 다음을 수행하여 관리 콘솔에 로그인하십시오.
  - a. 로그인 페이지에서 관리 콘솔의 로그인 사용자 이름과 암호를 입력하고 로그인을 누르십시오. 둘 이상의 상점이나 언어를 사용하여 작업할 수 있는 권한이 있으면, 관리 콘솔 사이트/상점 선택 페이지가 표시됩니다. 단일 상점과 언어를 사용하여 작업할 수 있는 권한이 있으면 관리 콘솔 홈페이지가 표시됩니다. 수행할 수 있는 권한이 부여된 태스크들이 관리 콘솔 홈페이지에 표시됩니다.
3. 관리 콘솔 사이트/상점 선택 페이지에서 사이트 또는 상점 라디오 버튼을 선택하고 확인을 누르십시오.
  - 사이트를 선택하면 사이트 관리 콘솔 홈페이지가 표시됩니다.
  - 상점을 선택하면 이름과 언어 드롭 다운 상자에서 상점 이름과 언어를 선택할 수 있습니다. 관리 콘솔 홈페이지가 표시됩니다.

## 조직 관리 콘솔 열기

다음 중 하나를 수행하여 조직 관리 콘솔에 로그인하십시오.

- 다음 중 하나를 수행하여 조직 관리 콘솔을 시작하십시오.
  - 브라우저에서 URL `https://host_name:8004/orgadminconsole`에 액세스하십시오.

여기서 *host\_name*은 WebSphere Commerce Server의 완전한 TCP/IP 이름입니다.

- WebSphere Commerce Studio를 설치하고 이 시스템에서 조직 관리 콘솔을 열려면, 다음을 수행하십시오.

1. WebSphere Commerce Studio에서 WebSphere Commerce를 시작했는지 확인하십시오. 추가 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.
2. 브라우저에서 다음 웹 주소에 액세스하십시오.

```
https://host_name/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=
buyerconsole.BuyAdminConsoleLogon
```

여기서 *host\_name*은 WebSphere Commerce Studio 시스템의 완전한 HTTP 호스트 이름입니다.

- 조직 관리 콘솔에 로그인하려면 다음을 수행하십시오.

1. 로그인 페이지에서 조직 관리 콘솔 로그인 사용자 이름과 암호를 입력하고 로그인을 누르십시오. 둘 이상의 상점이나 언어에 대해 작업할 수 있는 권한이 있으면 조직 관리 콘솔 홈페이지가 표시됩니다. 수행할 수 있는 권한이 부여된 태스크들이 조직 관리 콘솔 홈페이지에 표시됩니다.
- 조직 관리 콘솔에 로그인하여 관리 콘솔의 암호를 변경하려면 다음을 수행하십시오.
  1. 로그인 페이지에서 조직 관리 콘솔 로그인 사용자 이름과 암호를 입력하십시오.
  2. 암호 변경 선택란을 선택하고 로그인을 누르십시오. 암호 변경 페이지가 표시됩니다.
  3. 기존 암호 필드에 현재 조직 관리 콘솔 로그인 암호를 입력하십시오. 이 필드에는 최대 128자까지 영숫자를 입력할 수 있습니다.
  4. 새 암호 필드에 새 로그인 암호를 입력하십시오. 이 필드에는 최대 128자까지 영숫자를 입력할 수 있습니다.
  5. 새 암호 확인 필드에 새 로그인 암호를 다시 입력하십시오.
  6. 변경을 눌러 새 암호를 저장하십시오. 조직 관리 콘솔 홈페이지가 표시됩니다. 수행할 수 있는 권한이 부여된 태스크들이 조직 관리 콘솔 홈페이지에 표시됩니다.

## WebSphere Commerce 구성요소의 사용 및 사용 안함

운영자는 특정 시간에 WebSphere Commerce의 특정 구성요소만 사용 가능하게 하기를 원할 수 있습니다. 고객의 서로 다른 요구에 따라 WebSphere Commerce는 관리 콘솔의 구성 메뉴를 사용하여 필요할 경우 WebSphere Commerce 구성요소를 사용하거나 사용하지 않을 수 있습니다.

WebSphere Commerce 구성요소를 사용하거나 사용하지 않으려면 다음을 수행하십시오.

1. 관리 콘솔을 열고 사이트 운영자로 로그인하십시오.
2. 구성 메뉴에서 구성요소 구성을 누르십시오. 사이트에서 사용 가능한 모든 구성요소를 표시하는 페이지가 표시됩니다.
3. 구성요소를 사용하려면 다음을 수행하십시오.
  - a. 사용 가능한 구성요소 목록에서 사용하려는 구성요소를 선택하고 추가를 누르십시오.
  - b. 사용하려는 모든 구성요소가 선택한 구성요소 목록에 있는지 확인하고 확인을 누르십시오.

주: 구성요소를 사용하기 전에 해당 구성요소의 문서를 참조하여 구성요소가 제대로 구성되었는지 확인하십시오.
4. 구성요소를 사용하지 않으려면 다음을 수행하십시오.
  - a. 선택한 구성요소 목록에서 사용하지 않으려는 구성요소를 선택하고 제거를 누르십시오.

- b. 사용하지 않으려는 모든 구성요소가 사용 가능한 구성요소 목록에 있는지 확인하고 확인을 누르십시오.

## WebSphere Commerce Payments 시작 및 중지

WebSphere Commerce Payments는 온라인 판매자를 위한 프로토콜과 무관한 지불 트랜잭션 서버입니다. Payment Manager는 프로토콜 고유 카세트를 사용한 복수의 지불 방법을 지원하여 현금 등록기와 유사한 기능을 제공합니다. 이러한 카세트는 소프트웨어 구성요소로, 지불 프레임워크에 연결하여 일반 지불 및 관리 명령을 지불 프로토콜별 요청으로 해석합니다. 그런 다음, 이 요청은 승인처의 지불 게이트웨이와 같은 적절한 수신자에게 전달됩니다. 최종 결과는 계산원이 보통 상점의 계산대에서 지불 카드를 긁는 것과 유사합니다. Payments는 판매자(또는 상점 그룹과 같은 판매자 그룹) 대신 인터넷 지불의 모든 배경 정보를 처리하고 트랜잭션 관리를 단순화할 수 있도록 그래픽 인터페이스를 제공합니다. Payments는 다음 기능을 수행합니다.

- 우선, 카세트에 따라 모든 필수 승인처 인증서가 있는지 확인합니다. 인증서가 없으면 자동으로 승인처 또는 지불 게이트웨이에 요청합니다.
- 지속적으로 실행되면 전용 포트에서 대기하다가 Wallet 및 판매자 서버로부터 수신되는 모든 요청을 처리합니다.
- 카세트에 따라 TCP/IP 프로토콜이나 기타 프로토콜을 사용하여 인터넷을 통해 승인처와 통신합니다.
- 데이터베이스 테이블을 사용하여 트랜잭션, 승인 요청 및 예치 요청의 상태에 관한 정보를 관리합니다. 테이블의 레코드는 작업 목적으로 보관되어 레코드 추적과 유지에 사용될 수 있습니다. 데이터베이스에는 판매자, 지불 카드 브랜드 및 판매자에게 카드 인증 및 지불 캡처 서비스를 제공하는 승인처에 대한 정보를 보유하는 구성 테이블도 있습니다.

판매자 서버는 Payments API를 사용하여 Payments와 상호작용합니다. 이러한 API는 다른 지불 기술 및 SET™(Secure Electronic Transactions)를 지원할 수 있도록 일반적으로 설계되었습니다.

관리자는 Payments 서비스를 시작하고 중지해야 합니다. 지불 엔진을 시작하고 중지하려면 Payments가 설치된 디렉토리에 .payment 파일이 있어야 합니다. .payment 파일은 Payments 설치 중에 작성됩니다.

## WebSphere Commerce Payments 시작

WebSphere Commerce Payments를 시작하려면 다음을 수행하십시오.

1. WebSphere Application Server 관리 콘솔을 시작하고 WebSphere Payments Application Server가 시작되었는지 확인하십시오.
2. 다음과 같이 로그인하십시오.

▶ AIX
▶ Solaris
▶ Linux
▶ 400
 데이터베이스 인스턴스 ID로 로그인하십시오

오.

▶ Windows 관리자 그룹 권한이 있는 사용자로 로그인하십시오.

3. WebSphere Commerce Payments가 설치된 디렉토리로 이동하십시오. 기본 설치 위치는 다음과 같습니다.

▶ AIX /usr/lpp/WebSphere/CommerceServer55/payments

▶ 400 /QIBM/ProdData/CommercePayments/V55

▶ Solaris ▶ Linux /opt/WebSphere/CommerceServer55/payments

▶ Windows drive:\Program Files\IBM\WebSphere Commerce Payments

4. 다음 명령을 입력하십시오.

▶ AIX ▶ Solaris ▶ Linux ./IBMPayServer

▶ Windows IBMPayServer

▶ 400 이 명령의 실행 후에 암호를 지정해야 합니다.

주: WebSphere Commerce Payments를 구성할 때 암호 필수를 선택한 경우, WebSphere Commerce Payments 응용프로그램을 완전히 초기화하려면 IBMPayServer 명령을 호출해야 합니다. IBMPayServer 명령은 데이터베이스에 저장되는 중요한 데이터를 암호 해독하는 데 사용되는 WebSphere Commerce Payments 암호를 지정하는 데 사용됩니다. 이 명령에 대한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 "IBMPayServer 명령 실행"을 참조하십시오. WebSphere Commerce Payments 구성에 대한 추가 정보는 82 페이지의 『WebSphere Commerce Payments 구성』을 참조하십시오. WebSphere Commerce Payments 구성에 대한 자세한 정보는 181 페이지의 부록 C 『WebSphere Commerce Payments 학습서』를 참조하십시오.

## WebSphere Commerce Payments 중지

WebSphere Commerce Payments를 중지하려면 다음을 수행해야 합니다.

- WebSphere Commerce Payments를 중지하십시오.
- WebSphere Application Server에서 Payment Manager Application Server를 중지하십시오.

WebSphere Commerce Payments를 중지하려면 22 페이지의 『WebSphere Commerce Payments 시작』에 대해 1-3단계를 완료하고 4단계에서 다음 명령을 사용하십시오.

▶ AIX ▶ Solaris ▶ Linux ./StopIBMPayServer

▶ Windows StopIBMPayServer

주: 구성 관리자 또는 WebSphere Commerce Studio에서 Commerce Payments 중지  
지에 대한 정보는 WebSphere Commerce Studio의 "WebSphere Commerce Payments 시작 및 중지"와 WebSphere Commerce Production 온라인 도움말의 "구성 관리자에서 WebSphere Commerce Payments 시작 및 중지"를 참조하십시오.  
오.

---

## 설치 안내서의 관리 태스크

*WebSphere Commerce* 설치 안내서에서는 다음 관리 태스크에 대해 설명합니다.

- WebSphere Commerce 인스턴스 또는 Commerce Payments 인스턴스 수정
- WebSphere Commerce 인스턴스 삭제
- 기타 명령행 구성 태스크
- 원격 DB2 데이터베이스 카탈로그
- WebSphere Commerce 암호화 암호 작성
- WebSphere Commerce Payments 암호화 암호 작성

---

## 제 3 장 구성

이 장에서는 사이트 운영자가 WebSphere Commerce 사이트의 최적 기능을 위해 사용하는 구성 도구 및 태스크에 대해 설명합니다.

---

### 구성 서비스

#### 구조적 구성

사이트 운영자는 설계자와 함께 WebSphere Commerce가 실행될 구성을 구현합니다. 예를 들어, WebSphere Commerce 및 필수 스택(웹 서버, WAS, 데이터베이스 및 Commerce Payments)이 모두 동일한 실제 시스템이나 노드에 위치하여 단일 노드 아키텍처를 구성하거나, WebSphere Commerce의 여러 계층(소프트웨어 제품)이 서로 다른 실제 노드(시스템)에 위치하여 복수 노드 아키텍처를 구성할 수 있습니다. 솔루션의 복잡도에 따라 실제로 서로 다른 시스템이나 복제 간에 로드 공유가 구현된 클러스터 환경이 있을 수 있습니다. 사이트 운영자는 테스트, 스테이징 및 프로덕션과 같은 여러 개발 단계에 대해 여러 서버 구성을 설정하고 유지보수도 담당합니다. WebSphere Commerce에서 사용할 수 있는 다른 구성에 대한 추가 정보는 *WebSphere Commerce 설치 안내서*를 참조하십시오.

#### 메시지 전달 서비스

관리 콘솔의 메시지 전달 서비스 인터페이스를 사용하면 사이트 운영자가 사이트 및 상점의 메시지 전달을 설정하고 관리할 수 있습니다. WebSphere Commerce 메시지 전달 시스템을 통해 WebSphere Commerce 내에서 작성되는 메시지의 정의 및 전송에 대한 모든 측면을 관리할 수 있습니다. 메시지 전달 시스템은 전자 우편 및 텍스트 파일과 같은 전송을 통해 메시지를 보낼 수 있습니다. 전자 우편의 경우, 지원되는 아웃바운드 프로토콜은 SMTP이고, 메시지 인코딩은 지정된 언어에 따라 달라집니다. 일반 파일 메시지는 UTF-8 인코딩 표준을 사용합니다. 선택적으로 WebSphere MQ 및 IBM CrossWorlds<sup>®</sup>를 사용하여 메시지 전달 시스템이 백엔드 시스템으로 메시지를 보내도록 구성할 수 있습니다. 아웃바운드 메시지를 보내고 일정 기간동안 응답 메시지를 기다리도록 메시지 전달 서비스를 구성할 수도 있습니다. 아웃바운드 메시지 전달 시스템을 구성하려면 관리 콘솔의 구성 메뉴를 사용하십시오.

WebSphere Commerce는 J2EE/CA(Java 2 Enterprise Edition Connector Architecture)를 구현하는 플러그인 모델을 사용하여 시스템과 다양한 전송 간의 공통 인터페이스를 제공합니다. 외부 전송과의 공통 인터페이스를 사용하면 전송의 구현 정보를 메시지 전달 시스템의 작동과 분리시킬 수 있습니다. 이 아키텍처는 J2EE/CA 인터페이스를 준



수하는 추가 전송을 플러그인하는 것을 가능하게 합니다. 이 방법으로, 사용자의 비즈니스 필요 및 환경에 맞게 솔루션을 쉽게 사용자 정의할 수 있습니다.

## 스케줄 서비스

스케줄러는 주로 작업을 스케줄하고 시간 설계에 따라 작업을 실행하는 데 사용되는 Commerc Server의 구성요소입니다. 스케줄된 각 작업은 개별 스레드로 실행됩니다. 복수 작업을 동시에 실행하도록 스케줄할 수는 있지만, 워크로드를 제어하고 긴 대기 시간을 피하기 위해 작업을 분산시키는 것이 좋습니다. 작업은 지정된 시간이나 간격에 따라 실행되도록 스케줄된 WebSphere Commerce 명령입니다. 작업의 시간을 지정하려면 AddJob 명령의 명령 시작 및 간격 매개변수를 사용하십시오. AddJob 명령에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 "AddJob 명령"을 참조하십시오.

다음 경우에 스케줄러를 실행해야 합니다.

- IBM WebSphere Commerce Payments가 실행 중입니다.
- 경매가 실행 중입니다.
- 규칙 서비스를 프로덕션 환경으로 전달 중입니다.
- 사이트 운영자는 시스템을 변경 중이며 클러스터의 각 인스턴스를 다시 시작하지 않고 모든 복제본으로 변경사항을 전달하려고 합니다..
- ATP(Available To Promise) 기능이 사용됩니다.
- 사이트가 거래 메시지를 보냅니다.

이러한 상황 중 어느 것에도 해당되지 않으면, 사이트 운영자가 스케줄러를 사용하지 않기로 선택할 수 있습니다.

## 로그 작성 서비스

WebSphere Commerce 서버에서 로그 작성 메시지의 목적은 운영자에게 WebSphere Commerce 응용프로그램에 예상하지 못한 오류 또는 비정상 조건이 발생했음을 알리기 위한 것입니다. 메시지 로그 및 추적 정보는 사이트 운영자가 문제의 원인을 판별하는 데 도움이 되는 중요한 진단 도구입니다. 추적은 문제점 판별 메커니즘입니다. 추적은 개발자가 개발 단계 중에 코드를 디버깅하는 데 도움이 되고 기술 지원부에서 고객 문제를 해결하는 데 도움을 줍니다. WebSphere Commerce의 로그 작성이 WebSphere Application Server의 로그 작성 기능을 사용하므로 로그 분석기를 사용할 수 있습니다. 로그 분석기는 로그 파일 보기 및 분석을 용이하게 하는 그래픽 유틸리티입니다. 로그 분석기에 관한 추가 정보는 WebSphere Application Server InfoCenter를 참조하십시오.



WebSphere Commerce는 로그 작성 기능을 제공합니다. 기존 고객을 위해 ECTrace 및 ECTrace를 계속 지원합니다. 새로 구현하는 경우 WebSphere Application Server에서 제공하고 WebSphere Commerce 응용프로그램에서 사용 가능한 JRAS를 사용할 것을 권장합니다.

WebSphere Commerce 로그 API는 다음과 같습니다.

- ECTrace는 데이터 플로우를 추적합니다. 추적 항목은 디버깅을 위해 파일에 캡처됩니다.
- ECTrace는 진단 메시지를 로깅합니다. 메시지는 로케일에 따라 해당 로케일로 작성되며 파일 시스템의 로그 파일에 저장됩니다. 오류 전달 정보 기능이 사용 가능한 경우, 기술 지원부에 경고 알림 메시지가 전달됩니다. 진단 로그는 문제점 판별을 위해 사용됩니다. 기본 로그 파일 이름은 activity.log입니다.
- JRAS는 메시지 로그 작성 및 진단 추적 원형을 제공하는 독립형 로그 작성 툴킷입니다.

기본 출력 파일의 위치는 `WAS_installdir/logs/WC_<instance name>/`입니다.

**400** `WAS_userdir/logs/WC_instance_name/`입니다. 기본 출력 파일은 다음과 같습니다.

- native\_stderr.log는 stderr(표준 오류) 스트림에 기록된 텍스트를 포함하고 있는 프로세스 로그입니다.
- native\_stdout.log는 stdout(표준 출력) 스트림에 기록된 텍스트를 포함하고 있는 프로세스 로그입니다.
- startServer.log는 서버를 시작할 때의 로그입니다.
- stopServer.log는 서버를 중지할 때의 로그입니다.
- SystemErr.log는 서버 실행 중의 모든 시스템 오류를 로그합니다.
- SystemOut.log는 서버 실행 중의 시스템 출력 파일을 로그합니다.
- activity.log는 연속적인 활동을 로그합니다. 이 로그는 `WAS_installdir/logs` 디렉토리에 있습니다.

WebSphere Application Server 로그 API에 관한 정보는 WebSphere Application Server InfoCenter의 로그 작성 및 추적에 대한 절을 참조하십시오.

## ECTrace

진단 로그는 문제점 판별을 위해 사용됩니다. ECTraceLog는 진단 메시지를 기록하며 ECTrace는 한글화됩니다. ECTrace는 다음 카테고리로 나뉩니다.

- 시스템 메시지
- 사용자 메시지

**시스템 및 사용자 메시지:** 시스템 메시지는 로그에 표시되며 문제점의 디버깅에 도움을 주기 위한 것입니다. 시스템 메시지는 사이트 운영자에게 진단 정보를 제공합니다. 이들 메시지는 시스템 오작동의 결과로서 생기거나, 기타 중대한 이벤트를 표시합니다.

시스템 메시지에는 제품 특정 메시지 ID가 지정됩니다.

CMNnnnns

여기서

nnnn: 관련된 항목을 식별하는 데 사용하는 키 번호

s: 메시지의 심각도.

사용자 메시지는 흔히 브라우저에 표시되고 사이트를 방문하는 고객을 위한 것입니다. 사용자 메시지는 문제점에 대한 정보를 제공합니다. 예를 들면 지정된 매개변수가 올바른지 않음을 표시하여 고객에게 요청을 다시 제출할 때 수정할 값을 알려줍니다. 사이트 운영자는 메시지 ID를 사용하여 해당 메시지와 연관된 정보를 더 자세히 조사할 수 있습니다. 즉, 고객은 문제점 해결을 위해 메시지를 지원 담당자에게 보고할 수 있습니다.

**로그 작성 레벨:** WebSphere Commerce 로그 시스템에는 다섯 가지의 로그 레벨, 즉 심각도가 있습니다. 시스템 메시지의 심각도에는 오류, 상태, 경고, 정보 및 디버그가 있습니다.

- 오류 메시지는 기본으로 항상 기록됩니다. 오류 메시지는 시스템 오작동의 원인이 될 수 있는 오류 조건을 표시합니다. 오류 메시지는 전자 우편, WebSphere MQ 메시지 또는 다른 형식의 알림으로 메시지 전달에 등록된 사이트 운영자에게 전송될 수 있습니다.
- 상태 메시지는 WebSphere Commerce의 특정 상태를 나타냅니다. 예를 들어, 각 구성요소는 로드되거나 메시지가 특정 상태로 바뀌었을 때 메시지를 보냅니다. 상태 메시지는 기술 지원부에서 구성요소 및 응용프로그램의 상태를 이해하도록 도와줍니다.
- 경고 메시지는 잠재적인 문제점을 알려줍니다.
- 정보 메시지는 WebSphere Commerce 시스템 내에서 이벤트가 발생한 후에 표시됩니다. 정보 메시지는 시스템 상태의 변경을 트리거하는 이벤트와 관련됩니다. 예를 들어, 주문이 제출되었을 때 정보 메시지가 표시됩니다.
- 디버그 메시지는 추적 구성요소와 기능이 비슷합니다. 그러나 사이트 운영자가 디버그 메시지를 사용하면 기술 지원부에 문의하지 않고도 문제점을 스스로 조사할 수 있습니다.

WebSphere Commerce에서 WebSphere Application Server 로그 작성 기능을 사용하고 WebSphere Application Server는 세 가지 유형의 로그 레벨만을 가지므로 WebSphere Commerce 로그 작성 레벨은 다음과 같이 WebSphere Application Server에 맵핑됩니다.

표 3.

WebSphere Commerce 의 로그 작성 레벨	WebSphere Application Server의 로그 작성 레벨
ERROR/ERR	TYPE_ERROR/TYPE_ERR
INFORMATION/INFO	TYPE_INFORMATION/TYPE_INFO
DEBUG	TYPE_INFORMATION/TYPE_INFO
WARNING/WARN	TYPE_WARNING/TYPE_WARN
STATUS	TYPE_INFORMATION/TYPE_INFO

WebSphere Application GUI는 로그 작성 레벨 또는 심각도 유형의 사용을 제어하는 데 사용됩니다. 사이트 운영자는 WebSphere Application Server 관리 콘솔에 기록할 로그 작성 심각도를 지정할 수 있습니다. 로그 작성 레벨 사용에 대한 추가 정보는 WebSphere Application Server InfoCenter의 정보를 참조하십시오.

### ECTrace

추적은 문제점 판별에 사용됩니다. 추적은 개발자가 개발 단계 중에 코드를 디버깅하는 데 도움이 되고 기술 지원부에서 고객 문제를 해결하는 데 도움을 줍니다

추적 데이터는 향후 참조를 위해 추적 파일에 지속됩니다. 데이터 구조는 클래스 이름, 메소드 이름, 텍스트 메시지와 같은 컨텍스트 정보로 구성됩니다. 여러 데이터 구조가 한 소프트웨어 응용프로그램의 데이터 플로우를 설명합니다. 개발자는 데이터 구조 순서를 분석함으로써 실행된 경로를 이해할 수 있습니다. 이는 오작동의 원인을 판별하는 데 도움이 됩니다.

### JRas

JRas는 메시지 로그 작성 및 진단 추적 원본을 제공하는 여러 Java 패키지로 이루어져 있습니다. 이 원본은 어떠한 특정 제품 또는 플랫폼에도 구속되지 않습니다. JRas는 기본적으로 다음의 여러 구성요소로 구성됩니다.

- **로그 작성 프로그램(Logger):** 로그 작성 프로그램은 사용자 코드가 상호작용하는 기본 오브젝트입니다. 메시지 로그 작성 프로그램에는 메시지 로그 작성 프로그램과 추적 로그 작성 프로그램의 두 가지 유형이 있습니다. 메시지 로그 작성 프로그램은 단지 메시지 레코드를 작성하며 추적 로그 작성 프로그램이 추적 레코드를 작성합니다. 로그 작성 프로그램은 추가 처리를 위해 이벤트를 전송하는 하나 이상의 핸들러를 포함합니다.
- **핸들러:** 핸들러는 로그 작성 프로그램으로부터 이벤트를 수신하고, 출력 장치 또는 이벤트 소비자에게 추상 개념을 제공합니다. 이벤트를 파일에 기록하는 방법을 알고 있는 파일 핸들러가 한 예입니다.
- **포맷터:** 핸들러는 특정 유형의 이벤트를 포맷하는 법을 알고 있는 포맷터와 함께 구성됩니다.
- **이벤트 유형 -** 메시지 및 추적은 고유한 사전 정의된 이벤트 유형을 가집니다.

- 이벤트 클래스 - 독립형 JRas 로그 작성 툴킷은 메시지 및 추적 클래스를 모두 정의합니다.

주: JRas 로그 작성 툴킷에 대한 추가 정보는 WebSphere Application Server 정보 센터의 WebSphere Application Server JRas 문서를 참조하십시오.

**WebSphere JRas 확장자:** WebSphere Application Server 런타임에 통합하거나 J2EE 환경에서 사용하기 위해 WebSphere는 확장자 클래스 세트를 제공합니다. WebSphere Commerce에서의 로그 작성은 WebSphere Application Server 로그 작성 기능을 사용하며 이들 확장자 클래스는 다른 WebSphere 제품에서 작성되는 메시지와 추적에 대해 더 나은 상관 관계를 제공합니다. 이러한 확장자 클래스 컬렉션을 WebSphere JRas 확장자라고 합니다. WebSphere JRas 확장자는 적절한 로그 작성 프로그램 구현 클래스를 제공합니다. 이들 메시지 및 추적 로그 작성 프로그램 클래스의 인스턴스는 `com.ibm.websphere.ras` 패키지에 있는 WebSphere Manager 클래스가 직접 독점적으로 획득합니다. Payments 및 JCA 메시지 전달 프레임워크와 같은 기타 구성요소도 WebSphere JRas 확장자를 사용합니다. WebSphere Commerce 및 Payments 추적 구성요소와 연관된 WebSphere JRas 확장자 목록은 70 페이지의 『추적 구성요소 사용』을 참조하십시오.

주: WebSphere Commerce는 WebSphere JRas 확장 API를 호출하는 ECTMessage 및 ECTTrace에 대한 래퍼를 제공합니다. 그러나 사이트 운영자가 직접 JRas API를 호출하는 것은 권장하지 않습니다.

## WebSphere Commerce Payments 문제점 판별

WebSphere Commerce Payments를 사용하기 시작할 때, 실제로 응용프로그램을 사용하는 동안 발생하는 문제점을 진단할 필요가 있습니다. WebSphere 로그 분석기는 조치 순서를 판별하기 위해 오류를 분석하고 로그를 추적할 수 있는 인터페이스를 제공합니다. 다음 메시지를 제외한 모든 Payment 오류 및 추적 로그가 로그 분석기를 통해 사용할 수 있습니다.

- 타사 카세트
- 독립형 Payments UI
- SampleCheckout
- Cashier

## 규칙 서비스

WebSphere Commerce는 규칙 서비스를 사용하여 Advisor Rule Server와 상호작용합니다. 규칙 서비스는 두 응용프로그램간의 통신을 손쉽게 하기 위한 인터페이스 역할을 합니다. 또한 규칙 서비스는 전체 WebSphere Commerce를 중지할 필요 없이 사이트의 규칙 기반 부분을 편리하게 갱신할 수 있도록 합니다. 규칙 서비스는 캠페인과 1 대 1 관계로 존재합니다. WebSphere Commerce 액셀러레이터를 사용하여 공개된

각 캠페인에는 해당 규칙 서비스가 있습니다. 이 설계는 각 마케팅 서비스 및 캠페인을 다른 모든 서비스와 무관하게 시작, 중지 및 최신 정보로 고칠 수 있는 유연성을 제공합니다. WebSphere Commerce 액셀러레이터를 사용하여 고객 프로파일, 캠페인 또는 규칙 서비스를 갱신할 때마다 파일을 프로덕션 서버의 해당 위치로 전송하여 규칙 서비스를 최신 정보로 고쳐야 합니다. 각 응용프로그램 복제 또는 JVM(Java Virtual Machine)에 규칙 서버의 로컬 인스턴스가 있습니다. 응용프로그램 복제 경계를 넘어서 통신할 필요는 극히 적지만, 필요한 경우가 하나 있습니다. 규칙 서비스 상태 갱신이 그것입니다. 각 응용프로그램 복제로부터 상태 보고서를 수신하기 위해 시스템은 스케줄러를 사용하여 규칙 서비스 상태 갱신 요청을 브로드캐스트합니다. 결과가 수집되고 사용자가 "상태 보기" 버튼을 눌렀을 때 표시됩니다.

### **Advisor Rule Server**

WebSphere Commerce는 Advisor Rule Server를 사용하여 규칙을 처리하고 광고 및 제안 판매 기술로 이루어진 개인 마케팅 콘텐츠를 제공합니다. 이 서버는 WebSphere Commerce Server에 통합되어 있습니다. 규칙 서버는 현재 구매 환경에 근거한 정보를 전달하는 WebSphere Commerce Server에 의해 호출됩니다. 규칙 서버는 이 정보를 판매자 또는 마케팅 담당자가 작성한 규칙 콜렉션과 대조하여 처리하며 특정 상황에 대한 출력을 컴파일합니다. 이 출력은 WebSphere Commerce 액셀러레이터를 사용하여 정의된 기준에 근거합니다. 규칙 서버는 라이선스 정보 소책자에 포함되어 있는 일정한 한계가 적용됩니다. 이 한계를 초과하려면 HNC Inc.의 별도의 라이선스가 필요합니다.

## **LDAP -- Lightweight Directory Access Protocol**

LDAP는 디렉토리 서비스에 액세스하는데 사용되는 클라이언트-서버 프로토콜입니다. 본래 X.500의 실행 단계 준비로서 사용되었지만, 독자적으로 그리고 다른 종류의 Directory Server와 함께 사용될 수도 있습니다. LDAP는 다양한 응용프로그램 간에 정보 공유를 지원하기 위한 중앙 저장소로서 사용될 수 있습니다.

LDAP 정보 모델은 예를 들어 사용자와 같이 어떤 오브젝트에 관한 정보가 들어 있는 항목에 기초합니다. 항목은 유형과 하나 또는 그 이상의 값이 있는 속성들로 구성됩니다. 각 속성에는 속성에서 허용되는 값의 종류와 디렉토리 조작 중에 그 값이 움직이는 방법을 결정하는 구문이 있습니다. IA5(ASCII) 문자열, JPEG 사진, u-law 인코딩 사운드, URL 및 PGP 키 등이 속성 구문의 예입니다. 일반적으로, 항목은 그 이름이나 인식 이름으로 고유하게 식별됩니다.

WebSphere Commerce에서 등록된 사용자는 인식 이름과 같은 포맷을 갖는 문자열로 된 고유 식별자를 가집니다. WebSphere Commerce 데이터베이스를 구성원 저장소로 사용하는 경우, DN의 형식은 'uid=logonIDvalue' 다음에 사용자가 속하는 상위 조직 엔티티의 DN이 따릅니다. 따라서, 사용자의 로그인 ID가 'john'이고 IBM 조직 내의 SWG 조직 단위에 속하는 경우, 사용자의 고유 식별자는 'uid=john, ou=SWG, o=IBM, o=Root Organization'이 됩니다. 디렉토리 서버를 구성원 저장소로 사용하는 경우, 등

록된 사용자의 고유 식별자는 디렉토리 서버에 있는 사용자의 DN이 됩니다. 이 경우, WebSphere Commerce가 루트 조직에 없을 수 있는 디렉토리 서버의 기존 사용자 항목을 지원하기 때문에 'o=Root Organization' 부분이 사용자 DN의 일부가 아닐 수 있습니다.

WebSphere Commerce는 또한 디렉토리 서버에 조직 엔티티를 저장할 수 있도록 지원하고 모든 조직 엔티티는 DN과 같은 형식의 고유 식별자를 가집니다. WebSphere Commerce 데이터베이스를 구성원 저장소로 사용하는 경우, 고유 식별자의 형식은 'ou=organizationalEntityName'이고 뒤에 상위 조직 엔티티의 DN이 따릅니다. 따라서 마케팅 조직 단위가 IBM 조직의 SWG 조직 단위 내에 있는 경우, 고유 식별자는 'ou=Marketing,ou=SWG,o=IBM,o=Root Organization'이 됩니다. 디렉토리 서버를 구성원 저장소로 사용하는 경우, 조직 엔티티의 고유 식별자는 디렉토리 서버에 있는 DN이 되고, 이 경우, 'o=Root Organization' 부분은 DN의 일부가 될 수 없습니다. 이는 WebSphere Commerce가 루트 조직에 없을 수 있는 디렉토리 서버의 기존 조직 항목을 지원하기 때문입니다. 또한, DN에서 'o' 또는 'ou'의 사용 여부는 조직 엔티티가 조직 또는 조직 단위인지에 따라 다릅니다.

디렉토리 서버를 구성원 저장소로 사용하는 경우, 사용자 및 조직 엔티티를 디렉토리 서버에 저장할 수 있고, RDN 속성으로 사용할 LDAP 속성 및 RDN 속성에 값을 제공하는 WebSphere Commerce 속성을 구성할 수 있습니다. 일반적인 DN은 다음과 같습니다.

```
uid=jsmith, l=Toronto, st=Ontario, c=CA
```

여기서

**uid** 사용자 ID. 위 예제에서 uid 속성을 RDN 속성이라 합니다. 이 속성은 DN이 *l=Toronto, st=Ontario, c=CA*.인 상위 항목 아래에서 LDAP 항목을 고유하게 식별합니다. WebSphere Commerce에서 인증 요구사항이 로그인 ID 및 암호인 경우, WebSphere Commerce는 주어진 검색 공간에서 RDN = *logonID*인 노드에 대해 LDAP 서버를 검색합니다. X.509 인증이 챌린지 유형일 경우, WebSphere Commerce는 인증의 제목 이름을 RDN 속성값으로 사용합니다.

- l**     사용자의 구/군/시
- st**    사용자의 시도
- c**     사용자의 국가 또는 지역

다음 IBM(R) 레드북에는 LDAP에 대한 추가 정보가 들어 있습니다.

- *LDAP Implementation Cookbook*
- *Understanding LDAP*

레드북은 다음 사이트에서 볼 수 있습니다. <http://www.redbooks.ibm.com>.



WebSphere Commerce Server와 LDAP 서버 간에 SSL을 사용하려면 LDAP 서버가 제공하는 지시사항을 참조하여 설치하고(예를 들어, 키 링 파일 작성) *instancename.xml* 파일에서 *JNDIEnvPropNameX* 및 *JNDIEnvPropValueX* 속성을 사용하여 WebSphere Commerce Server에 필요한 환경 특성을 지정하십시오.

## LDAP를 이용한 인증

사용자 프로파일 정보는 WebSphere Commerce 데이터베이스 또는 디렉토리 서버에 저장할 수 있습니다. 인증 옵션은 다음과 같습니다.

- LDAP

사용자가 DN 또는 RDN 값과 암호로 로그인할 수 있습니다. RDN을 사용하는 경우, 사용자는 *ldapentry.xml* 파일에 구성된 검색 기준을 사용하여 검색됩니다. 사용자는 LDAP 서버에 대해 인증됩니다. LDAP 서버의 사용자 정보는 런타임 조작 시 사용할 수 있도록 WebSphere Commerce 데이터베이스로 복제됩니다.

- 데이터베이스

사용자는 사용자가 제공한 로그인 ID 및 암호를 사용하여 WebSphere Commerce 데이터베이스에 대해 인증됩니다.

- 기타

타사의 인터페이스인 WebSphere Commerce 데이터베이스의 상점 프로파일 데이터를 사용하여 인증합니다.

인증 모드를 지정하려면 구성 관리자의 구성원 서브시스템 페이지를 사용하십시오. X.509 인증이 웹 서버 사용자 인증에 사용되는 경우 X.509 인증을 LDAP 또는 데이터베이스 인증에 사용할 수 있습니다. 이 경우, 인증 모드 설정에 따라 프로파일 데이터 저장 위치가 LDAP 또는 WebSphere Commerce 데이터베이스로 결정됩니다.

## LDAP 레지스트리

WebSphere Commerce에서 LDAP 또는 WebSphere Commerce 데이터베이스를 구성원 저장소로 사용할 수 있습니다. 사용자 및 조직 엔티티는 LDAP 서버에 저장할 수 있습니다. 현재 구성원 그룹은 WebSphere Commerce 데이터베이스에만 저장할 수 있습니다. LDAP를 레지스트리로 사용하는 경우, 이는 *instancename.xml* 파일의 *AuthenticationMode=LDAP* 및 *ProfileDataStorage=LDAP* 설정에 해당합니다.

데이터는 LDAP 서버와 WebSphere Commerce 데이터베이스 간에 복제됩니다. *ldapentry.xml* 파일을 사용하여 복제할 데이터를 구성할 수 있습니다.

다음 테이블은 WebSphere Commerce에 제공된 LDAP 서비스 레벨에 대해 설명합니다.

표 4. 사용자 등록

	레지스트리로서의 LDAP	LDAP 지원 안됨
WebSphere Commerce를 통해 새 사용자 등록 또는 등록된 사용자의 정보를 갱신합니다.	사용자 정보가 WebSphere Commerce 데이터베이스 및 LDAP 서버에 작성되거나 갱신됩니다.	사용자 정보가 WebSphere Commerce 데이터베이스에 작성되거나 갱신됩니다.
LDAP 사용자는 다른 LDAP 응용프로그램으로부터 정보를 등록하고 갱신합니다.	사용자 정보는 사용자가 WebSphere Commerce Server에 로그인하거나, 사용자 정보가 WebSphere Commerce Server 내의 로직에 필요한 경우에만 WebSphere Commerce 데이터베이스로 복제됩니다.	없음
LDAP 서버 가용성	LDAP 서버는 항상 사용 가능해야 합니다.	없음
LDAP 연결 실패	오류 페이지가 사용자에게 표시됩니다.  오류 메시지가 시스템 로그 파일에 로그됩니다.	없음

표 5. 조직 엔티티 등록

	레지스트리로서의 LDAP	LDAP 지원 안됨
WebSphere Commerce를 통해 새 조직 엔티티 등록 또는 조직 엔티티 정보를 갱신합니다.	조직 엔티티 정보가 WebSphere Commerce 데이터베이스 및 LDAP 서버에 작성되거나 갱신됩니다.	조직 엔티티 정보가 WebSphere Commerce 데이터베이스에 작성되거나 갱신됩니다.
다른 LDAP 응용프로그램으로부터 조직 엔티티 정보가 작성되거나 갱신됩니다.	조직 엔티티는 조직 엔티티 정보가 WebSphere Commerce Server 내의 로직에 필요한 경우에만 WebSphere Commerce 데이터베이스에 작성되거나 갱신됩니다.	적용되지 않음.

등록이 승인 보류 중인 사용자도 디렉토리 서버에 저장됩니다. 컨텍스트 종속 속성은 기본으로 디렉토리 서버에 저장되지 않습니다.

사용자 등록 주소만이 LDAP에 유지됩니다. LDAP 사용자 주소록에서 친구, 친척 또는 업무 관계자 주소는 지속되지 않습니다.

주: LDAP 서버를 구성원 저장소로 선택한 경우, 로더 패키지를 사용하여 사용자 및 조직 엔티티 정보를 WebSphere Commerce 데이터베이스로 대량 로드해서는 안됩니다. WebSphere Commerce 5.5에서는 데이터를 데이터베이스로 대량 로드하는 비즈니스 모델이 도입되었습니다. 이 비즈니스 모델에 대한 추가 정보는 *WebSphere Commerce 상점 개발 안내서*를 참조하십시오.



## LDAP 구성 매개변수

사용할 LDAP 시스템을 설치한 후, 구성 관리자의 구성원 서브시스템 페이지에서 해당 필드를 완성하거나, LDAP 지원을 수동으로 구성하는 경우에는 *instancename.xml* 파일을 완성해야 합니다. 이 파일은 *instancename.xml* 파일의 MemberSubSystem 섹션에 있는 매개변수에 대해 설명합니다. WebSphere Commerce의 각 인스턴스에는 *instancename.xml* 파일이 포함됩니다. 충돌을 피하려면 LDAP 구성 정보가 각 인스턴스에 대해 같은지 확인하십시오.

주: 다음은 예제 코드이고, 값은 이 예제에서만 사용하기 위해 가정된 값입니다. 사용자 파일에서는 올바른 매개변수 값을 제공해야 합니다.

```
<MemberSubSystem name="Member SubSystem"
    ProfileDataStorage="LDAP"
    AuthenticationMode="LDAP">
<Directory LdapPort="636"
    LdapType="SECUREWAY"
    LdapAuthenticationMode="SIMPLE"
    EntryFileName="d:/WebSphere/CommerceServer55/
xml/ldap/ldapentry.xml"
    LdapAdminPW="EaDPFd9VAf0="
    LdapVersion="3"
    LdapHost="basswood.torolab.ibm.com"
    SingleSignOn="0"
    LdapAdminDN="cn=root"
    display="false"
    MigrateUsersFromWCdb="ON"
    JNDIEnvPropName1="java.naming.referral"
    JNDIEnvPropValue1="ignore"
    JNDIEnvPropName2="java.naming.security.
protocol"
    JNDIEnvPropValue2="ssl"
    ...
    JNDIEnvPropNameN="java.naming.language"
    JNDIEnvPropValueN="en-US"
```

```
LdapTimeOut="0" />
</MemberSubSystem>
```

### 매개변수 설명

#### **ProfileDataStorage**

사용자 및 조직 엔티티에 대한 프로파일 데이터가 저장되는 위치를 결정합니다. 값이 LDAP인 경우에도 일부 프로파일 데이터는 WebSphere Commerce 데이터베이스로 복제됩니다.

#### **AuthenticationMode**

사용자가 인증되는 시스템을 결정합니다. 유효한 값은 다음과 같습니다.

- LDAP(기본)
- DB
- 기타(타사 메커니즘에 대해 사용)

#### **LdapAdminDN**

LDAP에서 조작을 수행하는 데 사용되는 운영자의 DN. 이 DN은 LDAP에 설정되어 있어야 하며 LDAP에 대한 사용권한이 있어야 합니다.

#### **LdapAuthenticationMode**

사용 중인 디렉토리 서버에 따라 달라집니다.

Simple 및 None이 모두 IBM SecureWay® 및 Netscape iPlanet에 올바른 값입니다.

- Simple(기본값)
- None

#### **LdapTimeOut**

LDAP 검색이 시간종료될 때까지의 시간(초 단위). 시간종료 값은 LDAP 서버에서도 지정할 수 있습니다. 두 값 중 작은 값이 우선권이 있습니다.

LdapTimeOut의 값이 0일 경우, 검색 시간종료는 LDAP 서버 구성에 의해서만 결정됩니다. 기본값은 0입니다.

#### **LdapVersion**

LDAP 프로토콜 버전 번호. 올바른 값은 3뿐입니다.

#### **EntryFileName**

LDAP 속성에 맵핑되는 WebSphere Commerce 속성에 대한 정보가 들어 있는 XML 파일.

기본값은 다음과 같습니다.

Solaris
Linux
Windows

```
WC installdir/
xml/ldap/ldapentry.xml
```

`WC_installdir/xml/ldap/ldapentry.xml`

### LdapPort

LDAP 서버에서 사용되는 포트. 기본값은 389입니다.

### SingleSignOn

'0' 값은 단일 사인온이 사용 안함으로 설정됨을 의미하고, '1'은 사용하도록 설정됨을 의미합니다.

### LdapAdminPW

기본값 없음 - 공백

관리자의 암호화된 암호. XML을 직접 수정한 경우, `wcs_encrypt.bat` 또는 `wcs_encrypt.sh`를 사용하여 작성됩니다.

### LdapHost

LDAP 서버의 호스트 이름. 기본값은 현재 호스트 이름입니다.

### MigrateUsersFromWCdb

상점 공개 프로세스에 의해 작성된 부트스트랩 사용자 및 모든 조직을 확실하게 LDAP에 푸쉬하려면 이 옵션을 사용해야 합니다. 그렇지 않은 경우에는 사용 안함으로 설정해야 합니다. 기본값은 사용 안함입니다.

### LdapType

사용 중인 LDAP 디렉토리 서버.

예:

▶ Windows ▶ AIX ▶ Solaris ▶ Linux IBM SecureWay V5.1(기본값)

▶ 400

- OS400 - 디렉토리 서비스
- IBM SecureWay V3.2.2

### JNDIEnvPropNameX 및 JNDIEnvPropValueX

이러한 속성을 사용하여 WebSphere Commerce Server가 설정할 JNDI 환경 특성을 지정하십시오. 이러한 속성은 `instancename.xml` 파일에서 직접 지정해야 합니다.

WebSphere Commerce 5.1에서 다음 사항이 변경되었습니다.

- `MappingFileName`이 `EntryFileName`으로 변경되었습니다.
- `ldapmap.xml`이 `ldapentry.xml`로 변경되었습니다.
- `LdapPersonRDN`, `LdapPersonSearchRoot`, `LdapPersonDefaultBase`, `LdapPersonOCS`, `LdapOrgOCS` 및 `LdapOrgUnitOCS`은 이제 `ldapentry.xml` 파일의 일부이며, 이름이 다음과 같이 변경되었습니다.
  - `LdapPersonRDN`은 `rdnName`로 변경

- LdapPersonSearchRoot는 searchBase로 변경
- LdapPersonDefaultBase는 defaultBase로 변경
- LdapPersonOCS는 objClass로 변경
- LdapOrgOCS는 objClass로 변경
- LdapOrgUnitOCS는 objClass로 변경
- LDAP 프로토콜 버전 3만 지원됩니다.

ldapentry.xml 및 ldapmap.dtd 파일에 대한 정보는 175 페이지의 부록 B 『LDAP 파일』을 참조하십시오.

### LDAP 설계 특징

다음 정보는 WebSphere Commerce에서 LDAP 구현의 몇몇 설계 특징을 강조합니다.

- 여러 WebSphere Commerce 속성을 분리자 문자로 지정하여 하나의 LDAP 속성으로 복제할 수 있습니다.
- 여러 LDAP 속성에서 하나의 WebSphere Commerce 속성으로 복제할 수 있습니다. 이를 수행하려면 ldapentry.xml 파일에 있는 여러 <map> 섹션을 사용하십시오. 그러나 겹쳐쓰기가 발생할 수 있음을 유의하십시오.
- 하나의 WebSphere Commerce 속성을 여러 LDAP 속성으로 복제할 수 있습니다.
- 하나의 LDAP 속성을 여러 WebSphere Commerce 속성으로 복제할 수 있습니다.
- 사용자의 기본 주소만이 LDAP에 존속됩니다. 기본 등록 주소는 selfaddress=1 및 logonid=nickname을 사용하여 사용자 주소록의 주소로 정의됩니다. logonID는 USEREG 테이블의 logonid이고 nickname은 ADDRESS 테이블의 NICKNAME입니다. 다른 주소는 LDAP에 저장되지 않는 사용자 주소록입니다.
- WebSphere Commerce에서 등록된 사용자만 LDAP에서 지속되고 게스트 사용자는 지속되지 않습니다.
- WebSphere Commerce에서는 LDAP에서 사용자 항목을 검색할 때 여러 개의 검색 기준을 지정할 수 있습니다. 그러나, 검색 기준에 맞는 복수 사용자 항목이 발견되면 오류가 작성됩니다.

다음 테이블은 WebSphere Commerce에서 LDAP 지원의 몇몇 제한 사항에 대해 설명합니다.

제한 사항	설명
LDAP에 여러 값 속성이 없음	WebSphere Commerce는 여러 값을 갖는 속성을 저장하거나 검색하지 않습니다. 그러나, LDAP에서 여러 값을 갖는 속성은 첫 번째 값만 검색됩니다. 갱신 중에 새 값이 이미 LDAP에 있으면 갱신이 수행되지 않습니다. 그렇지 않은 경우, 여러 값을 갖는 LDAP 속성 중 첫 번째 값만 갱신됩니다.

제한 사항	설명
LDAP 참조 횟수 및 참조	WebSphere Commerce는 LDAP 참조 횟수 및 참조를 지원하지 않습니다.
여러 구성요소 RDN	LDAP 속성을 조합한 것이 아닌 하나의 LDAP 속성만을 RDN 속성으로 지정할 수 있습니다.

## 스테이징 서버

대부분의 온라인 상점은 하루 24시간, 1년 365일간 운영되므로 유지보수나 시스템 변경 테스트를 수행하기 어렵습니다. WebSphere Commerce 스테이징 서버를 사용하면 사이트 운영자가 고객에게 영향을 미치지 않고 갱신사항을 테스트하기 위해 프로덕션 데이터베이스를 스테이징 데이터베이스로 복사할 수 있습니다. 이는 상품 카탈로그에 대한 갱신사항을 테스트하는 데 유용하지만 새 쇼핑 프로세서 명령을 테스트하는 데도 중요합니다.

스테이징 서버에는 다음의 구성요소가 있습니다.

### WebSphere Commerce 인스턴스

데이터를 테스트하고 수정합니다.

### 데이터베이스 스키마 스크립트

스테이징 데이터베이스에 대한 스테이징 테이블 및 트리거를 작성합니다. 스테이징 데이터베이스는 프로덕션 데이터베이스와 동일한 스키마 및 테이블을 포함하며 스테이징 데이터베이스에서 작성된 변경을 로그하기 위한 트리거 세트도 포함하고 있습니다. 스테이징 데이터베이스 스키마 스크립트는 트리거를 데이터베이스에 추가합니다.

변경사항은 데이터베이스 트리거를 사용하여 STAGLOG 테이블(스테이징 테이블)에 로그됩니다. 스테이징 데이터베이스에서 데이터베이스 테이블 레코드를 변경할 때마다 STAGLOG 테이블에 변경사항이 기록됩니다.

### 스테이징 복사 유틸리티

관리자가 데이터를 프로덕션 데이터베이스에서 스테이징 데이터베이스로 복사할 수 있도록 합니다. 데이터를 사이트 관련 테이블, 판매자 관련 테이블 또는 개별 테이블로 복사할 수 있습니다.

### 스테이징 전달 유틸리티

관리자가 변경사항을 스테이징 데이터베이스에서 프로덕션 데이터베이스로 전달할 수 있습니다. STAGLOG 테이블에 있는 정보는 스테이징 데이터베이스에 있는 레코드 중에 어느 레코드를 프로덕션 데이터베이스에서 삽입, 갱신 또는 삭제해야 하는지를 식별합니다. 그런 다음, 식별된 레코드가 프로덕션 데이터베이스에서 갱신됩니다. 처리된 레코드는 STAGLOG 테이블의 STGPROCESSED 열에서 1로 표시됩니다.

## 스태이징 확인 유틸리티

관리자가 스테이징 서버 및 프로덕션 서버의 두 테이블 간에 고유 색인 키 충돌 가능성이 있는지를 확인할 수 있게 합니다.

## 스태이징 서버 제한 사항

스태이징 서버를 사용하기 전에 다음 제한 사항을 알고 있어야 합니다.

- 구매자 조직 자체 관리 기능을 사용하여 스테이징 서버를 사용할 수 없습니다.
- 모든 스테이징 테이블의 member\_id 열(MEMBER, MBRREL, MBRROLE 및 MBRATTRVAL 제외)은 사용자가 아닌 조직 또는 구성원 그룹이어야 합니다.
- 모든 사이트 테이블에서 member\_id는 -2001 또는 0이어야 합니다. 사이트 및 판매자 데이터가 들어 있는 모든 테이블에서 사이트 데이터와 관련된 행의 member\_id는 0 또는 -2001이어야 합니다.
- 프로덕션 시스템에 RFQ 특징이 있는 경우 스테이징 복사 명령을 사용할 수 없습니다. 스테이징 복사 명령을 사용할 수 없으므로 푸쉬 모드를 사용해야 합니다. 프로덕션 사이트를 실행하기 전에 스테이징 서버를 작성하고 스테이징 데이터베이스를 설정하십시오. 스테이징 서버에서 데이터를 전개 및 테스트한 후 스테이징 전달 명령을 사용하여 프로덕션 서버에 푸쉬하십시오.
- 스테이징 서버에서는 RFQ를 작성하거나 갱신할 수 없습니다.

**키 분할:** 재판매자 MarketPlace에서 IBM 및 사이트 운영자는 스테이징 서버의 데이터를 갱신하고 재판매자는 프로덕션 서버의 데이터를 갱신합니다. 이로 인해 1차 키 충돌이 발생할 수 있습니다.

WebSphere Commerce는 키 관리자를 사용하여 테이블의 1차 키를 작성하고 키 범위는 KEYS 테이블에 정의됩니다. 프로덕션 서버와 스테이징 서버가 같은 키 범위를 사용하는 경우, 키 관리자가 프로덕션 및 스테이징 서버에서 같은 테이블에 대해 같은 1차 키 값을 할당할 수 있고, 이로 인해 1차 키 충돌이 발생합니다. 따라서, 스테이징 인스턴스 및 프로덕션 인스턴스를 작성한 후에 즉시 키 범위를 분할해야 합니다. 그러면, 스테이징 서버와 프로덕션 서버가 다른 키 범위를 사용하게 됩니다.

현재, WebSphere Commerce는 다음 SQL을 사용하여 스테이징 서버 및 프로덕션 서버에서 키 범위를 분할합니다. SQL문은 현재 키 범위를 세 개의 동일한 부분으로 나눕니다.

### 부분 1

```
update keys set upperbound=(upperbound-lowerbound)/3 + lowerbound
where tablename in (select tablename from stgmertab) or tablename
in (select tablename from stgsitetab)
```

### 부분 2

```

db2 update keys set upperbound = (upperbound-lowerbound)/3*2 + lowerbound,
lowerbound = (upperbound-lowerbound)/3 +lowerbound+1, counter =
counter+(upperbound-lowerbound)/3 + 1 where tablename in
(select tablename from stgmertab)
or tablename in (select tablename from stgsitetab)

```

부분 3

```

db2 update keys set lowerbound = (upperbound-lowerbound)/3*2 + lowerbound +1
, counter = counter +(upperbound-lowerbound)/3*2 +1 where tablename in
(select tablename from stgmertab) or tablename in (select tablename from stgsitetab)

```

부분 1인 첫 번째 SQL문은 프로덕션 서버에서 실행해야 합니다. 그러면, 프로덕션 서버가 전체 키 범위의 3분의 1을 차지합니다. 부분 2인 두 번째 SQL문은 스테이징 서버에서 실행해야 합니다. 이는 스테이징 서버가 전체 키 범위의 두 번째 3분의 1을 차지하게 됨을 의미합니다. 부분 3인 세 번째 SQL문은 나중에 잠재적인 두 번째 스테이징 서버를 위해 보관됩니다.

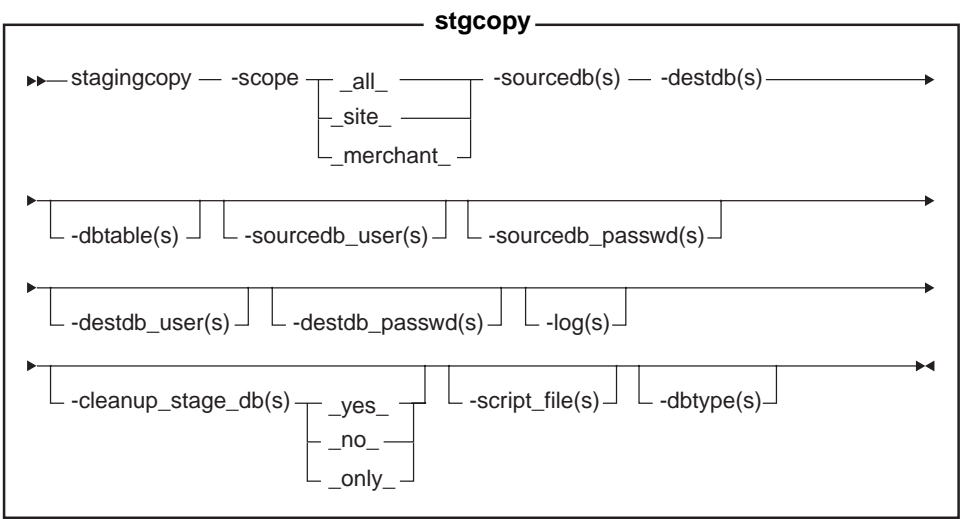
주: 위의 키 분할 SQL은 이미 인스턴스 구성 프로세스와 통합되었습니다. 이를 직접 실행할 필요가 없습니다.

### 스테이징 서버 명령

#### 스테이징 복사 유틸리티 명령(Windows 2000, AIX, Linux, iSeries 및 Solaris):

스테이징 복사 유틸리티는 프로덕션 데이터베이스에서 스테이징 데이터베이스로 데이터를 복사합니다. 프로덕션 시스템에 RFQ가 있으면 이 명령을 사용할 수 없습니다. 자세한 정보는 40 페이지의 『스테이징 서버 제한 사항』을 참조하십시오.

스테이징 복사 유틸리티를 실행하려면 스테이징 서버와 프로덕션 서버 데이터베이스에 모두 연결할 수 있는 시스템의 명령행에서 다음을 입력하십시오. 전체 명령을 한 행에 입력하십시오. 여기서는 단지 보기 편하게 하기 위해 둘 이상의 행에 나타낸 것입니다.



주:

1. **Oracle** 현재 이 유틸리티를 같은 사용자 ID로 실행 중인 경우에도 명령에 선택적 매개변수, 로그인 사용자 ID 및 암호를 포함시켜야 합니다.
2. **400** iSeries는 이제 stagingCopy.sh를 사용합니다. iSeries에서 셸 스크립트를 실행하려면 다음을 수행하십시오.
  - a. 65535외의 다른 CCSID를 가진 사용자 프로파일로 로그인하십시오.
  - b. OS/400 명령행에서 STRQSH 명령을 입력하여 QSHELL 명령창을 여십시오.
  - c. 다음과 같이 명령을 실행하십시오.

`/QIBM/ProdData/CommerceServer55/bin/stagingCopy.sh(parameters . . .)`

주: 사용자 매개변수를 항상 지정해야 합니다.

### 매개변수 값

**scope** 스테이징 서버로 복사할 범위의 레벨. 다음 중 하나를 지정하십시오.

- `_all_`  
사이트 및 모든 판매자 관련 레코드를 복사하려면 `_all_`을 입력하십시오.
- `_site_`  
사이트 관련 레코드만 복사하려면 `_site_`를 입력하십시오.
- `_merchant_`  
모든 판매자 관련 레코드만 복사하려면 `_merchant_`를 입력하십시오.

### sourcedb

프로덕션 서버의 데이터베이스 이름.

**400** 관계형 데이터베이스 디렉토리에 표시되는 대로의 프로덕션 서버상의 데이터베이스 이름.

**Oracle** `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

### destdb

스테이징 서버의 데이터베이스 이름.

**400** 관계형 데이터베이스 디렉토리에 표시되는 대로의 프로덕션 서버상의 데이터베이스 이름.

**Oracle** Use `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

### dbtable

(선택적) 복사될 특정 테이블의 이름. 레코드가 `scope` 매개변수로 지정된 범위 내에 있으면, 이 테이블의 모든 레코드가 복사됩니다. 그렇지 않으면, 레코드가 복사되지 않습니다.



### sourcedb\_user

(선택적) 소스 데이터베이스 스키마를 작성한 데이터베이스 관리자의 로그인 ID. 지정하지 않으면 현재 이 유틸리티를 호출 중인 사용자의 ID가 사용됩니다.

▶ 400 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 소스 데이터베이스 스키마와 동일합니다.

### sourcedb\_passwd

(선택적) sourcedb\_user 매개변수로 지정된 로그인 ID의 암호.

### destdb\_user

(선택적) 대상 데이터베이스 스키마를 작성한 데이터베이스 관리자의 로그인 ID.

▶ 400 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 대상 데이터베이스 스키마와 동일합니다.

주: 지정하지 않으면 유틸리티를 호출 중인 사용자의 ID가 사용됩니다.

### destdb\_passwd

(선택적) destdb\_user 매개변수로 지정된 로그인 ID의 암호. 암호를 지정하지 않으면 시스템에서 암호를 입력하라는 메시지를 표시합니다.

### log

(선택적) 스테이징 복사 유틸리티가 활동 및 오류를 기록하는 파일의 경로 및 이름. 이 매개변수를 지정하지 않으면, 다음 log 디렉토리에 stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log 로그 파일이 작성됩니다.

▶ AIX    ▶ Solaris    ▶ Linux    WC\_installdir/logs  
 ▶ Windows WC\_installdir\logs  
 ▶ 400    WC\_userdir/instances.

### cleanup\_stage\_db

(선택적) 이 매개변수를 사용하여 스테이징 복사 유틸리티를 사용하기 전에 스테이징 테이블을 지우십시오. -cleanup\_stage\_db 매개변수를 사용하여 사이트 데이터를 정리하는 경우, 계단식 삭제로 인해 판매자 데이터도 삭제될 수 있음에 주의하십시오. 사이트 데이터를 지우고 복사한 다음에 판매자 데이터를 지우고 복사해야 합니다. Yes가 기본값입니다. no를 지정하면, 스테이징 테이블에서 아무 것도 삭제되지 않습니다. 복사 데이터에서 충돌, 1차 키의 이중 키 또는 고유 색인이 작성되면 복사에 실패할 수 있습니다. 프로덕션 데이터베이스에서 데이터를 복사하지 않고 스테이징 복사를 사용하여 스테이징 데이터베이스만 정리하려면, -cleanup\_stage\_db를 only로 지정하십시오.

### script file

(선택적) 지정된 범위에서 프로덕션 데이터베이스를 스테이징 데이터베이스로 복사하기 위해 반출 및 반입을 사용할 때 스테이징 복사 유틸리티 명령에 의해 작성된 SQL 스크립트 파일의 이름. 또한, 스크립트 파일은 기본값을 사용하거나 -cleanup\_stage\_db를 yes로 지정하는 경우에 스테이징 데이터베이스를 삭제하는 삭제 명령문을 작성합니다.

주: 이 스크립트 파일은 iSeries에서 지원되지 않습니다.

스크립트를 실행하기 전에 반출된 테이블을 보유하는 데 충분한 디스크 공간이 있는지 검증하십시오. 스크립트 파일은 스테이징 복사 유틸리티를 호출한 스테이징 복사 유틸리티 디렉토리에 있습니다.

▶ DB2 스크립트 파일을 실행하려면 `db2 -vtd# -f script_file_name`을 사용하십시오.

## dbtype

(선택적) 데이터베이스 유형 (DB2 또는 Oracle). 기본값은 DB2입니다.

스테이징 서버 복사에 대한 자세한 내용은 예를 참조하십시오.

**스테이징 데이터베이스로의 데이터 복사 예:** 다음은 프로덕션 데이터베이스에서 스테이징 데이터베이스로 테이블을 복사할 수 있는 방법에 관한 예입니다. 프로덕션 시스템에 RFQ가 있는 경우 스테이징 복사 유틸리티를 사용할 수 없다는 것을 기억해야 합니다. 자세한 정보는 40 페이지의 『스테이징 서버 제한 사항』을 참조하십시오.

전체 명령을 한 행에 입력해야 함을 주의하십시오. 여기서는 단지 보기 편하게 하기 위해 둘 이상의 행에 나타낸 것뿐입니다.

### 예 1

스테이징 데이터베이스를 지운 후 범위를 all로 설정하여 프로덕션 데이터베이스를 스테이징 데이터베이스로 복사하십시오.

1. PATH 환경 변수를 설정하십시오. ▶ 400 적용되지 않음.
2. 데이터베이스를 구성하십시오. ▶ 400 적용되지 않음.
3. 로그 파일을 기록할 디렉토리로 변경하십시오. ▶ 400 적용되지 않음. iSeries의 경우 기본 로그 파일은 다음과 같습니다.

```
/QIBM/UserData/CommerceServer55/instances/stagingcopy_{sourcedb_user}
_{destdb_user}_{timestamp}.log
```

4. 다음을 입력하십시오.

- ▶ Windows ▶ DB2  
`stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name`

- ▶ Windows ▶ Oracle  
`stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2  
`. stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name -sourcedb_user user -destdb_user user`

- ▶ AIX   ▶ Solaris   ▶ Oracle

```
. stagingcopy.sh -scope _all_ -sourcedb production_database_name
-destdb staging_database_name dbtype oracle -sourcedb_user user
-sourcedb_passwd password -destdb_user user -destdb_passwd password
```

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

5. stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

로그 파일 이름 및 경로를 지정하려면 로그 파일 매개변수를 사용하십시오.

- ▶ Windows   ▶ DB2

```
stagingcopy -scope _all_ -sourcedb production_database_name -destdb
staging_database_name -log log_file_name
```

- ▶ Windows   ▶ Oracle

```
stagingcopy -scope _all_ -sourcedb production_database_name -destdb
staging_database_name -log log_file_name -dbtype oracle -sourcedb_user
user -sourcedb_passwd password -destdb_user user -destdb_passwd
password
```

+  
+  
+  
+

- ▶ AIX   ▶ 400   ▶ Solaris   ▶ Linux   ▶ DB2

```
. stagingcopy.sh -scope _all_ -sourcedb production_database_name
-destdb staging_database_name -log log_file_name -sourcedb_user user
-destdb_user user
```

- ▶ AIX   ▶ Solaris   ▶ Oracle . stagingcopy.sh -scope \_all\_ -sourcedb
*production\_database\_name* -destdb *staging\_database\_name* -log
*log\_file\_name* dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd
*password* -destdb\_user *user* -destdb\_passwd *password*

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.













예: myhost:1521:mydb

DB2를 사용 중이고 데이터베이스 관리자 로 로그인하지 않은 경우, -sourcedb\_user, -sourcedb\_passwd, -destdb\_user 및 -destdb\_passwd 옵션 값을 제공해야 합니다.

## 예 2

스테이징 데이터베이스에서 MERCHANT 테이블을 삭제한 다음, 판매자 관련 테이블을 프로덕션 데이터베이스에서 스테이징 데이터베이스로 복사하십시오.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name`
-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
- +      `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -sourcedb_user user -destdb_user user`
- +    `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

5. `stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

스테이징 데이터베이스만 정리하려면 `-cleanup_stage_db` 매개변수를 지정하십시오.

-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only`
-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
- +      `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -sourcedb_user user -destdb_user user`
- +    `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

데이터만 복사하려면 `-cleanup_stage_db no` 매개변수를 지정하십시오.

-   stagingcopy -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no
-   stagingcopy -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no -dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*
- +      stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no -sourcedb\_user *user* -destdb\_user *user*
- +    stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.










예: myhost:1521:mydb

중요: 범위를 판매자로 설정하여 복사할 때, 사이트 범위 데이터가 먼저 복사되었는지 확인하십시오. 그렇지 않으면, 복사에 실패합니다.

### 예 3

스테이징 데이터베이스에서 사이트 테이블을 삭제한 다음, 프로덕션 데이터베이스에서 스테이징 데이터베이스로 이 사이트 테이블을 복사하십시오.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-   stagingcopy -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name*
-   stagingcopy -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*
- +      stagingcopy.sh -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -sourcedb\_user *user* -destdb\_user *user*

- ▶ AIX
▶ Solaris
▶ Oracle
 stagingcopy.sh -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* - dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

- stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

사이트 테이블을 삭제하면 연속 삭제 제한 사항 때문에 **MERCHANT** 테이블에 영향을 미칩니다. 먼저 판매자 데이터를 삭제한 다음, 사이트 데이터를 지우고 데이터를 복사하십시오.

- ▶ Windows
▶ DB2
 stagingcopy -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db only

- ▶ Windows
▶ Oracle

```
stagingcopy -scope _merchant_
-sourcedb
```

```
production_database_name -destdb staging_database_name
-cleanup_stage_db only -dbtype oracle -sourcedb_user
user -sourcedb_passwd
password -destdb_user
user -destdb_passwd
password
```

- ▶ AIX
▶ 400
▶ Solaris
▶ Linux
▶ DB2
 stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db only -sourcedb\_user *user* -destdb\_user *user*

- ▶ AIX
▶ Solaris
▶ Oracle
 stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db only dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.






예: myhost:1521:mydb




- ▶ Windows
▶ DB2
 stagingcopy -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db only

- ▶ Windows
▶ Oracle
 stagingcopy -scope \_site\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name*

- cleanup\_stage\_db only -dbtype oracle -sourcedb\_user user  
-sourcedb\_passwd password -destdb\_user user -destdb\_passwd password
- + •      stagingcopy.sh -scope  
\_site\_ -sourcedb production\_database\_name -destdb  
staging\_database\_name -cleanup\_stage\_db only -sourcedb\_user user  
-destdb\_user user
- + •    stagingcopy.sh -scope \_site\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db only dbtype oracle -sourcedb\_user user  
-sourcedb\_passwd password -destdb\_user user -destdb\_passwd password
- + 주: Oracle 데이터베이스 이름으로 host:port:sid를 사용하십시오.  
예: myhost:1521:mydb
- + •   stagingcopy -scope \_site\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db no
- + •   stagingcopy -scope \_site\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db no -dbtype oracle -sourcedb\_user user  
-sourcedb\_passwd password -destdb\_user user -destdb\_passwd password
- + •      stagingcopy.sh -scope  
\_site\_ -sourcedb production\_database\_name -destdb  
staging\_database\_name -cleanup\_stage\_db no -sourcedb\_user user  
-destdb\_user user
- + •    stagingcopy.sh -scope \_site\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db no dbtype oracle -sourcedb\_user user -sourcedb\_passwd  
password -destdb\_user user -destdb\_passwd password
- + 주: Oracle 데이터베이스 이름으로 host:port:sid를 사용하십시오.  
예: myhost:1521:mydb
- + •   stagingcopy -scope \_merchant\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db no
- + •   stagingcopy -scope \_merchant\_ -sourcedb  
production\_database\_name -destdb staging\_database\_name  
-cleanup\_stage\_db no -dbtype oracle -sourcedb\_user user  
-sourcedb\_passwd password -destdb\_user user -destdb\_passwd password




+  
+  
+  
+  
 •      stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no -sourcedb\_user *user* -destdb\_user *user*

•    stagingcopy.sh -scope \_merchant\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -cleanup\_stage\_db no dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*



주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.



예: myhost:1521:mydb





#### 예 4




다음과 같은 스크립트를 작성하여 프로덕션 데이터베이스를 지운 다음 범위를 all로 설정하여 스테이징 데이터베이스로 복사하십시오.  -script 옵션이 iSeries에서는 지원되지 않으므로 이 예는 iSeries용 OS/400에 적용되지 않습니다.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

•   stagingcopy -scope \_all\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -script\_file *stage\_copy.sql*

•   stagingcopy -scope \_all\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -script\_file *stage\_copy.sql* -dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*

+  
+  
+  
 •     stagingcopy.sh -scope \_all\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -script\_file *stage\_copy.sql* -sourcedb\_user *user* -destdb\_user *user*

•    stagingcopy.sh -scope \_all\_ -sourcedb *production\_database\_name* -destdb *staging\_database\_name* -script\_file *stage\_copy.sql* dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

5. stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.



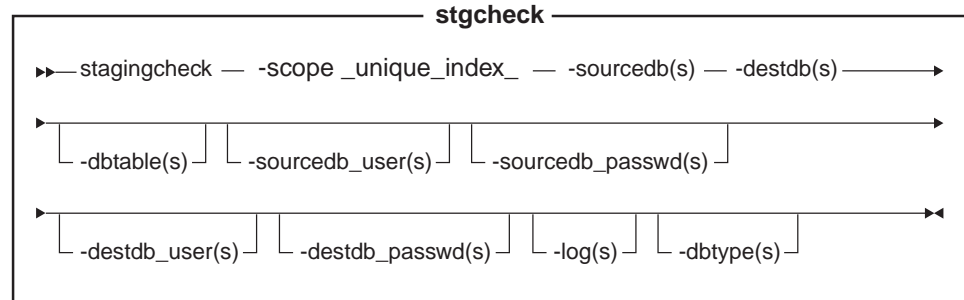
스테이징 복사 유틸리티는 `stage_copy.sql` 스크립트를 작성하여 데이터베이스를 정리하고 복사합니다. **DB2** DB2를 사용하는 경우 다음 스크립트를 실행하십시오.

1. 데이터베이스 관리자(DBA)로 로그인하십시오.
2. DB2 명령창을 여십시오.
3. 다음 명령을 입력하십시오: `db2 -vtd# -f stage_copy.sql`

**Oracle** Oracle을 사용하는 경우 다음 스크립트를 실행하십시오.

1. SQLPlus 창을 여십시오.
2. dba로 연결하십시오.
3. 다음 명령을 입력하십시오: `@stage_copy.sql`

**스테이징 확인 명령(Windows 2000, AIX, iSeries, Linux 및 Solaris):** 스테이징 확인 명령은 스테이징 데이터베이스와 프로덕션 데이터베이스 간에 고유 색인 키 충돌이 있는지 여부를 판별합니다. 스테이징 확인 유틸리티를 실행하려면, 스테이징 서버 또는 프로덕션 서버의 명령행에서 다음을 입력하십시오. 전체 명령을 한 행에 입력하십시오. 여기서는 보기 편하게 하기 위해 둘 이상의 행에 나타냅니다.



**Oracle** 주: 현재 같은 사용자 ID로 이 유틸리티를 실행하는 경우에도 명령에 선택적 매개변수, 로그인 사용자 ID 및 암호를 포함시켜야 합니다.

**400** iSeries는 이제 `stagingCheck.sh`를 사용합니다. iSeries에서 셸 스크립트를 실행하려면 다음을 수행하십시오.

1. 65535 외의 다른 CCSID를 가진 사용자 프로파일로 로그인하십시오.
2. OS/400 명령행에서 STRQSH 명령을 입력하여 QSHELL 명령창을 여십시오.
3. 다음과 같이 명령을 실행하십시오.  
`/QIBM/ProdData/CommerceServer55/bin/stagingCheck.sh(parameters . . .)`

주: 사용자 매개변수를 항상 지정해야 합니다.

#### 매개변수 값

**scope** 스테이징 서버로 복사할 범위의 레벨. `_unique_index_`를 지정하십시오.

### sourcedb

스테이징 서버의 데이터베이스 이름.

▶ Oracle *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

### destdb

프로덕션 서버의 데이터베이스 이름.

▶ Oracle *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

### dbtable

(선택적) 고유 키 충돌을 확인할 특정 테이블의 이름.

### sourcedb\_user

(선택적) 스테이징 데이터베이스의 스키마를 작성한 데이터베이스 관리자의 로그인 ID. 지정하지 않으면 현재 이 유틸리티를 호출 중인 사용자의 ID가 사용됩니다.

▶ 400 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 소스 데이터베이스 스키마와 동일합니다.

### sourcedb\_passwd

(선택적) sourcedb 매개변수로 지정된 로그인 ID의 암호.

### destdb\_user

(선택적) 프로덕션 데이터베이스의 스키마를 작성한 데이터베이스 관리자의 로그인 ID. 지정하지 않으면 유틸리티를 호출 중인 사용자의 ID가 사용됩니다.

▶ 400 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 대상 데이터베이스 스키마와 동일합니다.

### destdb\_passwd

(선택적) destdb\_user 매개변수로 지정된 로그인 ID의 암호. 암호를 지정하지 않으면 시스템에서 암호를 입력하라는 메시지를 표시합니다.

### log

(선택적) 스테이징 복사 유틸리티가 활동 및 오류를 기록하는 파일의 경로 및 이름. 이 매개변수를 지정하지 않으면, 다음 log 디렉토리에 stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log 로그 파일이 작성됩니다.

▶ AIX ▶ Solaris ▶ Linux WC\_installdir/logs

▶ Windows WC\_installdir\logs

▶ 400 WC\_userdir/instances.

### dbtype

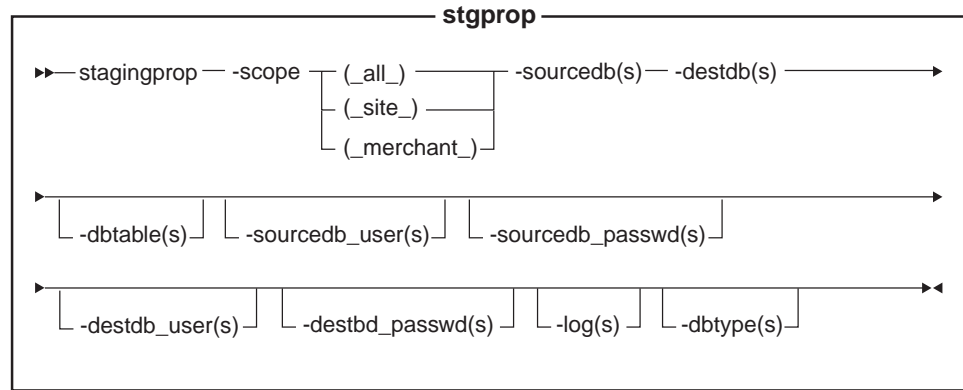
(선택적) 데이터베이스 유형(DB2 또는 Oracle). 기본값은 DB2입니다.

고유 색인 키 충돌 확인에 대한 자세한 내용은 예를 참조하십시오.

### 스테이징 전달 유틸리티 명령(Windows 2000, AIX, iSeries, Linux 및 Solaris):

스테이징 전달 유틸리티는 데이터를 스테이징 데이터베이스에서 프로덕션 데이터베이스로 복사합니다. 스테이징 서버에서는 RFQ 오브젝트를 작성하거나 갱신할 수 없다는 점에 유의하십시오.

스테이징 복사 유틸리티를 실행하려면 스테이징 서버와 프로덕션 서버 데이터베이스에 모두 연결할 수 있는 시스템의 명령행에서 다음을 입력하십시오. 전체 명령을 한 행에 입력하십시오. 여기서는 단지 보기 편하게 하기 위해 둘 이상의 행에 나타난 것입니다.



#### ▶ Oracle

주: 현재 같은 사용자 ID를 사용하여 이 유틸리티를 실행하는 경우에도 명령에 선택적 매개변수, 로그인 사용자 ID 및 암호를 포함시켜야 합니다.

▶ 400 iSeries는 이제 stagingProp.sh를 사용합니다. iSeries에서 셸 스크립트를 실행하려면 다음을 수행하십시오.

1. 65535와의 다른 CCSID를 가진 사용자 프로파일로 로그인하십시오.
2. OS/400 명령행에서 STRQSH 명령을 입력하여 QSHELL 명령창을 여십시오.
3. 다음과 같이 명령을 실행하십시오.

```
/QIBM/ProdData/CommerceServer55/bin/stagingProp.sh(parameters . . .)
```

주: 사용자 매개변수를 항상 지정해야 합니다.

#### 매개변수 값


##### scope

프로덕션 서버로 전달하기 위한 범위 레벨. 다음 중 하나를 지정하십시오.

- `_all_`  
사이트 및 모든 판매자 관련 레코드를 전달하려면 `_all_`을 입력하십시오.
- `_site_`  
사이트 관련 레코드만 전달하려면 `_site_`를 입력하십시오.

- `_merchant_`  
모든 판매자 관련 레코드만 전달하려면 `_merchant_`를 입력하십시오.


#### sourcedb

스테이징 서버의 데이터베이스 이름.  `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

#### destdb

프로덕션 서버의 데이터베이스 이름.

 `host:port:sid`를 사용하십시오.


예: `myhost:1521:mydb`

#### dbtable

(선택적) 전달될 특정 테이블의 이름. 레코드가 `scope` 매개변수로 지정된 범위 내에 있으면 이 테이블에서 변경된 모든 레코드가 전달됩니다. 그렇지 않으면, 레코드가 전달되지 않습니다.

#### sourcedb\_user

(선택적) 소스 데이터베이스 스키마를 작성한 데이터베이스 관리자의 로그인 ID. 지정하지 않으면 현재 이 유틸리티를 호출 중인 사용자의 ID가 사용됩니다.


 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 소스 데이터베이스 스키마와 동일합니다.

#### sourcedb\_passwd

(선택적) `sourcedb_user` 매개변수로 지정된 로그인 ID의 암호.

#### destdb\_user

(선택적) 대상 데이터베이스 스키마를 작성한 데이터베이스 관리자의 로그인 ID. 지정하지 않으면 유틸리티를 호출 중인 사용자의 ID가 사용됩니다. 원격 데이터베이스를 사용하는 경우 이 매개변수는 필수입니다.

 (필수) Commerce 인스턴스와 연관된 사용자 프로파일. 대상 데이터베이스 스키마와 동일합니다.

#### destdb\_passwd

(선택적) `destdb_user` 매개변수로 지정된 로그인 ID의 암호. 암호를 지정하지 않으면 시스템에서 암호를 입력하라는 메시지를 표시합니다. 원격 데이터베이스 사용시 이 매개변수는 필수 매개변수입니다.

#### log

(선택적) 스테이징 전달 유틸리티가 활동 및 오류를 기록하는 파일의 경로 및 이름. 이 매개변수를 지정하지 않으면, 다음 `log` 디렉토리에 `stagingprop_yyyy.mm.dd_hh.mm.ss.zzz.log` 로그 파일이 작성됩니다.

 `Solaris`  `WC_installdir/logs`

 `WC_installdir\logs`

 `WC_userdir/instances`

## dbtype

(선택적) 데이터베이스 유형(DB2 또는 Oracle). 기본값은 DB2입니다.

스테이징 서버로 전파하는 방법에 관한 추가 정보는 다음 예를 참조하십시오.













**프로덕션 데이터베이스로 데이터 전달 예:** 다음은 변경된 레코드를 스테이징 데이터베이스에서 프로덕션 데이터베이스로 전달하는 방법의 예입니다.

명령을 한 행에 입력해야 함을 주의하십시오. 여기서는 단지 보기 편하게 하기 위해 둘 이상의 행에 나타낸 것뿐입니다.

### 예 1

모든 변경사항을 스테이징 서버 데이터베이스에서 프로덕션 데이터베이스로 전달합니다.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-   stagingprop -scope \_all\_ -sourcedb *staging\_database\_name* -destdb *production\_database\_name*
-   stagingprop -scope \_all\_ -sourcedb *staging\_database\_name* -destdb *production\_database\_name* -dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*
-      stagingprop.sh -scope \_all\_ -sourcedb *staging\_database\_name* -destdb *production\_database\_name* -sourcedb\_user *user* -destdb\_user *user*
-    stagingprop.sh -scope \_all\_ -sourcedb *staging\_database\_name* -destdb *production\_database\_name* dbtype oracle -sourcedb\_user *user* -sourcedb\_passwd *password* -destdb\_user *user* -destdb\_passwd *password*













주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

5. 로그 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오. 로그 매개변수에 대한 설명을 확인하여 데이터베이스 유형 및 플랫폼에 대한 이 로그 파일의 기본 위치를 판별하십시오.

스테이징 전달 유틸리티는 먼저 모든 사이트 데이터를 전달한 다음 모든 판매자 데이터를 프로덕션 데이터베이스로 전달합니다. 오류가 발생하면 전체 트랜잭션이 롤백됩니다.

로그 파일 이름 및 경로를 지정하려면 로그 파일 매개변수를 사용하십시오.

-   stagingprop -scope \_all\_  
-sourcedb staging\_database\_name -destdb production\_database\_name  
-loglog\_file\_name
-   stagingprop -scope \_all\_  
-sourcedb staging\_database\_name -destdb production\_database\_name  
-loglog\_file\_name -dbtype oracle -sourcedb\_user user -sourcedb\_passwd  
password -destdb\_user user -destdb\_passwd password
- +      stagingprop.sh -scope  
+ \_all\_ -sourcedb staging\_database\_name -destdb production\_database\_name  
+ -loglog\_file\_name -sourcedb\_user user -destdb\_user user
-     
stagingprop.sh -scope \_all\_ -sourcedb staging\_database\_name -destdb  
production\_database\_name -loglog\_file\_name -dbtype oracle  
-sourcedb\_user user -sourcedb\_passwd password -destdb\_user user  
-destdb\_passwd password

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.










예: myhost:1521:mydb

DB2를 사용 중이고 데이터베이스 관리자로 로그인하지 않은 경우, -sourcedb\_user, -sourcedb\_passwd, -destdb\_user 및 -destdb\_passwd 옵션 값을 제공해야 합니다.




## 예 2

수정된 모든 사이트 데이터를 스테이징 데이터베이스에서 프로덕션 데이터베이스로 전달합니다.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-   stagingprop -scope \_site\_  
-sourcedb staging\_database\_name -destdb production\_database\_name
-   stagingprop -scope \_site\_  
-sourcedb staging\_database\_name -destdb production\_database\_name  
-dbtype oracle -sourcedb\_user user -sourcedb\_passwd password  
-destdb\_user user -destdb\_passwd password
- +      stagingprop.sh -scope  
+ \_site\_

+ -sourcedb *staging\_database\_name* -destdb *production\_database\_name*  
+ -sourcedb\_user *user* -destdb\_user *user*

-    `stagingprop.sh -scope _site_  
-sourcedb staging_database_name-destdb production_database_name  
dbtype oracle -sourcedb_user user -sourcedb_passwd password  
-destdb_user user -destdb_passwd password`

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.





예: *myhost:1521:mydb*






5. 로그 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오. 로그 매개변수에 대한 설명을 확인하여 데이터베이스 유형 및 플랫폼에 대한 이 로그 파일의 기본 위치를 판별하십시오.




### 예 3

(사이트 데이터를 전달한 후) 스테이징 데이터베이스에서 프로덕션 데이터베이스로 수정된 모든 판매자 데이터를 전달합니다.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-   `stagingprop -scope _merchant_  
-sourcedb staging_database_name-destdb production_database_name`
-   `stagingprop -scope _merchant_  
-sourcedb staging_database_name -destdb production_database_name  
-dbtype oracle -sourcedb_user user -sourcedb_passwd password  
-destdb_user user -destdb_passwd password`

+      `stagingprop.sh -scope  
+ _merchant_ -sourcedb staging_database_name -destdb  
+ production_database_name -sourcedb_user user -destdb_user user`

-    `stagingprop.sh -scope _merchant_  
-sourcedb staging_database_name -destdb production_database_name  
dbtype oracle -sourcedb_user user -sourcedb_passwd password  
-destdb_user user -destdb_passwd password`

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.


예: *myhost:1521:mydb*

5. 로그 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오. 로그 매개변수에 대한 설명을 확인하여 데이터베이스 유형 및 플랫폼에 대한 이 로그 파일의 기본 위치를 판별하십시오.



## 사용자 정의 데이터베이스 테이블 요구사항

새 테이블을 작성하여 데이터베이스 스키마를 사용자 정의한 경우, 스테이징 서버를 사용하려면 다음 요구사항에 만족되어야 합니다.

- **1차 키 또는 고유 색인을 정의해야 합니다.**  
스테이징 서버는 키에 기초하여 수행됩니다. STAGLOG 테이블에 너무 많은 데이터가 기록되는 것을 방지하려면 키(1차 키 및 고유 색인)만 기록하십시오. 스테이징 유틸리티는 압축을 위해 키를 사용하고 전달될 데이터를 찾습니다. 키가 없으면 스테이징 유틸리티는 작동할 수 없습니다.
- **참조 무결성(RI) 제한조건 순환은 여러 테이블에 있을 수 없습니다.**  
스테이징 서버는 항상 하위 테이블 전에 상위 테이블을 전달합니다. RI 제한조건 순환이 없으면 스테이징 서버는 상위와 하위 테이블을 구별할 수 없습니다.
- **데이터베이스 테이블에는 구성 데이터만 포함됩니다.**  
비즈니스 대 고객 시나리오에서 카탈로그 및 카탈로그 항목과 같은 구성 데이터는 사이트 운영자의 제어하에 있습니다. 테이블에 조작 데이터가 들어 있는 경우, 고객은 사이트 운영자가 스테이징 데이터베이스로 이 테이블을 복사한 다음 프로덕션 데이터베이스에 있는 같은 테이블을 변경할 수 있습니다. 이 때 키 충돌 또는 RI 제한조건 위반이 발생할 수 있습니다.
- **데이터베이스 테이블에는 조작 테이블 참조가 포함될 수 없습니다.**  
전달할 테이블에는 조작 테이블의 1차 키에 대한 외부 키 참조가 포함되어서는 안 됩니다. 이러한 참조가 있을 경우, 고객이 스테이징 복사 후에 1차 키를 삭제하면 데이터를 복원할 수 없습니다.
-  두 테이블을 프로덕션 데이터베이스에 삽입하는 경우 삽입 트리거는 사용할 수 없습니다.  
스테이징 서버가 관리하는 두 테이블(예를 들어, R1 및 R2)의 경우, 프로덕션 데이터베이스에서 R2 및 R1에 삽입할 때 R1 또는 R2에 행을 삽입하는 트리거가 존재해서는 안 됩니다. 삽입 트리거는 두 데이터베이스에서 모두 갱신을 작성하며 키 문제점을 발생시킵니다.
- **MEMBER 테이블에는 고유 색인이 포함될 수 없습니다.**
- **사용자 정의 데이터베이스 테이블의 삭제 제한은 주의하여 사용해야 합니다.**  
삭제 제한은 데이터베이스 정리 유틸리티의 성능을 저하시킵니다. 스테이징 데이터베이스를 정리할 때 문제가 발생할 수도 있습니다. 스테이징 데이터베이스를 정리하기 전에 force 옵션을 가진 데이터베이스 정리 유틸리티 명령을 사용하여 직접 테이블을 정리하십시오. 그렇지 않으면, 스테이징 데이터베이스 정리에 실패합니다.

사용자 정의 테이블에 대해 스테이징 서버를 준비하려면 사용자 정의 테이블에 대해 스테이징 서버 구성을 참조하십시오.



## 스태이징 서버 문제점 해결(Windows 2000, AIX, Linux 및 Solaris)

1. 스테이징 서버 명령 stgprop을 사용하는 중에 다음 메시지를 수신할 수 있습니다.  
SQLSTATE 54001: 명령문이 너무 길거나 복잡합니다.

데이터베이스의 stmheap 크기를 데이터베이스 구성시 제안된 대로 설정했는지 확인하십시오. 증분을 지원할 만큼 메모리가 충분한지 확인해야 합니다.

2. 스테이징 복사 유틸리티가 처리를 완료하지 못했다면 데이터베이스 로그가 너무 작기 때문일 수 있습니다. DB2 명령창에서 다음 명령을 실행하십시오.

```
update database configuration for staging_server using logprimary 50
logfilsiz 1000
db2 terminate
db2stop
db2start
```

여기서, *staging\_server*는 스테이징 서버 데이터베이스의 이름입니다. 문제가 지속되면, logprimary 또는 logfilsiz 매개변수에 더 큰 값을 사용해 보십시오.

3. **AIX** db2inst1과 같은 DB2 사용자의 자원이 올바르게 구성되었는지 확인하십시오. 다음 명령을 입력하면 자원 제한 사항이 표시됩니다. ulimit -a  
데이터 세그먼트를 240MB로 구성하고 스택을 16MB로 구성하십시오.  
추가 정보는 *DB2* 명령 참조서를 참조하십시오.

## WebSphere Commerce 스테이징 테이블

프로덕션 서버에서 스테이징 서버로 WebSphere Commerce 스테이징 테이블을 복사할 수 있습니다. 이들 테이블은 사이트 관련 데이터, 판매자 관련 데이터 또는 사이트와 판매자 관련 데이터가 모두 들어 있는지에 따라 그룹화됩니다.

각 테이블 그룹은 다음과 같은 페이지에서 표시됩니다.

- 사이트 데이터 범위
- 판매자 데이터 범위
- 사이트 및 판매자 데이터 범위

## 스태이징 테이블의 트리거

WebSphere Commerce 스테이징 테이블에 대해 다음 트리거가 정의되었습니다. 사용자 정의 테이블에 같은 데이터 범위와 키 특성이 있으면 다음 설정을 적용시킬 수 있습니다. 데이터 범위로 테이블을 보려면 아래의 링크를 참조하십시오.

- 사이트 데이터 범위
- 판매자 데이터 범위
- 사이트 또는 판매자 데이터 범위

사이트 데이터 범위: 다음 도표는 사이트 데이터 범위의 데이터베이스 테이블을 표시합니다.

테이블 이름	고유 색인 사용				
	정수 1	정수 2	도표 1	정수 3	도표 2
ACACGPDESC	예	예			
ACACTACTGP	예	예			
ACACTDESC	예	예			
ACACTGRP	예				
ACACTION	예				
ACATTR	예				
ACATTRDESC	예	예			
ACCCMDTYPE			예		
ACRELATION	예				
ACRELDISC	예	예			
ACRESACT	예	예			
ACRESATREL	예	예			
ACRESCGRY	예				
ACRESGRP	예				
ACRESGPDES	예	예			
ACRESGPRES	예	예			
ACRESPRIM	예				
ACRESREL	예	예			
ATTACHUSG			예		
ATTRTYPE			예		
BUYERPOTYP	예				
CALUSAGE	예				
CATENTTYPE			예		
CATRELTYPE			예		
CHKARRANG	예	예			
CHKCMD	예				
COUNTCODE			예		예
COUNTRY	예		예		
DEVICEFMT	예				
FLCOMPOSE	예	예		예	
FLDOMNDESC	예	예			
FLOW	예				
FLOWDESC	예	예			
FLOWDOMAIN	예				
FLOWTYPE	예				
FLSTATEDCT	예				
FLSTATEGP	예				

FLSTATEREL	예	예			
FLSTDCTDSC	예	예			
FLSTGPDSC	예	예			
FLTRANDSC	예	예			
FLTRANSITN	예				
FLTYPEDESC	예	예			
ICDATAREG	예				
INVRSRVTYP	예				
INVRSRVDSC	예	예			
ITEMTYPE			예		
LANGUAGE	예				
LANGUAGEDS	예	예			
MASSOC			예		
MASSOCTYPE			예		
MBRATTR	예				
MBRGRPTYPE	예				
OPERATOR	예				
OPERATRDSC	예	예			
ORDCHNLTY	예				
OUTPUTQ	예				
OUTPUTQDSC	예	예			
PARTROLE	예				
PARTROLED	예	예			
PLCYTYPDSC	예		예		
POLICYTYPE	예				
QTYCONVERT			예		예
QTYUNIT			예		
QTYUNITDSC	예		예		
ROLE	예				
SCHCMD	예				
SETCURR			예		
SETCURRDSC	예		예		
STATECODE			예		예
STATEPROV	예		예		
STORECGRY	예				
TAXTYPE	예				
TCSUBTYPDS	예		예		
TCSUBTYPE			예		
TCTYPE			예		
TFALGOPOL	예				
TFALGOTYPE	예				
TFALGPOLDS	예	예			

TFALGTYPDS	예	예			
TFDOMAIN	예				
TFDOMDSC	예	예			
TFSBDOMAIN	예				
TFSBDOMDSC	예	예			
TRDTYPE	예				
TRDTYPEDESC	예	예			
TXCDScheme	예				

**판매자 데이터 범위:** 다음 도표는 판매자 데이터 범위에 있는 데이터베이스 테이블을 표시합니다.

테이블 이름	고유 색인 사용				
	정수 1	정수 2	도표 1	정수 3	도표 2
ACCCMDGRP	예				
ACCCUSTEXC	예	예			
ACCOUNT	예				
ACORGPOL	예	예			
ACPOLDESC	예	예			
ACPOLICY	예				
ATTACHMENT	예				
ATTRIBUTE	예	예			
ATTRVALUE	예	예			
BASEITEM	예				
BASEITEM	예	예			
BUYERPO	예				
CALCODE	예				
CALCODEDSC	예	예			
CALCODEMGP	예	예			
CALCODTXEX	예	예			
CALMETHOD	예				
CALRANGE	예				
CALRLOOKUP	예				
CALRULE	예				
CALRULEMGP	예	예			
CALSCALE	예				
CALSCALEDS	예	예			
CATALOG	예				
CATALOGDSC	예	예			
CATCNTR	예	예			
CATCONFINF	예				
CATENCALCD	예				

CATENTATTR	예				
CATENTDESC	예	예			
CATENTREL	예	예	예		
CATENTRY	예				
CATENTSHIP	예				
CATGPCALCD	예				
CATGPENREL	예	예		예	
CATGROUP	예				
CATGRPATR	예	예			
CATGRPDESC	예	예			
CATGRPPS	예	예		예	
CATGRPREL	예	예		예	
CATGRPTPC	예	예		예	
CATTOGRP	예	예			
CHARGETYPE	예				
CHRGTYPDSC	예	예			
CNTRDISPLY	예				
CNTRNAME	예		예		
CONTRACT	예				
CREDITLINE	예				
CRULESCALE	예	예			
CURCONVERT	예				
CURCVLIST	예	예	예		예
CURFMTDESC	예	예	예		
CURFORMAT	예		예		
CURLIST	예		예		
DISPCGPREL	예				
DISPENTREL	예				
DISTARRANG	예				
FLOWADMIN	예				
INVADJCODE	예				
INVADJDESC	예	예			
ITEMSPC	예				
ITEMVERSN	예				
JURST	예				
JURSTGPREL	예	예			
JURSTGROUP	예				
LANGPAIR	예	예		예	
LISTPRICE	예		예		
MASSOCCECE	예				
MASSOCGPGP	예				
MBRATTRVAL	예				

MBRGRPUSG	예	예			
MBRREL	예	예			
MBRROLE	예	예		예	
MGPTRDPSCN	예	예			
OFFER	예				
OFFERDESC	예	예			
OFFERPRICE	예		예		
PARTICIPNT	예				
PATRDESC	예	예			
PATATTRIBUTE	예				
PATTRPROD	예	예			
PKGATTR	예				
PKGATTRVAL	예				
PKGITEMREL	예	예		예	
POLICY	예				
POLICYCMD	예		예		
POLICYDESC	예	예			
POLICYTC	예	예			
PRODSETDSC	예	예			
PRODUCTSET	예				
PRSETCEREL	예	예			
QTYFMTDESC	예	예	예		
QTYFORMAT	예		예		
RTNDNYDESC	예	예			
RTNDNYRSN	예				
RTNDSPCODE	예				
RTNDSPDESC	예	예			
RTNREASON	예				
RTNRSNDESC	예	예			
SHPARJURGP	예	예			
SHPARRANGE	예				
SHPJCRULE	예				
SHPMODEDESC	예	예			
STENCALUSG	예	예			
STORECAT	예	예			
STORECENT	예	예			
STORECGRP	예	예			
STORECNTR	예	예			
STOREDEF	예				
STOREITEM	예	예			
STORELANG	예	예			
STOREMBRGP	예	예			

STORITMFFC	예	예		예	
STORLANGDS	예	예		예	
TAXCGRY	예				
TAXCGRYDS	예	예			
TAXJCRULE	예	예		예	
TCDESC	예	예			
TDPSCNCNTR	예	예			
TERMCOND	예				
TRADEPOSCN	예				
TRADING	예				
TRDATTACH	예	예			
TRDESC	예	예			
TXCDCLASS	예				
VENDOR	예				
VENDORDESC	예	예			
VERSIONSPC	예				
VIEWREG	예	예	예		

**사이트 및 판매자 데이터 범위:** 다음 도표에서는 사이트 및 판매자 데이터 범위 내의 데이터베이스 테이블을 표시합니다.

테이블 이름	고유 색인 사용				
	정수 1	정수 2	도표 1	정수 3	도표 2
CMDREG	예		예		
FFMCENTDS	예	예			
FFMCENTER	예				
MBRGRP	예				
MBRGRPCOND	예				
MEMBER	예				
ORGENTITY	예				
SHIPMODE	예				
STADDRESS	예				
STORE	예				
STOREENT	예				
STOREENTDS	예	예			
STOREGRP	예				
URLREG	예		예		
VIEWREG	예	예	예		

## 연결 서비스

WebSphere Commerce는 외부 시스템과 연결하기 위해 HTTP 프로그램 어댑터, WebSphere MQ용 어댑터 및 CrossWorlds용 어댑터를 사용합니다. 다음 절에서는 각 어댑터 및 WebSphere Commerce에서 각 어댑터의 사용 방법에 대해 설명합니다.

### HTTP용 프로그램 어댑터

HTTP용 프로그램 어댑터를 사용하면 외부 시스템이 HTTP 프로토콜을 통해 XML 요청을 전달함으로써 WebSphere Commerce와 통신할 수 있습니다. HTTP용 프로그램 어댑터는 조달 시스템과 같은 외부 시스템에 HTTP를 통해 WebSphere Commerce와 통신하는 일반적인 방법을 제공하여 구매자/공급자 트랜잭션에서 WebSphere Commerce가 이러한 시스템의 공급자 역할을 수행할 수 있도록 합니다. HTTP용 프로그램 어댑터는 다음 조치를 수행하여 수신되는 XML 요청을 처리합니다.

- 요청을 인식하여 이것이 XML 요청인지를 검증합니다. 다음 세 가지의 요청 속성이 지원되면 XML 요청으로 구분할 수 있습니다.
  - 콘텐츠 유형
  - 방법
  - 문자 인코딩지원되는 요청 속성은 어댑터 구성에 지정되어 있습니다.
- 요청의 입력 스트림 추출.
- 메시지 매퍼 호출 및 입력 스트림의 콘텐츠 전달.
- 메시지 매퍼가 리턴한 WebSphere Commerce 명령을 나타내는 CommandProperty 오브젝트 수신.
- 응답을 작성할 적절한 장치 형식 결정.
- 명령 실행.
- JSP로 작성되고, 명령 실행으로 지정된 보기 이름 및 수신된 요청의 장치 형식에 기초한 XML 응답 메시지 송신.

각 요청은 다른 세션으로 취급됩니다. 메시지의 신임장은 메시지의 제어 영역에 지정됩니다. 기본적으로 Program Adapter for HTTP는 메시지의 사용자 ID 및 암호를 매개변수를 확인하여 요청의 인증성을 판별합니다. Program Adapter for HTTP는 레거시 메시지가 신임장 스펙을 지원하지 않으므로 레거시 메시지를 지원하지 않습니다.

Program Adapter for HTTP의 사용 주기는 WebSphere Commerce 인스턴스 전체에 존재합니다. 프로그램 어댑터는 구성 매개변수가 제거되었거나 어댑터가 사용 불가능하지 않으면 인스턴스를 시작할 때 초기화되며, 인스턴스가 실행되는 동안 상주합니다.

주: HTTP용 프로그램 어댑터는 기본적으로 사용되지 않습니다.



WebSphere Commerce가 장치에서 수신한 요청을 처리하는 방법에 대한 구조적 정보는 *WebSphere Commerce 프로그래밍 안내서* 및 학습서를 참조하십시오. HTTP용 프로그램 어댑터의 구성 및 기타 연결 서비스에 관한 정보는 WebSphere Commerce Production 온라인 도움말을 참조하십시오.

## WebSphere MQ용 리스너

WebSphere MQ용 리스너는 WebSphere MQ를 통해 인바운드 메시지를 처리하여 백엔드 시스템과의 통합을 가능케 하는 WebSphere Commerce의 구성요소입니다. The Listener for WebSphere MQ용 리스너는 MQ 메시지를 검색하기 위한 Websphere MQ용 어댑터와 해당 메시지를 실행하기 위해 호출하는 프로그램 어댑터의 조합입니다.

WebSphere MQ용 리스너에는 WebSphere Commerce 비즈니스 처리와 백엔드 또는 외부 시스템 처리의 통합에 도움을 주는 사전 정의된 메시지 세트가 있습니다. 수신되는 각 메시지는 WebSphere Commerce 내의 프로세스를 활성화하여 데이터베이스 테이블을 갱신하거나 다른 조작을 수행합니다. 제공된 메시지에 대한 자세한 내용은 WebSphere Commerce 온라인 도움말에서 백엔드 통합 및 서비스 센터 통합 메시지 정보를 참조하십시오. 기존의 사전 정의된 메시지 이외에 WebSphere MQ용 리스너는 메시지 확장 및 새 메시지를 지원합니다.

**미들웨어로서의 WebSphere MQ:** WebSphereMQ용 어댑터는 WebSphere MQ를 미들웨어로 사용하여 백엔드 및 외부 시스템을 WebSphere Commerce와 통합할 수 있도록 합니다. WebSphereMQ용 어댑터는 WebSphere Commerce가 백엔드 시스템 및 외부 시스템에서 메시지를 받을 수 있도록 합니다. 지원되는 소프트웨어는 WebSphere MQ 클라이언트 제품 확장이 포함된 WebSphere MQ 버전 5.3 이상입니다. WebSphere MQ 클라이언트는 WebSphere MQ용 클라이언트이며 별도로 다운로드할 수 있습니다.

바인딩 또는 클라이언트 모드를 사용하여 WebSphere MQ를 미들웨어로 설정할 수 있습니다. 바인딩 모드는 WebSphere Commerce가 WebSphere MQ 서버와 동일한 시스템에 설치되어 JNI(Java Native Interface)를 사용하여 WebSphere MQ Java를 통해 WebSphere MQ 서버에 연결하는 경우에 사용해야 합니다. 통신이 네트워크를 이용하지 않고 큐 관리자 API에 대해 직접 JNI를 호출하여 이루어지므로 바인딩 모드는 일반적으로 네트워크 연결을 사용하는 클라이언트 모드에 비해 더 좋은 성능을 제공합니다. 클라이언트 모드는 WebSphere Commerce가 한 시스템에 설치되고 WebSphere MQ 서버가 백엔드 시스템에 설치되어 있는 경우에 사용해야 합니다.

WebSphere MQ 연결, 큐 및 채널을 확인하려면 테스트 프로그램을 실행하여 WebSphere MQ 큐에 메시지를 넣고 가져오십시오. 자세한 내용은 WebSphere MQ 문서 및 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오.

## Adapter for CrossWorlds

Adpater for CrossWorlds는 IBM CrossWorlds와의 IBM WebSphere Commerce 비즈니스 통합을 확장하는 새로운 메커니즘을 제공합니다. 이 새 어댑터는 외부 프로세스

가 IBM CW ICS(CrossWorlds InterChange Server)에서 협업을 실행할 수 있도록 하는 응용프로그램 인터페이스(API)인 CrossWorlds Server Access Interface를 사용합니다. WebSphere Commerce는 이 어댑터를 사용하여 IBM CrossWorlds를 통해 외부 시스템에 동기 메시지를 전송하여 외부 시스템과 통합할 수 있습니다. 이 어댑터를 사용하면 WebSphere Commerce가 CrossWorlds 서버에 메시지를 보내고 응답을 대기할 수 있습니다. 응답을 받은 후, 호출 명령은 계속 다른 비즈니스 로직을 처리할 수 있습니다.

IBM CrossWorlds 시스템은 협업이라고 불리는 공통 비즈니스 통합 요구사항을 위한 사전 빌드된 비즈니스 로직 템플릿과 다양한 개발 및 관리 도구를 포함하고 있는 소프트웨어 통합 제품 모음입니다. 협업은 주문 관리 및 제조 부품표 관리와 같은 공통적인 산업에 고유한 비즈니스 프로세스 단계를 정의하고 자동화합니다. 또한 협업을 사용하여 다른 종류의 엔터프라이즈 소프트웨어 제품의 비즈니스 프로세스를 통합하고 확장할 수 있으며, 그들 사이의 의미있는 데이터 교환을 용이하게 합니다.

주: Adapter for CrossWorlds에대한 추가 정보는 *WebSphere Commerce 추가 소프트웨어 안내서* 및 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오.

### **WebSphere Commerce와 IBM CrossWorlds InterChange Server의 통합**

: Websphere Commerce는 IBM CrossWorlds ICS(InterChange Server)와 통합하도록 설계되었습니다. 이렇게 하면 WebSphere Commerce를 ICS를 통해 백엔드 시스템과 통합할 수 있습니다. WebSphere Commerce와 ICS의 통합은 다음 세 가지 유형의 메시지 플로우를 지원합니다.

- 아웃바운드 비동기 메시지 플로우를 사용하여 Websphere Commerce는 처리할 메시지를 ICS로 보낼 수 있습니다.
- 인바운드 메시지 플로우를 사용하여 Websphere Commerce는 ICS가 보낸 메시지를 수신할 수 있습니다.
- 동기 메시지 플로우를 사용하면 WebSphere Commerce가 ICS로 요청/응답 메시지를 보낼 수 있습니다.

아웃바운드 및 인바운드 플로우는 WebSphere Commerce용 WBI(WebSphere Business Integration) 어댑터를 사용합니다. 요청/응답 플로우는 ICS용 WebSphere Commerce SAI 어댑터를 사용합니다.

**WebSphere Commerce 아웃바운드 프로세스:** WebSphere Commerce의 트랜잭션은 WebSphere Commerce 메시지 전달 시스템을 통해 기타 응용프로그램으로 보낸 메시지를 트리거할 수 있습니다. 메시지 전달 시스템을 통해 작성된 메시지를 사용하여 WebSphere Business Integration Server와 같은 통합 서버를 통하여 기타 응용프로그램을 통합할 수 있습니다. 예를 들면, OrderCreate XML 메시지를 WebSphere Commerce에서 작성하여 백엔드 시스템에서 주문을 작성할 수 있습니다.

WebSphere Commerce에서 주문을 작성할 때, 주문 관련 정보를 이행과 같은 추후 처리를 위해 다른 시스템으로 보내도록 구성할 수 있습니다. IBM Websphere Business Integration Server를 사용하는 고객은 Websphere Commerce의 XML 메시지를 처리를 위해 IBM CrossWorlds ICS로 보내도록 WebSphere Commerce와 IBM CrossWorlds ICS를 구성할 수 있습니다. 이 경우, 주문은 XML 메시지를 통해 WebSphere Commerce에서 ICS로 보내집니다. 이 XML 메시지는 IBM CrossWorlds ICS에 의해 처리되어 SAP과 같은 다른 시스템으로 보낼 수 있습니다.

WebSphere Commerce에서 주문을 작성할 때, XML 형식의 OrderCreate 메시지가 작성되어 WebSphere MQ 출력 큐에 배치됩니다. WebSphere Commerce 에이전트용 WBI 어댑터는 계속해서 새로운 메시지를 풀하여 WebSphere MQ 출력 큐에서 커넥터 컨트롤러로 전달합니다.

컨트롤러는 WebSphere Commerce 특정 비즈니스 오브젝트를 수신하고 맵을 호출하여 일반 비즈니스 오브젝트(GBO)를 작성합니다. GBO는 대응하는 협업으로 전달되며, 협업이 GBO를 처리하고 요청을 백엔드 시스템으로 보냅니다. Websphere Commerce 및 IBM CrossWorlds ICS 통합에 대한 자세한 정보는 Websphere Commerce용 WBI 어댑터 문서(<http://www-3.ibm.com/software/websphere/crossworlds/library/doc/v411/welcome.html>)를 참조하십시오.

**WebSphere Commerce 인바운드 프로세스:** Websphere Commerce는 ICS에서 Websphere Commerce로 수신되는 요청을 지원하기 위해 IBM CorssWorlds ICS와 통합하도록 설계되었습니다. Websphere Commerce 메시지 전달 시스템이 Websphere Commerce 비즈니스 로직을 호출하여 ICS에서 수신되는 요청을 지원합니다.

이 시나리오에서는 기존 WebSphere Commerce 메시지를 사용하여 IBM CrossWorlds ICS에서 수신되는 비즈니스 로직 트리거 요청을 호출합니다.

WebSphere Commerce용 WBI 어댑터는 협업으로부터 비즈니스 오브젝트를 수신하고 데이터 핸들러를 사용하여 비즈니스 오브젝트를 XML 형식 메시지로 변환한 다음, 메시지를 WebSphere MQ 큐로 전달합니다. WebSphere Commerce와 IBM CrossWorlds ICS 통합에 대한 자세한 정보는 CrossWorlds용 어댑터 문서(<http://www-3.ibm.com/software/websphere/crossworlds/library/doc/v411/welcome.html>)를 참조하십시오.

---

## 구성 태스크

### 메시지 전달 서비스 구성

사이트 운영자는 시스템 관리 중에 다음 태스크를 수행할 수 있습니다.

- 메시지 전달 수단("전송"으로 부름) 추가
- 전송 구성

- 메시지 전송 지정 보기
- 메시지 유형 구성(메시지 설명을 추가, 변경 또는 삭제, 심각도, 전송 또는 장치 형식)
- 전송 방법을 상점에 추가
- 전송 방법 활성화 또는 비활성화

메시지 전달 시스템과 대한 추가 정보는 **WebSphere Commerce Production** 온라인 도움말을 참조하십시오.

## 스케줄 서비스 구성

관리 콘솔에서 스케줄러에 액세스하십시오. 스케줄러 사용에 대한 추가 정보는 **WebSphere Commerce Production** 온라인 도움말을 참조하십시오. 스케줄러 유지보수에 대한 정보는 160 페이지의 『스케줄러 유지보수』를 참조하십시오.

## 로그 작성 구성

모든 로그 작성 및 추적 구성은 이제 **WebSphere Application Server** 관리 콘솔에서 수행합니다. 다음 절은 독립형 도구만을 다룹니다.

- ECMessage 시작 옵션 수정
- ECTrace 시작 옵션 수정

**WebSphere Application Server** 로그 작성 및 추적에 대한 추가 정보는 **WebSphere Application Server** 정보 센터를 참조하십시오.

## 추적 구성요소 사용

**WebSphere Commerce** 시스템을 추적하려면 **WebSphere Application Server** 관리 콘솔을 사용해야 합니다.

### 서버 시작 시 추적 사용:

1. **WebSphere Application Server** 관리 콘솔을 여십시오. 이의 수행 방법 정보는 **WebSphere Application Server** 정보 센터를 참조하십시오.
2. 콘솔 탐색 트리에서 문제점 해결 > 로그 작성 및 추적을 누른 다음 서버 > 진단 추적을 누르십시오.
3. 구성을 누르십시오.
4. 추적 사용 선택란을 선택하여 추적을 사용하거나 선택란을 지워서 추적을 사용 불가능하게 하십시오.
5. 해당 TraceString을 입력하여 추적 스펙을 원하는 상태로 설정하십시오.
6. 추적 출력을 파일로 보낼 것인지 메모리 상의 순환 버퍼로 보낼 것인지 선택하십시오.
7. 작성된 추적에 대해 원하는 포맷을 선택하십시오.

8. 변경된 구성을 저장하십시오.
9. 서버를 시작하십시오.

주: 이들 단계에 대한 추가 정보는 WebSphere Application Server InfoCenter를 참조하십시오.

**실행 중인 서버에서 추적 사용:** 다음 프로시저를 사용하여 실행 중인 서버에 대해 활발하게 추적되고 있는 구성요소를 판별하는 추적 서비스 상태를 수정할 수 있습니다.

1. WebSphere Application Server 관리 콘솔을 여십시오. 이를 수행하는 방법에 대한 정보는 다음을 참조하십시오.
2. 콘솔 탐색 트리에서 문제점 해결 > 로그 작성 및 추적을 누른 다음 서버 > 진단 추적을 누르십시오.
3. 런타임 탭을 선택하십시오.
4. 추적 스펙을 원하는 상태로 변경하여 기존 추적 상태를 변경하십시오.
5. (선택) 기존 추적 출력을 변경하려면 추적 출력을 구성하십시오.
6. 적용을 누르십시오.

주: 이들 단계에 대한 추가 정보는 WebSphere Application Server InfoCenter를 참조하십시오.

**WebSphere Commerce 추적 구성요소:** 다음은 WebSphere Commerce V5.5의 추적 구성요소입니다.

표 6.

추적 구성요소	WebSphere JRAS 확장자 추적 로그 작성 프로그램
SERVER	com.ibm.websphere.commerce.WC_SERVER
CATALOG	com.ibm.websphere.commerce.WC_CATALOG
DATASOURCE	com.ibm.websphere.commerce.WC_DATASOURCE
ORDER	com.ibm.websphere.commerce.WC_ORDER
USER	com.ibm.websphere.commerce.WC_USER
COMMAND	com.ibm.websphere.commerce.WC_COMMAND
CF	com.ibm.websphere.commerce.WC_CF
NEGOTIATION	com.ibm.websphere.commerce.WC_NEGOTIATION
RAS	com.ibm.websphere.commerce.WC_RAS
DB	com.ibm.websphere.commerce.WC_DB
METAPHOR	com.ibm.websphere.commerce.WC_METAPHOR
SCHEDULER	com.ibm.websphere.commerce.WC_SCHEDULER
DEVTOOLS	com.ibm.websphere.commerce.WC_DEVTOOLS
TOOLSFRAMEWORK	com.ibm.websphere.commerce.WC_TOOLSFRAMEWORK
RULESYSTEM	com.ibm.websphere.commerce.WC_RULESYSTEM
MERCHANDISING	com.ibm.websphere.commerce.WC_MERCHANDISING
MARKETING	com.ibm.websphere.commerce.WC_MARKETING

표 6. (계속)

REPORTING	com.ibm.websphere.commerce.WC_REPORTING
TRANSPORT_ADAPTER	com.ibm.websphere.commerce.WC_TRANSPORT_ADAPTER
PERFMONITOR	com.ibm.websphere.commerce.WC_PERFMONITOR
MESSAGING	com.ibm.websphere.commerce.WC_MESSAGING
STOREOPERATIONS	com.ibm.websphere.commerce.WC_STOREOPERATIONS
CACHE	com.ibm.websphere.commerce.WC_CACHE
EVENT	com.ibm.websphere.commerce.WC_EVENT
EJB	com.ibm.websphere.commerce.WC_EJB
CURRENCY	com.ibm.websphere.commerce.WC_CURRENCY
CATALOGTOOL	com.ibm.websphere.commerce.WC_CATALOGTOOL
CONTRACT	com.ibm.websphere.commerce.WC_CONTRACT
UBF	com.ibm.websphere.commerce.WC_UBF
BI	com.ibm.websphere.commerce.WC_BI
INVENTORY	com.ibm.websphere.commerce.WC_INVENTORY
UTF	com.ibm.websphere.commerce.WC_UTF
RFQ	com.ibm.websphere.commerce.WC_RFQ
EXCHANGE	com.ibm.websphere.commerce.WC_EXCHANGE
ACCESSCONTROL	com.ibm.websphere.commerce.WC_ACCESSCONTROL
AC_UNITTEST	com.ibm.websphere.commerce.WC_AC_UNITTEST
APPROVAL	com.ibm.websphere.commerce.WC_APPROVAL
COLLABORATION	com.ibm.websphere.commerce.WC_COLLABORATION
THREAD	com.ibm.websphere.commerce.WC_THREAD
SENSITIVE_INFO	com.ibm.websphere.commerce.WC_SENSITIVE_INFO

**JCA 커넥터 추적 구성요소:** 다음은 WebSphere Commerce V5.5의 JCA 커넥터 추적 구성요소입니다.

표 7.

추적 구성요소	WebSphere JRAS 확장자 추적 로그 작성 프로그램
JCA 전자 우편 커넥터	com.ibm.websphere.commerce.jcaemail
JCA 파일 커넥터	com.ibm.websphere.commerce.jcafile
JCA JMS 커넥터	com.ibm.websphere.commerce.jcajms
JCA 견본 커넥터	com.ibm.websphere.commerce.jcasample

**Commerce Payments 추적 구성요소:** 다음은 WebSphere Commerce V5.5의 Commerce Payments 추적 구성요소입니다.

표 8.

추적 구성요소	WebSphere JRAS 확장자 추적 로그 작성 프로그램
* (모든 것)	com.ibm.websphere.commerce.payments.*
MPF( Payments 프레임워크)	com.ibm.websphere.commerce.payments.MPF

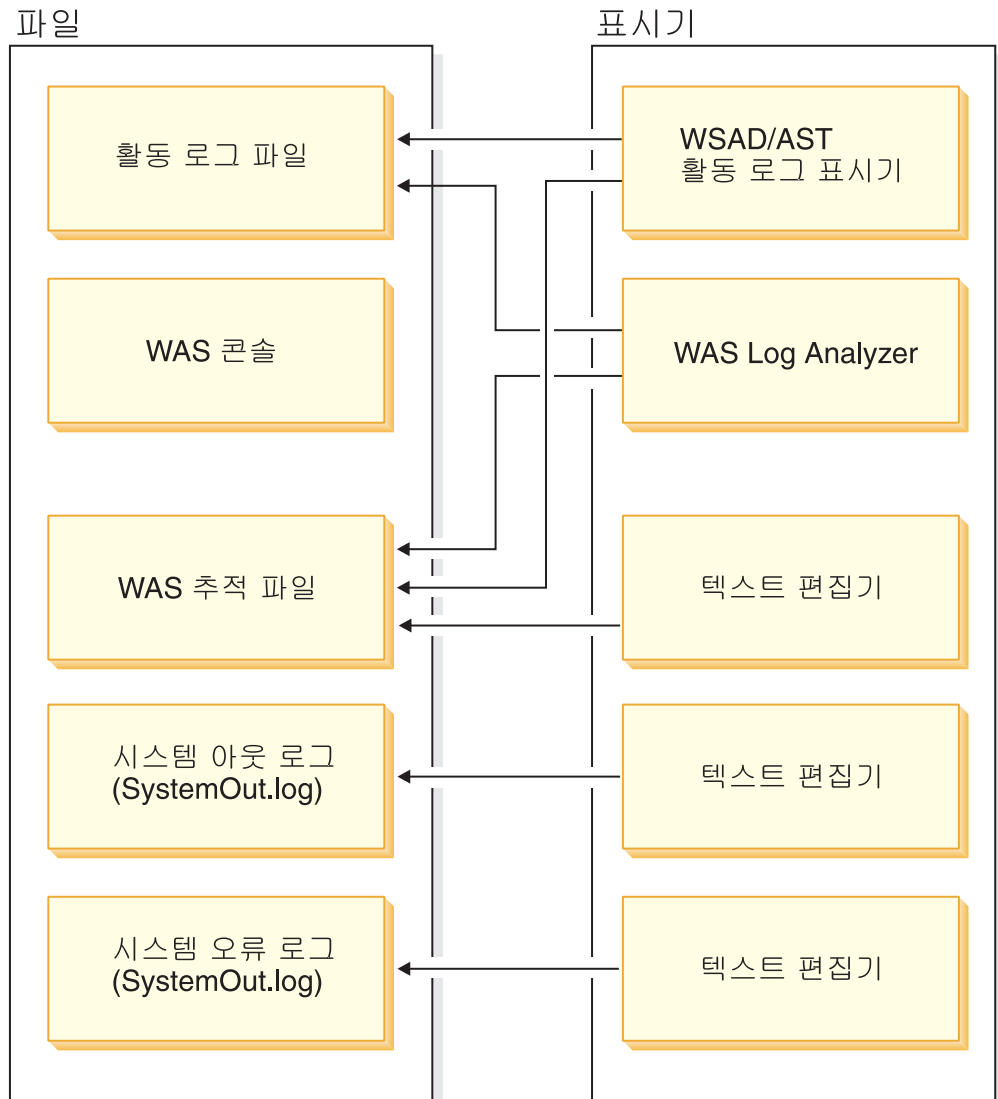
표 8. (계속)

MPFUI( Payments UI)	com.ibm.websphere.commerce.payments.MPFUI
<Cassette_name>(이것은 모든 IBM 또는 타사의 카세트 추적을 가능케 합니다. WebSphere Commerce와 함께 제공되는 카세트는 VisaNet, Paymentech, BankServACH, OfflineCard, 및 CustomOffline입니다.)	com.ibm.websphere.commerce.payments.<Cassette_name>

주: WebSphere Commerce Payments에 대한 추가 정보는 WebSphere Commerce Payments 라이브러리를 참조하십시오.

### 로그 파일 보기

다음 다이어그램은 로그 파일을 볼 수 있는 다양한 방법을 보여줍니다.





## 로그 분석기 사용




WebSphere Application Server의 일부인 로그 분석기는 하나 이상의 서비스 또는 활동 로드로 모든 데이터를 병합하고 항목을 표시합니다. 이 도구는 증상 데이터베이스를 근거로 로그 항목내의 이벤트 또는 오류 조건을 분석하고 해석하여 문제점 진단을 도와 줍니다. 로그 분석기는 IBM 웹 사이트에서 최신 증상 데이터베이스를 다운로드할 수 있는 특별 기능을 가지고 있습니다. 증상 데이터베이스에 대한 최신 갱신을 다운로드하려면 로그 분석기 인터페이스에서 파일 -> 데이터베이스 갱신 -> WebSphere Application Server의 WebSphere Application Server 증상 데이터베이스 옵션 또는 WebSphere Application Server Network Deployment의 경우 WebSphere Application Server Network Deployment 증상 데이터베이스 옵션을 사용하십시오.

**서비스 및 활동 로그:** WebSphere Application Server 서버는 다양한 WebSphere Application Server 구성요소의 활동에서 서비스 또는 활동 로그를 작성합니다. 서비스 또는 활동 로그 파일 activity.log은 installation\_directory/logs/activity 로그에 있는 2진 파일입니다. 로그 분석기는 서비스 또는 활동 로그 파일을 보는 데 사용되며, 서비스 또는 활동 로그 파일을 하나의 로그 파일로 병합할 수 있습니다. 서비스 또는 활동 로그는 텍스트 편집기를 사용하여 볼 수 없습니다. 로그 분석기 도구는 이 파일을 보기 위해 설계되었습니다. activity.log에 대한 추가 정보는 *WebSphere Application Server 시스템 관리 안내서*를 참조하십시오.

**로그 분석기를 사용한 서비스 및 활동 로그 보기:** 로그 분석기를 사용하여 서비스 또는 활동 로그를 보려면 다음을 수행하십시오.

1. WASinstall\_dir/bin 디렉토리로 이동하십시오.

2. waslogbr 스크립트 파일을 실행하십시오.

-  v waslogbr.bat.
-   v waslogbr.

그러면 로그 분석기 인터페이스가 시작됩니다.

3. 파일 > 열기를 선택하십시오.

4. 서비스 또는 활동 로그 파일을 포함하고 있는 디렉토리로 이동하십시오.

5. 서비스 또는 활동 로그 파일을 선택하고 열기를 누르십시오.

6. 레코드를 분석하려면, 왼쪽의 트리에서 레코드를 누르고, 오른쪽-클릭 메뉴에서 **UnitOfWorkView**를 선택하고 분석을 선택하십시오.

이제 옆에 녹색 확인 표시가 있는 모든 레코드는 증상 데이터베이스의 레코드와 일치합니다. 확인 표시된 레코드를 선택하면 오른쪽 하단 분할창에 문제점에 대한 설명이 표시됩니다. 로그 분석기 사용자 인터페이스에 대한 추가 정보는 WebSphere Application Server 정보 센터를 참조하십시오.



## 텍스트 파일 사용

텍스트 편집기를 사용하여 로그 파일을 볼 수 있습니다. 로그 파일은 WebSphere Commerce Server가 설치된 logs 디렉토리에서 찾을 수 있습니다. 일반적으로 ../appserver/logs/ 디렉토리에 있지만 변경할 수 있습니다. 다음은 로그 파일을 찾기 위해 사용할 수 있는 경로의 예입니다.

▶ 400 ▶ AIX ▶ Windows ▶ Linux ▶ Solaris

WAS\_installdir/logs/WC\_instanceName/trace.log

주: ▶ AIX 작성된 WebSphere Commerce 로그 파일을 보려면 vi가 아닌 다른 텍스트 편집기를 사용하십시오. vi에서는 행이 너무 길게 표시되므로 제대로 볼 수가 없습니다.

## LDAP 구성

WebSphere Commerce를 설치하면, LDAP 서버를 구성원 저장소로 사용하도록 선택하거나, 데이터베이스를 구성원 저장소로 사용하여 시작한 후 나중에 LDAP 서버를 사용하도록 전환할 수 있습니다. LDAP 구성에 대한 추가 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*에 있는 "LDAP용 WebSphere Commerce 구성"을 참조하십시오.

## 규칙 서비스 구성

규칙 서비스를 구성하려면 관리 콘솔에 로그인하고 규칙 서비스 메뉴에서 관리를 누르십시오. 사이트 운영자는 관리 콘솔을 사용하여 다음 규칙 서비스 태스크를 수행할 수 있습니다.

- 규칙 서비스 추가
- 규칙 서비스 편집
- 규칙 서비스 상태 확인
- 규칙 서비스를 최신 정보로 고침
- 규칙 서비스 삭제
- 규칙 서비스 시작
- 규칙 서비스 중지

규칙 서비스 구성 태스크에 대한 추가 정보는 WebSphere Commerce 온라인 도움말을 참조하십시오.

## 스테이징 서버 구성

### 스테이징 서버 유틸리티 사용

스테이징 유틸리티는 스테이징 서버에 있고 표준 DB2/JDBC 통신을 사용하여 프로덕션 및 스테이징 데이터베이스에 연결합니다. 운영자는 DB2 클라이언트/서버 환경이 제

대로 설정되었는지 확인해야 합니다. DB2 클라이언트는 TCP/IP 포트를 통해 DB2 서버와 통신합니다. 번호는 DB2 인스턴스 변수 SVCENAME에 설정됩니다. 이 포트 번호를 알고 있어야 하고 프록시가 이 포트를 방해하지 않는지 확인해야 합니다. 기본적으로, DB2는 50000을 사용합니다. JDBC 및 DB2 클라이언트/서버 환경 설정에 대한 추가 정보는 *DB2 관리 안내서*를 참조하십시오.

환경을 설정하고 데이터를 스테이징 서버에서 프로덕션 서버로 복사할 준비가 되면, 스테이징 전달 유틸리티를 사용하여 데이터를 푸쉬하십시오. 복사할 수 있는 테이블은 복사 테이블에 표시되어 있습니다. 사용자 정의된 테이블용 스테이징 서버를 구성한 후 사용자 정의 데이터를 저장하기 위해 작성했던 모든 새 테이블도 스테이징 서버로 복사할 수 있습니다.

STAGLOG 테이블은 내부 로그처럼 작동합니다. 스테이징 서버의 테이블에 있는 레코드를 변경할 때마다 트리거는 STAGLOG 테이블에 이 변경사항을 기록합니다. 수정된 각 레코드에 대해 트리거는 수정 유형(삽입, 삭제 또는 갱신), 레코드가 있는 테이블 이름, 레코드의 1차 키 또는 고유 색인을 기록합니다. 스테이징 서버에서 데이터베이스 레코드의 변경 및 테스트를 완료하면 스테이징 전달 유틸리티를 사용하여 프로덕션 서버로 변경사항을 다시 전달하십시오.

스테이징 서버가 관리하는 데이터베이스 테이블은 스테이징 세션 중에 프로덕션 데이터베이스에서 갱신될 수 없습니다. 스테이징 복사 유틸리티를 사용하여 프로덕션 데이터베이스를 스테이징 데이터베이스로 복사하면 스테이징 세션이 시작됩니다. 스테이징 세션은 스테이징 복사 유틸리티를 사용하여 다른 스테이징 세션을 시작할 때 종료됩니다.

프로덕션 데이터베이스에서 스테이징 데이터베이스로 데이터가 복사된 후, 스테이징 서버가 다루는 테이블에 대해서는 스테이징 데이터베이스와 프로덕션 데이터베이스가 동기화됩니다. 테이블이 동기화된 이후에는 프로덕션 데이터베이스의 테이블은 변경할 수 없습니다. 스테이징 데이터베이스를 갱신한 후 스테이징 전달 유틸리티를 사용하여 프로덕션 데이터베이스로 변경사항을 전달할 수만 있습니다. 두 데이터베이스를 모두 갱신하면, 잠재적인 키 충돌이나 참조 무결성 위반으로 인해 전달에 실패할 수 있습니다. 스테이징 세션 중에 프로덕션 데이터베이스를 갱신해야 하면 스테이징 복사 유틸리티를 사용하여 데이터베이스를 동기화하고 새 스테이징 세션을 시작하십시오.

스테이징 세션 중에 프로덕션 데이터베이스의 테이블이 갱신되지 않도록 하려면, 테이블을 사이트 운영자만 제어할 수 있어야 합니다. 스테이징 복사를 통해 데이터를 복사한 후 고객이나 판매자가 개별적으로 프로덕션 데이터베이스의 스테이징 테이블을 갱신하는 경우도 있습니다. 예를 들어, 프로덕션 데이터베이스로 복사한 후에는 판매자가 프로덕션 데이터베이스의 OFFER 테이블을 수정하지 못하도록 할 수는 없습니다. 이 경우에는 스테이징 서버를 사용할 수 없습니다. 그러나, RFQ 오브젝트는 예외입니다. 프로덕션 데이터베이스에 RFQ 오브젝트를 작성하면 프로덕션 데이터베이스의 거래 테이블에 행이 삽입됩니다. 스테이징 데이터베이스에 장기 구매 계약을 작성해도 스테이징

데이터베이스의 거래 테이블에 행이 삽입됩니다. 이 경우, 스테이징 데이터베이스와 프로덕션 데이터베이스에서 같은 테이블을 갱신하게 됩니다.

RFQ 오브젝트를 사용할 경우, 스테이징 서버에 대한 몇 가지 제한 사항이 있습니다. 스테이징 전달 유틸리티를 실행하여 변경사항을 프로덕션 데이터베이스로 전달하기 전에 스테이징 확인 유틸리티를 사용하여 충돌 가능성이 있는 고유 색인 키를 찾아야 합니다.

일반적인 B2C 사이트에서 테이블을 구성 데이터와 조작 데이터의 두 그룹으로 나눌 수 있습니다. 구성 테이블에는 상점, 카탈로그, 카탈로그 항목, 언어, 세금 및 할인과 같은 데이터가 수록됩니다. 이 테이블은 사이트 운영자가 제어하게 됩니다. 각 고객은 이 테이블을 수정할 수 없습니다. 조작 테이블에는 고객 정보, 주소, 주문, SET 관련 데이터와 같은 데이터가 수록됩니다. 고객은 조작 테이블을 수정할 수 있습니다. 스테이징 서버는 구성 테이블만을 관리합니다. 스테이징 서버가 다루는 테이블 목록에 대해서는 WebSphere Commerce 스테이징 테이블을 참조하십시오.

또한, 스테이징 서버가 다루는 테이블에 조작 테이블에 대한 외부 키 참조가 없는지도 확인해야 합니다. 그렇지 않으면, 프로덕션 데이터베이스에서 1차 키가 삭제되었을 가능성이 있기 때문에 전달에 실패할 수 있습니다. 스테이징 서버를 사용하기 전에 카탈로그 운영자와 같은 개별 사용자가 아닌 조직에서만 운영 데이터를 소유하는지 확인해야 합니다.

스테이징 서버를 사용하기 전에 다음 사항에 주의하십시오.

- 스테이징 레코드가 참조하는 모든 새 이미지 파일, HTML 파일 또는 JSP 파일은 스테이징 서버에서 프로덕션 서버로 직접 복사되어야 합니다.
- 스테이징 서버는 데이터베이스 스키마 변경사항, 이미지 파일, HTML 파일 또는 JSP 파일을 복사하고 전달할 수 없습니다. 예를 들어, 스테이징 데이터베이스에서 새 색인 또는 테이블을 작성하는 경우, 프로덕션 데이터베이스에서 색인 또는 테이블을 수동으로 작성해야 합니다.
- 로더 패키지(로드 모드) 또는 DB2 로드 유틸리티는 스테이징 트리거를 무시하기 때문에 스테이징 전달 유틸리티가 이 두 방법으로 로드된 레코드를 전달할 수 없습니다. 이 유틸리티를 사용한 경우, 스테이징 복사를 사용하여 데이터베이스 테이블을 재동기화하고 새로운 스테이징 세션을 시작하십시오. 스테이징 세션 중에 스테이징 데이터베이스나 프로덕션 데이터베이스에서 로더 패키지(로드 모드) 또는 DB2 로드를 사용해서는 안 됩니다.
- 스테이징 복사 유틸리티를 사용한 후에는 스테이징 서버를 중지하고 다시 시작해야 합니다.
- 스테이징 서버는 DB2 Text Extender를 지원하지 않습니다.
- STAGLOG 테이블을 정리할 때를 제외하고는 스테이징 서버에서 데이터베이스 정리 유틸리티를 실행하지 마십시오.

- 특정 스테이징 서버 제한 사항이 있습니다. 스테이징 서버를 사용하려면 먼저 스테이징 서버 제한 사항을 이해해야 합니다. 추가 정보는 40 페이지의 『스테이징 서버 제한 사항』을 참조하십시오.

**스테이징 복사 유틸리티:** 스테이징 복사 유틸리티인 iSeries용 stagingCopy.sh는 프로덕션 데이터베이스에서 스테이징 데이터베이스로 데이터를 복사합니다. 사이트 관련 테이블, 판매자 관련 테이블 또는 개별 테이블의 데이터를 복사할 수 있습니다. 스테이징 복사 유틸리티를 사용하기 전에 명령 구문에서 cleanup\_stage\_db 매개변수를 사용하여 스테이징 데이터베이스를 정리할 수도 있습니다. yes를 지정하면, 스테이징 복사 유틸리티가 데이터를 복사하기 전에 스테이징 테이블을 모두 정리합니다. 이는 연속 삭제를 수행하므로 다른 테이블에도 영향을 미칠 수도 있습니다. no를 지정하면, 스테이징 복사 유틸리티가 스테이징 테이블에서 아무 것도 삭제하지 않습니다. 복사 데이터가 충돌을 발생시키거나 1차 키의 키나 고유 색인을 중복시킬 경우, 복사에 실패할 수 있습니다. 스테이징 복사 유틸리티를 사용하여 스테이징 데이터베이스만 정리하려면 only를 지정하십시오.

스테이징 복사 유틸리티 및 스테이징 전달 유틸리티는 데이터베이스 데이터를 사이트 관련 레벨 및 판매자 관련 레벨의 두 가지 범위 레벨로 나눕니다. 이 사이트 범위에는 시스템의 모든 판매자에게 공통된 데이터가 포함됩니다. 예를 들면, 시스템이 사용하는 언어 및 국가 또는 지역 코드와 같습니다. 판매자 범위에는 각 판매자 관련 데이터가 들어 있습니다. 예를 들어, 상점 정보는 각 판매자별로 사용자 정의되며 상점 테이블의 행은 각 판매자별로 특정화될 수 있습니다. 어떤 데이터베이스 테이블에는 사이트 정보와 판매자 정보가 모두 들어 있습니다. 스테이징 복사 유틸리티 실행 중에 scope 매개변수를 \_all\_로 지정하면, 이 사이트 데이터가 복사된 다음에 모든 판매자 데이터가 복사됩니다. scope 매개변수를 \_site\_로 지정하면, 사이트 데이터만 복사됩니다. scope를 \_merchant\_로 지정하면, 판매자 데이터만 복사됩니다. 각 판매자별로는 데이터를 복사할 수 없고 모든 판매자에 대해서만 데이터를 복사할 수 있습니다. scope를 \_all\_로 설정하지 않는 경우, 모든 판매자가 사이트 데이터를 사용하므로 판매자 데이터를 복사하기 전에 사이트 데이터를 복사하십시오. 그렇지 않으면 외부 키와 1차 키가 일치하지 않아서 복사에 실패하게 됩니다. cleanup\_stage\_db를 사용하여 사이트 데이터를 정리하는 경우, 계단식 삭제로 인해 판매자 데이터도 삭제될 수 있음에 주의하십시오. scope를 \_all\_로 설정하지 않는 경우, 판매자 데이터를 정리한 후 사이트 데이터를 정리한 다음, 사이트 데이터를 복사하고 나서 판매자 데이터를 복사해야 합니다.

스테이징 복사 유틸리티에서 사용할 수 있는 또 다른 옵션은 script\_file 매개변수입니다. 스크립트 파일 이름을 지정하여 지정된 범위에 따라 프로덕션 데이터베이스를 스테이징 데이터베이스로 복사하기 위해 스테이징 복사 유틸리티는 반출 및 반입에 사용하는 SQL 스크립트 파일을 작성합니다. 기본값을 사용하거나 cleanup\_stage\_db를 yes로 지정하는 경우, Delete 문도 작성되어 스테이징 데이터베이스를 정리합니다. 스크립트 파일은 스테이징 복사 유틸리티를 시작한 디렉토리에 있습니다. 스크립트 파일은 반출 및 반입을 사용하여 데이터베이스 복사 처리 속도를 빠르게 합니다. 작성된 스

크립트 파일을 변경하여 스테이징 복사 유틸리티의 행위를 수정할 수도 있습니다. 예를 들어, 반입 유틸리티 대신 DB2 로드 유틸리티를 사용하도록 스크립트를 변경하여 복사 처리를 보다 빠르게 할 수 있습니다. 작성된 스크립트는 모든 테이블을 유틸리티를 실행한 디렉토리로 반출한다는 점에 주의하십시오. 충분한 디스크 공간이 있는지 확인하십시오.

트랜잭션 범위를 이해하는 것은 매우 중요합니다. 스테이징 복사 유틸리티는 스테이징 데이터베이스를 정리할 때 각 테이블을 정리한 다음 트랜잭션을 확약합니다. 데이터를 복사할 때, 스테이징 복사 유틸리티는 각 테이블을 복사하고 KEYS 테이블을 동기화한 다음 트랜잭션을 확약합니다. 작성된 스크립트의 경우, 트랜잭션 범위는 DB2 반입 유틸리티 때문에 약간 다릅니다. DB2 반입 유틸리티는 반입 완료 후 자동으로 트랜잭션을 확약합니다. KEYS 테이블을 동기화하기 전에 트랜잭션을 확약합니다. 따라서, KEYS 테이블의 동기화는 별도의 트랜잭션에서 수행됩니다.

dbtable 매개변수를 사용하여 정리하거나 복사할 테이블을 지정할 수 있습니다. 정리하거나 복사할 테이블을 지정할 때 테이블을 분리할 수 없습니다. 어떤 테이블은 참조 제한조건에 의해 다른 테이블과 연관되어 있습니다. 지정된 테이블을 정리하면 연속 삭제에 의해 하위 테이블도 정리됩니다. 지정된 테이블을 복사하면 먼저 상위 테이블을 복사해야 합니다. 그렇지 않으면 정리 또는 복사에 실패하게 됩니다.

스테이징 복사 유틸리티는 구성 및 확장 가능합니다. 사용자 정의된 테이블을 처리하려면 테이블을 충족시키는 몇 가지 조건이 있어야 하고 스테이징 구성 테이블을 설정해야 합니다. 스테이징 복사 유틸리티를 사용하려면 먼저 데이터베이스 구성에 있는 단계를 따라야 합니다. 명령이 성공할 경우, 스테이징 복사 유틸리티는 STAGLOG 테이블의 모든 레코드를 삭제합니다.

주: 시스템에 RFQ가 있으면 스테이징 복사 명령을 사용할 수 없습니다. 자세한 정보는 40 페이지의 『스테이징 서버 제한 사항』을 참조하십시오.

**스테이징 확인 유틸리티:** 구성 및 조작 데이터가 같은 테이블을 공유할 경우, 스테이징 데이터베이스와 프로덕션 데이터베이스 간에 고유 색인 키 충돌이 발생할 수 있습니다. 변경사항을 프로덕션 데이터베이스로 전파하기 전에 스테이징 확인 명령 또는 iSeries용 stagingCheck.sh를 사용하여 고유 색인 충돌이 발생할 가능성이 있는지 판별하여 전파하기 전에 충돌을 해결하십시오.

프로덕션 데이터베이스에서 RFQ를 사용 중이고 스테이징 데이터베이스에서 장기 구매 계약을 작성 중이면, 두 데이터베이스에서 같은 테이블을 갱신하고 있는 것입니다. 예를 들어, 사이트 운영자는 스테이징 데이터베이스에서 장기 구매 계약을 작성하고, 이로 인해 스테이징 데이터베이스의 TRADING 테이블(및 다른 테이블)에 한 행이 삽입됩니다. 동시에 사용자가 프로덕션 데이터베이스에 RFQ를 작성합니다. 그러면 프로덕션 데이터베이스의 TRADING 테이블(및 다른 테이블)에 한 행이 삽입됩니다. 두 행이 TRADING 테이블에서 동일한 고유 색인 값을 갖게 될 수 있습니다. 스테이징 데이터



베이스에서 프로덕션 데이터베이스로 장기 구매 계약을 전달하면 고유 색인 키가 충돌하여 전달에 실패하게 됩니다. 전달하기 전에 스테이징 확인 유틸리티를 사용하여 충돌하는 고유 색인 키를 찾은 다음 이를 수정하십시오. 그런 다음, 변경사항을 전달할 수 있습니다.

스테이징 확인 유틸리티를 사용할 때, `-scope` 매개변수를 `_unique_index_`로 지정하여 스테이징 데이터베이스에서 델타 키 변경사항에 대해 키 충돌이 발생할 수 있는지 확인하십시오. 모든 삽입 및 갱신 조작에 대해 `STGUINDTAB` 테이블에 지정된 모든 테이블에 대해 색인 키 충돌 가능성이 있는지를 확인합니다. 각 테이블에 대해서는 모든 고유 색인을 검토하여 프로덕션 데이터베이스와 스테이징 데이터베이스 간에 키 충돌 가능성이 있는지 확인합니다. 키 충돌 가능성이 있으면 테이블 이름, 고유 색인 및 충돌된 키 값을 보고합니다.

스테이징 확인 명령은 데이터베이스를 변경하지 않고 해결해야 하는 키 충돌 가능성을 보고합니다. 이 명령을 사용할 때, 스테이징 데이터베이스로 `-sourcedb` 매개변수를 지정하십시오. 스테이징 확인 명령은 프로덕션 데이터베이스를 소스 데이터베이스로 지정하면 제대로 작동하지 않습니다.

스테이징 확인 유틸리티는 구성 및 확장이 가능합니다. 여러 개의 테이블 또는 조정된 테이블을 `STGUINDTAB` 테이블에 추가하고 키 충돌 가능성이 있는지를 검증하기 위해 스테이징 확인 명령을 실행할 수 있습니다.

**주:** 항상 구성과 조작 데이터가 같은 테이블을 공유하지 않도록 하십시오.

**스테이징 전달 유틸리티:** 스테이징 서버에서 데이터베이스 레코드를 변경하고 테스트한 후, 고유 색인 키 충돌이 발생할 수 있는지 확인하고 스테이징 확인 유틸리티를 사용하여 충돌을 해결하십시오. 이제 변경사항을 프로덕션 데이터베이스로 전달할 준비가 되었습니다.

스테이징 전달 유틸리티 또는 iSeries용 `stagingProp.sh`는 스테이징 데이터베이스에서 프로덕션 데이터베이스로 변경사항을 이동시킵니다. 스테이징 전달 유틸리티는 `STAGLOG` 테이블을 사용하여 스테이징 데이터베이스에서 변경된 레코드를 식별하고, 프로덕션 데이터베이스에서 해당 레코드를 갱신합니다. 처리된 레코드는 `STAGLOG` 테이블의 `STGPROCESSED` 열에서 1로 표시됩니다.

`scope` 매개변수를 지정하여 전달할 데이터 유형을 선택할 수 있습니다. `_site_`로 설정하면 변경된 모든 사이트 데이터가 스테이징 데이터베이스에서 프로덕션 데이터베이스로 전달됩니다. `_merchant_`로 설정하면 모든 판매자의 변경된 데이터가 전달됩니다. 개별 판매자 데이터를 전달할 수는 없습니다. `_all_`로 설정하면 사이트 및 판매자 데이터가 전달됩니다.

`dbtable` 매개변수를 사용하면 특정 테이블을 전달할 수 있습니다. 테이블을 지정하기 전에 상위 테이블이 전달되었는지 확인하십시오.

스테이징 전달 유틸리티에 대한 트랜잭션 범위는 스테이징 복사 유틸리티와 다릅니다. 스테이징 전달 유틸리티를 실행할 때마다 하나의 트랜잭션으로 계산됩니다. 예를 들어, scope를 `_site_`로 지정하면, 스테이징 전달 유틸리티가 수정된 모든 사이트 데이터에 대해 새 트랜잭션을 시작하고 전달이 완료된 후에 트랜잭션을 확약합니다. 전달에 실패하면 전달이 롤백되며 프로덕션 데이터베이스의 상태가 전과 같아집니다.

스테이징 전달 유틸리티는 구성 및 확장 가능합니다. 사용자 정의된 테이블을 전달하려면 이 테이블이 특정 조건을 충족해야 합니다. 자세한 내용은 사용자 정의 테이블에 대한 스테이징 서버 구성을 참조하십시오. 스테이징 전달 유틸리티를 사용하려면 먼저 데이터베이스 구성에 있는 단계를 따라야 합니다.

## 데이터베이스 구성






스테이징 복사 유틸리티, 스테이징 전과 유틸리티 또는 데이터베이스 정리 유틸리티를 사용하기 전에 다음을 수행할 수 있습니다.

주:

1. 다음 단계는 단지 제안사항이며 스테이징 복사 유틸리티에 문제점이 있는 경우에만 사용해야 합니다.

2. 다음은 `> 400`에 의해 표시되지 않으면 iSeries에 적용되지 않습니다.

1. PATH 환경 변수를 설정하십시오.

2.      DB2 데이터베이스를 사용하고 있다면, 다음 명령을 실행하여 스테이징 데이터베이스 및 프로덕션 데이터베이스를 구성하십시오.

```
db2 update db config for db_name using LOGPRIMARY 80
db2 update db config for db_name using LOGBUFSZ 512
db2 update db config for db_name using DBHEAP 2048
db2 update db config for db_name using APPLHEAPSZ 2048
db2 update db config for db_name using PCKCACHESZ 8200
```

여기서, `db_name`은 데이터베이스의 이름입니다.





주:

a. 기본 STMTHEAP 크기는 60000입니다.

b. 기본 LOCKLIST는 2400입니다.

c. 기본 STAT\_HEAP\_SZ는 2048입니다.

d. 기본 APP\_CTL\_HEAP\_SZ는 4096입니다.

3.     성능을 향상시키려면 버퍼DB2 데이터베이스 크기 및 사용 가능한 메모리에 기초하여 최적 버퍼 풀 크기를 결정하십시오. 다음 명령을 실행하여 기본 버퍼 풀 크기를 변경하십시오

```
db2 connect to db_name
db2 alter bufferpool IBMDEFAULTBP size n
db2 terminate
```

여기서, *n*은 최적 버퍼 풀 크기입니다.

4. **400** secofr 권한이 있고 65535 이외의 다른 *cssid*를 갖는 사용자 프로파일로 로그인하십시오.

## 사용자 정의 테이블의 스테이징 서버 구성

사용자 정의 데이터베이스 테이블에서 스테이징 서버를 사용하려면 다음과 같이 구성하십시오.

1. 사용자 정의 테이블 범위(사이트 데이터, 판매자 데이터 또는 사이트 및 판매자 데이터)를 식별하십시오.
2. 테이블 범위 및 색인 유형에 기초하여 해당 트리거 예를 사용하여 데이터베이스 테이블에 대한 트리거를 작성하십시오.
3. 사용자 정의 테이블을 STGSITETAB, STGMERTAB 및 STGMRSTTAB 테이블에 삽입하십시오.
  - 사이트 테이블의 경우, STGSITETAB에만 삽입하십시오.
  - 판매자 테이블의 경우, STGMERTAB에만 삽입하십시오.
  - 사이트 및 판매자 데이터가 모두 들어 있는 테이블의 경우, STGSITETAB, STGMERTAB 및 STGMRSTTAB에 삽입하십시오.

주: 모든 상위 테이블이 제대로 삽입되고 상위 테이블의 TABNBR 열이 하위 테이블의 열보다 작은지 확인해야 합니다. 사용자 정의 테이블이 WebSphere Commerce 테이블의 상위 테이블인 경우, 테이블의 TABNBR 열이 하위 테이블의 해당 열보다 작을지도 확인해야 합니다.

## 연결 구성

다양한 어댑터 구성 정보는 *WebSphere Commerce 추가 소프트웨어 안내서* 및 *WebSphere Commerce Production 온라인 도움말*을 참조하십시오.

## WebSphere Commerce Payments 구성

다음을 수행하여 Payments를 구성할 수 있습니다.

- WebSphere Commerce Payments가 WebSphere Commerce 설치의 일부인지 확인하십시오.
- WebSphere Commerce Payments 인스턴스를 작성하십시오.
- 구성 관리자를 사용하여 WebSphere Commerce Payments 인스턴스에 카세트를 추가하십시오.



- WebSphere Commerce Payments 인스턴스 및 WebSphere Commerce 인스턴스를 시작하십시오.
- 판매자 및 해당 판매자에 대한 판매자 관리자를 작성하십시오.

카세트를 구성하고 Payments를 사용하려면 판매자 관리자로서 WebSphere Commerce Payments에 로그인해야 합니다. WebSphere Commerce Payments는 CustomOffline Cassette 및 OfflineCard Cassette와 같이 설치됩니다. 이들 카세트가 지원하는 최소 프레임워크는 WebSphere Commerce Payments 5.5 버전입니다. 추가 정보는 *WebSphere Commerce 설치 안내서*를 참조하십시오.

다음은 WebSphere Commerce 및 Payments 기본 설치 디렉토리입니다.

표 9. 기본 설치 디렉토리

기본 설치 디렉토리	설명
<i>WC_installdir</i>	Windows: <i>drive:\WebSphere\CommerceServer55</i>  AIX: <i>/usr/WebSphere/CommerceServer55</i>  Solaris, Linux: <i>/opt/WebSphere/CommerceServer55</i>  400: <i>/QIBM/ProdData/CommerceServer55</i>
<i>Payments_installdir</i>	Windows: <i>drive:\WebSphere\CommerceServer55\payments</i>  AIX: <i>/usr/lpp/WebSphere/CommerceServer55/payments</i>  Solaris, Linux: <i>/opt/WebSphere/CommerceServer55/payments</i>  400: <i>/QIBM/ProdData/CommercePayments/55</i>

WebSphere Commerce Payments 구성에 대한 자세한 정보는 181 페이지의 부록 C 『WebSphere Commerce Payments 학습서』를 참조하십시오.



---

## 제 4 장 호스팅된 상점 관리

사이트 운영자가 호스팅된 상점 시나리오에서 수행해야 하는 여러 가지 태스크가 있습니다. 이 장에서는 사이트 운영자가 다음 카테고리에서 수행하는 태스크를 개략적으로 설명합니다.

- 재판매자 관리
- 배포업자 관리
- 사이트 관리

---

### 재판매자 관리

호스팅된 상점 시나리오에서 사이트 운영자는 재판매자 조직을 설정하고 재판매자와의 관계를 유지보수하는 책임을 가집니다. 이 조정에는 다음 태스크가 포함됩니다.

- 새 재판매자 조직 등록의 전자 우편 승인 보내기.

재판매자가 자신의 조직을 등록하면, 재판매자에게 조직에 대한 재판매자 요청이 승인되었음을 알리는 전자우편을 보내는 것은 사이트 운영자 책임입니다. 실제 조직 등록의 승인은 조직 관리 콘솔에서 수행됩니다. 조직 관리 콘솔 및 콘솔에서 수행할 수 있는 태스크에 관한 추가 정보는 *WebSphere Commerce Production* 온라인 도움말을 참조하십시오.

재판매자 조직이 승인되면 수요 체인 비즈니스 모델의 재판매자는 *WebSphere Commerce* 액셀러레이터를 사용하여 상점을 수정할 수 있습니다. 각 상점을 종료 상태로 작성되지만 재판매자는 *WebSphere Commerce* 액셀러레이터를 사용하여 상점을 갱신 또는 변경하여 결국 열 수 있습니다. 호스팅 비즈니스 모델에서 각 상점은 보류 상태로 작성되며 사이트 운영자 또는 채널 관리자가 다시 시작해야 합니다. 이 경우 판매자는 상점이 다시 시작될 때까지는 상점을 변경할 수 없습니다. 비즈니스 모델에 대한 추가 정보는 *WebSphere Commerce 기본 정보* 및 *WebSphere Commerce 상점 개발 안내서*를 참조하십시오. *WebSphere Commerce* 액셀러레이터에 대한 추가 정보는 *WebSphere Commerce Production* 온라인 도움말을 참조하십시오.

- 재판매자 호스팅된 상점 승인 보내기(호스팅 비즈니스 모델을 사용하는 상점에만 해당).

일단 호스팅된 상점이 설정되면 사이트 운영자는 호스팅된 상점 요청 승인이 포함된 전자 우편을 보냅니다. 이 전자 우편에는 또한 새 호스팅된 상점에 대한 링크와 호스팅된 상점 관리 방법에 대한 지시사항이 포함됩니다.

주: 재판매자 조직 관리에 관한 추가 정보는 *WebSphere Commerce Production* 온라인 도움말의 "재판매자 조직 관리"를 참조하십시오.

---

## 배급업자 관리

호스팅된 상점 시나리오에서 사이트 운영자는 배급업자 설정 및 재판매자와의 배급업자 관계를 작성할 책임이 있습니다. 이 관리에는 다음 태스크가 포함됩니다.

- 새 배급업자 작성.

사이트 운영자는 배급업자 서비스 계약을 작성하여 새 배급업자를 작성할 수 있습니다. 계약에 추가할 정보는 IBM에 직접 문의하여 얻으실 수 있습니다. 배급업자 서비스 계약에 대한 추가 정보는 WebSphere Commerce Production 온라인 도움말을 참조하십시오.

주: 이 태스크는 또한 채널 관리자가 수행할 수도 있습니다.

- 배급업자 설정 변경.

때때로 사이트 운영자가 배급업자에 대한 연결의 시간 종료 설정 같은 배급업자 설정의 일부를 변경하는 요청을 수신할 수 있습니다. 배급업자 설정은 SQL 문을 통해 데이터베이스에 액세스하여 변경할 수 있습니다. 추가 정보는 *WebSphere Commerce 상점 개발 안내서*를 참조하십시오.

- Adapter for CrossWorlds를 구성하고 설치하십시오.

Adapter for CrossWorlds는 호스팅된 상점을 위해 WebSphere Commerce와 외부 및 백엔드 시스템 간을 상호연결을 설정하는 데 사용되는 커넥터입니다. 사이트 운영자는 재판매자와 배급업자 사이트 간의 연결을 보장하기 위해 어댑터를 구성하고 설치할 책임이 있습니다. WebSphere MQ를 사용한 SAP R/3 백엔드 시스템을 갖는 WebSphere Commerce 사이의 사용작용도 작성할 수 있으며 이는 사이트 운영자가 설정해야 합니다. IBM CrossWorlds 및 WebSphere MQ에 대한 추가 정보는 *WebSphere Commerce 추가 소프트웨어 안내서*를 참조하십시오.

---

## 사이트 관리

호스팅된 상점에 특정한 사이트 관리 태스크는 다음을 포함합니다.

- 스테이징 서버 운영. 추가 정보는 39 페이지의 『스테이징 서버』를 참조하십시오.
- DBClean 운영. 추가 정보는 111 페이지의 『데이터베이스 정리 유틸리티』를 참조하십시오.
- WebSphere Commerce Analyzer 운영. 추가 정보는 WebSphere Commerce 설치 CD를 참조하십시오.
- WebSphere Commerce server의 시작 및 중지. 추가 정보는 13 페이지의 『WebSphere Commerce의 시작 및 중지』를 참조하십시오.

---

## 제 5 장 동적 캐시 작성

이 장에서는 WebSphere Commerce Servlet 또는 JSP(JavaServer Pages) 결과의 동적 캐시에 대해 설명합니다.

---

### WebSphere Commerce Servlet 또는 JSP 결과 캐시 작성

고객이 상품 또는 카테고리 페이지를 보기 위해 링크를 누르면 대부분의 시간이 HTTP 요청의 구문을 분석하고 데이터베이스에 액세스하여 동적으로 페이지를 작성하는 데 소요됩니다. 사이트 트래픽이 많고 데이터베이스에 많은 상품과 카테고리 항목이 있으면 Servlet 또는 JSP 파일(JavaServer Pages)을 로드하는 데 시간이 훨씬 더 걸릴 수 있습니다.

서버에서 대부분의 HTTP 요청은 카테고리 정보용입니다. WebSphere Commerce 명령인 CategoryDisplay, ProductDisplay, TopCategoriesDisplay 및 StoreCatalogDisplay는 데이터베이스에서 정보를 검색하고 JSP 페이지로 결과를 표시합니다. 카탈로그 정보가 마지막으로 본 이후에 변경되지 않은 경우, Servlet 또는 JSP 파일은 다음에 고객이 요청할 때 다시 실행할 필요가 없습니다. 캐시에 저장된 동등한 정적 Servlet 또는 JSP 파일을 제공하는 것이 더 빠릅니다. Servlet 또는 JSP의 캐시는 웹 모듈의 WEB-INF 디렉토리에 있는 cachespec.xml 파일에 cache-entry 요소를 정의하여 구성할 수 있습니다.

액세스 중인 페이지에 해당하는 캐시 항목이 메모리에 없으면 캐시 항목이 동적으로 작성됩니다. 그런 다음, 페이지가 메모리에 저장되므로 페이지 데이터가 수정될 때까지 다시 작성할 필요가 없습니다.

주: WebSphere Commerce 동적 캐시 및 URL 재작성은 상호작용할 수 없습니다. URL 재작성을 켜놓은 상태에서는 WebSphere Commerce 동적 캐시를 사용 안함으로 설정해야 하며 Servlet 또는 JSP 파일을 캐시해서는 안됩니다. URL 재작성에 대한 추가 정보는 *WebSphere Commerce 보안 안내서*의 세션 관리에 대한 장을 참조하십시오.

---

### WebSphere Application Server 동적 캐시

WebSphere Commerce는 Servlet 또는 JSP 파일 및 WebSphere Application Server CacheableCommand 인터페이스에서 확장된 명령을 캐시하기 위해 WebSphere Application Server 동적 캐시 서비스를 이용합니다. 동적 캐시 서비스, Servlet 캐시 작성 및 디스크 오프로드는 기본적으로 WebSphere Commerce 인스턴스 작성 중에 사용됩니다.

WebSphere Application Server 동적 캐시에 대한 추가 정보는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에 있는 『동적 캐시 서비스를 통한 성능 개선』을 참조하십시오.

## 동적 캐시

Servlet이나 JSP 파일 결과를 캐시 작성하면 응용프로그램 성능이 향상됩니다. WebSphere Application Server는 Servlet, 웹 서비스 및 WebSphere 명령을 포함한 여러 가지 캐시 활동을 동적 캐시라는 하나의 서비스로 통합합니다. 이들 캐시 작성 활동은 함께 작동하여 응용프로그램 성능을 향상시키고, 응용프로그램 서버의 동적 캐시 서비스에서 설정되는 많은 구성 매개변수를 공유합니다.

캐시 메모리로부터 요청에 서비스하여 Servlet과 JSP 파일의 성능을 향상시키기 위해 동적 캐시를 사용할 수 있습니다. 캐시 항목에는 Servlet 출력, Servlet 실행 결과 및 메타데이터가 들어있습니다.

추가 정보에 대해서는 WebSphere Application Server 정보 센터 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에 있는 『동적 캐시 서비스를 통한 성능 개선』 주제를 참조하십시오.

## 동적 캐시 서비스 및 Servlet 캐시 작성 사용

캐시 작성을 사용하려면 동적 캐시 서비스를 사용하고 Servlet 캐시 작성을 구성해야 합니다. 이들 단계를 수행하기 위한 정보에 대해서는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 『글로벌하게 동적 캐시 서비스 사용』 및 『Servlet 캐시 작성 구성』을 참조하십시오.

## 동적 캐시 모니터 사용

WebSphere Application Server는 캐시 통계, Edge Side 및 디스크 통계, 캐시 항목, 종속성 ID 및 캐시 정책 정보를 표시하기 위해 동적 캐시 모니터 응용프로그램을 제공합니다. WebSphere Application Server 동적 캐시의 콘텐츠와 행위를 점검하려면 WebSphere Application Server 동적 캐시 모니터를 설치해야 합니다. 동적 캐시 모니터를 사용하려면, 동적 캐시 작성을 사용하는 각 응용프로그램 서버에 `WAS_installdir` 디렉토리의 `installableApps` 서브디렉토리에 위치한 `CacheMonitor.ear` 파일을 설치하십시오.

WebSphere Commerce의 경우, 보안상의 이유로 동적 캐시 모니터를 위해 `VH_instance_name` 대신 가상 호스트 `VH_instance_name_admin`(예: `VH_demo_admin`)을 사용하도록 권장합니다. WebSphere Application Server 관리 콘솔에서 가상 호스트 이름을 추가하거나 변경할 수 있습니다.

웹 브라우저에서 다음 웹 주소를 사용하여 웹 응용프로그램에 액세스할 수 있습니다.  
`http://host_name:port/cachemonitor`

안전한 액세스를 위해 다음 관리 호스트 시스템에 액세스하는 것을 권장합니다.

`https://admin_host_name:port/cachemonitor`

예를 들어, 캐시 모니터 설치를 위해 가상 호스트 `VH_instance_name_admin`을 사용하면 캐시 모니터는 다음과 같이 액세스될 수 있습니다.

`https://admin_host_name:8002/cachemonitor`

주:

1. 모든 WebSphere Commerce 인스턴스에 대해 하나의 동적 캐시 모니터를 실행해야 합니다.
2. 캐시 모니터를 사용하려 하고 WebSphere Application Server EJB 보안이 사용 가능하게 설정된 경우에는 항상 *WebSphere Commerce* 보안 안내서의 "동적 캐시 모니터의 보안 구성" 절에 설명된 대로 몇 가지 추가적인 설정을 수행해야 합니다.

동적 캐시 모니터 설치에 대한 추가 정보는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "캐시 정보 표시"를 참조하십시오.

## WebSphere 동적 캐시 모니터에 대한 종속성 ID의 트리 보기

WebSphere Commerce는 캐시 모니터의 확장인 트리 보기 도구를 제공하는데 이 도구는 메모리에 있는 종속성 ID의 트리 보기를 제공합니다. 트리 보기 도구는 종속성 ID의 트리 보기를 제공합니다. 임의의 트리 노드를 펼치거나 접어서 계층 구조 보기를 가져올 뿐 아니라, 해당 노드 아래의 캐시 항목 그룹을 무효화할 수 있습니다. 원래 보기는 수정되지 않고 대신 새 링크가 "종속성 ID 트리 보기"로 나타납니다.

+  
+  
+

트리 보기 도구의 설치 및 사용에 대한 정보는 다음 디렉토리에 있는 WebSphere 동적 캐시 모니터에 대한 종속성 ID의 트리 보기"에 대한 README 파일을 참조하십시오.

[AIX](#) [400](#) [Linux](#) [Solaris](#) `WC_installdir/features/dynacache/CacheMonitor/`

[Windows](#) `WC_installdir\features\dynacache\CacheMonitor\`

---

## 캐시 가능 오브젝트 구성

캐시 가능 오브젝트는 웹 응용프로그램 아카이브(WAR) WEB-INF 또는 엔터프라이즈 bean WEB-INF 디렉토리에 있는 `cachespec.xml` 파일에 정의됩니다. 응용프로그램 서버 특성 디렉토리에 글로벌 `cachespec.xml` 파일을 배치할 수 있지만 권장 방법은 전개 모듈과 함께 캐시 구성 파일을 배치하는 것입니다. `cachespec.xml` 파일의 루트 요소는 `<cache-entry>` 요소를 포함하는 `<cache>`입니다.

Servlet 또는 JSP 결과 캐시 작성에 대한 캐시 항목을 지정하려면 cachespec.xml 파일에 다음 섹션을 추가하십시오.

```
+ <cache-entry>
+   <class>Servlet</class>
+   <name>name</name>
+   ...
+ </cache-entry>
```

여기서 *name*은 Servlet 또는 JSP의 상대 웹 경로 또는 Servlet 맵핑입니다.

명령 캐시 작성을 위한 캐시 항목을 지정하려면 cachespec.xml 파일에 다음 섹션을 추가하십시오.

```
+ <cache-entry>
+   <class>command</class>
+   <name>name</name>
+   ...
+ </cache-entry>
```

여기서 *name*은 명령 클래스(예: com.ibm.commerce.dynacache.commands.MemberGroupsCacheCmdImpl)의 전체 경로입니다.

<cache-entry>...</cache-entry> 요소에서 cache-ID, dependency-ID 및 무효화 규칙을 개발할 수 있습니다. 오브젝트를 캐시하려면 WebSphere Application Server가 해당 오브젝트의 다른 호출에 대해 고유한 ID를 작성할 수 있어야 합니다. <cache-id> 요소가 이 태스크를 수행합니다. 각 캐시 항목은 규칙이 비어 있지 않은 cache-ID를 리턴하거나 실행할 추가 규칙이 없을 때까지 순서대로 실행하는 복수의 cache-ID를 가질 수 있습니다. 어떤 cache-ID 작성 규칙도 유효한 캐시 ID를 작성하지 않는 경우, 오브젝트는 캐시되지 않습니다.

+ 주: 이 릴리스의 경우, 명령 캐시에 캐시 항목을 지정하면 <invalidate>...</invalidate>를 사용하고 있는 명령 invalidation만 지원됩니다(명령 캐시에 대한 <cache-id>...</cache-id> or <dependency-id>...</dependency-id> 요소를 사용할 수 없습니다).

```
+ <cache-entry>
+   <class>command</class>
+   <name>name</name>
+   <invalidate>
+   ...
+   </invalidate>
+ </cache-entry>
```

동적 캐시는 cachespec.xml 파일의 변경에 응답합니다. cachespec.xml의 새 버전이 검출될 때 이전 정책이 바뀝니다. 이전 정책 파일을 통해 캐시된 오브젝트는 캐시에서 자동으로 무효화되지 않습니다. 새 정책에서 재사용되거나 대체 알고리즘을 통해 캐시에서 제거됩니다. cachespec.xml 파일에 대한 추가 정보는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/>)



infocenter.html)의 『Cachespec.xml 파일』 주제를 참조하십시오. 또한 상점을 개발할 때의 추가 고려사항에 대해 *WebSphere Commerce* 상점 개발 안내서를 참조하십시오.

---

## 캐시 작성 기본값

설치 중에 *WebSphere Commerce*는 다음과 같은 기본값으로 캐시 시스템을 설정합니다.

- 동적 캐시가 사용됩니다.
- 디스크 오프로드가 사용됩니다.
- Servlet 캐시가 사용됩니다.
- 각 비즈니스 모델에 대한 기본 캐시 정책(샘플로 제공됨)이 다음 표시 명령에 대해 JSP 파일을 캐시합니다.
  - CategoryDisplay
  - ProductDisplay
  - StoreCatalogDisplay
  - TopCategoriesDisplayView

다음 위치에서 캐시 항목을 직접 병합해야 합니다.



`WC55_installdir/samples/dynacache/business_model/cachespec.xml`

에서 다음 위치로 병합하십시오.

`WAS_installdir/installedApps/cell_name/WC_instance_name.ear/Stores.war/WEB-INF/cachespec.xml`



`WC55_installdir\samples\dynacache\business_model\cachespec.xml`

에서 다음 위치로 병합하십시오.

`WAS_installdir\installedApps\cell_name\WC_instance_name.ear\Stores.war\WEB-INF\cachespec.xml`

상점에서 캐시 정책 사용에 대한 정보는 *WebSphere Commerce* 견본 상점 안내서 또는 *WebSphere Commerce* 상점 개발 안내서를 참조하십시오.

---

## 전체 페이지 및 단편 캐시 작성

*WebSphere Application Server* 동적 캐시가 JSP 파일을 캐시하는 방법은 JSP가 작성되는 방법을 기초로 합니다. 특정 *WebSphere Commerce* 명령의 페이지 출력이 항상 URL 매개변수와 요청 속성을 기반으로 동일한 결과를 작성하는 경우, 페이지 출력은 특성 요소인 consume-subfragments(CSF)와 함께 Servlet 이름으로 *WebSphere*

Commerce 컨트롤러 Servlet(`com.ibm.commerce.server.RequestServlet`)을 사용하여 캐시 항목과 함께 캐시할 수 있습니다. `cache-entry`가 이 방식으로 정의되면, 페이지 출력이 전체 페이지 캐시 작성이라는 방식으로 캐시됩니다. 컨트롤러 Servlet과 함께 `consume-subfragments`를 사용하는 경우의 큰 잇점은 성능이지만 이 메커니즘을 웹 페이지를 캐시하는데 사용하면 페이지 출력은 개별 정보를 가질 수 없습니다.

페이지 출력이 사용자에게 종속되는 절을 갖는 경우, 페이지 출력은 단편 캐시라는 방식으로 캐시됩니다. 즉, JSP 페이지는 별도의 캐시 항목으로 캐시되며 요청될 때 다시 합쳐집니다. 단편(JSP) 캐시의 경우, WebSphere Commerce는 명령을 실행하여 동적 캐시 메커니즘이 JSP가 캐시로부터 서비스되는지의 여부를 판별하기 전에 실행할 JSP를 판별합니다. 이러한 방법의 잇점은 유연성이며 이는 여러 캐시 항목이 사용자 정보를 기초로 페이지를 작성하기 위해 다시 합쳐질 수 있기 때문입니다.

## 전체 페이지 캐시 작성

특성 요소 `consume-subfragments(CSF)`를 사용하는 경우, 상위 항목(CSF로 표시한 항목)이 캐시 항목에 모든 단편의 모든 콘텐츠를 포함하여 포함 또는 전송은 하지 않지만 항목의 전체 트리의 콘텐츠를 갖는 하나의 큰 캐시 항목을 만듭니다.

Servlet이 캐시될 때는 해당 Servlet의 콘텐츠만이 저장됩니다. 캐시에는 그것이 포함하거나 전송하는 다른 단편에 대한 공간 보유자가 포함됩니다. `Consume-subfragments(CSF)`는 캐시에 캐시가 하위 Servlet을 포함할 때 콘텐츠 저장을 중지하지 말도록 지시합니다. 상위 항목(CSF를 표시한 항목)이 캐시 항목에 모든 단편의 모든 콘텐츠를 포함하므로, 포함이나 전송을 갖지 않지만 항목의 전체 트리의 콘텐츠를 갖는 하나의 큰 캐시 항목이 만들어집니다. 이로서 응용프로그램 서버 처리량을 절약할 수 있지만 일반적으로 외부 HTTP 요청이 포함된 단편의 전체 트리를 판별하기 위해 필요한 모든 정보를 포함할 때만 유용합니다.

예를 들어, `<cache-entry>`이 다음과 같이 정의되는 경우,

```
<cache-entry>
  <class>Servlet</class>
  <name>com.ibm.commerce.server.RequestServlet.class</name>
  <property name="consume-subfragments">true</property>
  <property name="save-attributes">false</property>
  <property name="store-cookies">false</property>

  <!-- StoreCatalogDisplay?storeId=<storeId> -->
  <cache-id>
    <component id="" type="pathinfo">
      <required>true</required>
      <value>/StoreCatalogDisplay</value>
    </component>
    <component id="storeId" type="parameter">
      <required>true</required>
    </component>
  </cache-id>
</cache-entry>
```

+

save-attributes 특성이 false로 설정될 때 요청 속성이 캐시 항목과 함께 저장되지 않음을 주의하십시오. store-cookies 특성이 false로 설정될 때 요청 속성이 캐시 항목과 함께 저장되지 않습니다.

위의 예에서 캐시 Servlet 항목에는 StoreCatalogDisplay 명령이 전달하는 JSP 파일인 StoreCatalogDisplay.jsp를 포함합니다.

## 단편 캐시

동적으로 포함된 각 JSP 파일은 동적 캐시가 요청을 수신할 때 동적 캐시를 제공하기 위해 cachespec.xml 파일에 고유한 <cache-entry>이 정의되어야 합니다. 그렇지 않으면 동적으로 포함된 각 JSP 파일이 각 요청에 대해 재실행됩니다. 예를 들어, StoreCatalogDisplay.jsp가 동적으로 header.jsp와 footer.jsp를 포함하고 StoreCatalogDisplay.jsp에 대해 <cache-entry>만을 설정하는 경우를 고려하십시오. 그런 다음 StoreCatalogDisplay 페이지를 요청할 때, header.jsp 및 footer.jsp 파일은 캐시되지 않은 경우 실행됩니다. 다음은 StoreCatalogDisplay.jsp에 대해 <cache-entry>를 정의하는 방법의 한 예입니다.

```
<cache-entry>
  <class>servlet</class>
  <name>/ToolTech/ShoppingArea/CatalogSection/CategorySubsection/
    StoreCatalogDisplay.jsp</name>
  <property name="save-attributes">>false</property>

  <cache-id>
    <component id="storeId" type="parameter">
      <required>>true</required>
    </component>
    <component id="catalogId" type="parameter">
      <required>>false</required>
    </component>
  </cache-id>
</cache-entry>
```

---

## 캐시 Servlet 필터링

WebSphere Commerce 버전 5.5 이전에는 WebSphere Commerce가 고유한 캐시 작성 메커니즘을 제공했습니다. 이전 메커니즘을 사용하면 웹 페이지가 다음 두 유형의 방법을 기반으로 캐시될 수 있습니다.

- 세션 독립(SI) 캐시 작성 - 페이지가 URL 매개변수를 바탕으로 캐시되었습니다.
- 세션 종속(SD) 캐시 작성 - 페이지가 URL 매개변수, 사용자의 언어, 선호 통화, 상위 조직, 장기 구매 계약 ID 및 구성원 그룹을 바탕으로 캐시되었습니다.

SI 캐시에 대한 캐시 ID는 URL 매개변수를 기반으로 작성되며 SD 캐시의 경우에는 캐시 ID가 URL 매개변수에 세션 정보를 더하여 작성되었습니다.

이전 세션 종속 캐시 작성과 동일한 기능을 제공하면서 WebSphere Application Server 동적 캐시 작성 메커니즘을 사용하기 위해 WebSphere Commerce 5.5는 캐시 필터라고 하는 Servlet 필터를 도입합니다. 이 캐시 필터는 동적 캐시가 캐시 ID를 구성하기 사용하는 세션 정보로부터 요청 속성을 설정하도록 설계됩니다. 세션 정보가 WebSphere Commerce 서버 런타임에 의해 설정되므로 웹 사이트에 대한 두 번째 요청 때까지 캐시 필터가 모든 요청 속성을 설정할 수 없습니다.

다음 테이블은 캐시 필터에 의해 설정되는 요청 속성을 표시합니다.

표 10. 캐시 필터가 설정하는 요청 속성

요청 속성	설명
DC_curr	사용자의 선호 통화
DC_lang	사용자의 선호 언어
DC_porg	사용자의 상위 조직
DC_cont	사용자의 현재 장기 구매 계약
DC_mg	사용자의 명시 구성원 그룹
DC_storeId	상점 식별자
DC_userId	사용자 식별자

한 사용자가 복수 장기 구매 계약을 체결할 자격이 있고 복수 구성원 그룹에 속하므로 요청 속성 DC\_cont 및 DC\_mg가 복수 값을 포함할 수 있습니다. 이러한 사용자의 경우, 값은 정렬되고 세미콜론(";")을 분리자로 사용하여 연결됩니다. 또한 복수 장기 구매 계약 및 구성원 그룹 요청 속성이 정의됩니다(예를 들면, DC\_cont0 , DC\_cont1, ... DC\_contN이며 여기서 N은 사용자가 자격을 갖는 장기 구매 계약의 수입니다). 예를 들어, 사용자가 장기 구매 계약 10004 및 10005의 자격이 있는 경우, 다음 요청 속성이 설정됩니다. DC\_cont는 10004; 10005이고 DC\_cont0은 10004, DC\_cont1은 10005입니다.

요청 속성 DC\_cont를 설정하는 목적은 제한된 수의 구성요소를 갖는 캐시 ID를 구성할 수 있도록 하는 것이며 각각의 요청 속성 DC\_cont0, DC\_cont1, ..., DC\_contN을 설정하는 목적은 보다 미세한 캐시 무효화를 위한 종속성 ID를 구성할 수 있도록 하는 것입니다.

구성원 그룹 정보가 세션 데이터의 일부가 아니므로, 캐시 필터가 사용자 ID를 바탕으로 데이터베이스에서 이 정보를 검색해야 합니다. 데이터베이스 조회 반복으로 인한 성능 저하를 방지하기 위해 캐시 필터는 WebSphere 명령 캐시를 사용하여 이 작업을 수행합니다. WebSphere 명령 프레임워크에서 직접 확장하는 새로운 명령인

`com.ibm.commerce.dynacache.command.MemberGroupsCacheCmdImpl`이 사용자 ID를 기본으로 사용자가 속하는 구성원 그룹을 캐시하는데 사용됩니다. WebSphere Commerce는 다음 캐시 항목을 `cachespec.xml`에 추가하여 동적 캐시 작성 서비스에 이 명령을 캐시하도록 알립니다.

```

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.dynacache.commands.MemberGroupsCacheCmdImpl</name>
  <cache-id>
    <component type="method" id="getUserId">
      <required>true</required>
    </component>
  </cache-id>

  <dependency-id>DC_userId</dependency-id>

  <dependency-id>DC_userId
    <component type="method" id="getUserId">
      <required>true</required>
    </component>
  </dependency-id>
</cache-entry>

```

WebSphere Application Server 동적 캐시 작성은 get 메소드 getUserId()를 바탕으로 클래스 MemberGroupsCacheCmdImpl의 명령 오브젝트를 캐시 ID로 캐시합니다. 따라서 다음에 동일한 사용자 ID의 구성원 그룹에 대한 요청이 있으면 동적 캐시가 이미 연산된 구성원 그룹과 함께 캐시된 명령 오브젝트를 리턴합니다. 그러면 이 구성원 그룹 정보는 캐시 필터가 요청 속성 DC\_mg에 이 사용자에게 대한 구성원 그룹을 설정하기 위해 호출하는 명령 메소드 getMemberGroups()를 사용하여 쉽게 액세스할 수 있습니다.

주: WebSphere Commerce는 내부적으로 위의 캐시 필터에서 MemberGroupsCacheCmdImpl 등과 함께 WebSphere 명령 캐시를 사용하지만 WebSphere Commerce는 공식적으로는 WebSphere Commerce 명령의 WebSphere 명령 캐시를 지원하지 않습니다.

## 상점 페이지 캐시 전략

상점 페이지 설계 시 캐시 고려사항에 대한 정보는 *WebSphere Commerce* 상점 개발 안내서의 『상점 페이지 캐시』 절을 참조하십시오. 여기에는 다음의 문제에 관한 지침이 포함되어 있습니다.

- 어느 페이지를 캐시해야 하는가?
- 전체 페이지 또는 페이지 단편을 캐시해야 하는가?
- 캐시 전략 개발

또한 *WebSphere Commerce* 상점 개발 안내서의 상점 관계를 캐시에 구현하는 방법을 설명하는 절을 참조하십시오.

## WebSphere Commerce의 캐시 무효화 메커니즘

WebSphere Application Server 동적 캐시 서비스에 의해 작성된 캐시 항목에는 Servlet 출력, Servlet 실행의 결과(예: 다른 Servlet 또는 JSP 파일에 대한 호출) 및 시간 종료 및 항목 우선순위 정보 같은 항목에 대한 메타데이터를 포함한 정보가 들어있습니다. 때때로 WebSphere Commerce는 이 정보가 캐시에 남아있는 동안 현재 정보이고 일관성이 있으며 정확한지 검증해야 합니다. 그러므로 WebSphere Commerce는 더 이상 유효하지 않은 캐시 정보를 식별하고 제거하기 위한 메커니즘이 필요합니다. WebSphere Application Server 동적 캐시 서비스는 캐시 무효화라는 프로세스로 캐시 정보를 유지보수하는 지원을 제공합니다. 캐시 무효화는 WebSphere Commerce가 서비스가 제공하는 다른 무효화 메커니즘을 활용하여 다른 상황에서 무효화를 수행할 수 있게 합니다. 이들 메커니즘은 규칙 기반, 시간 기반, 그룹 기반에서 프로그램적 접근까지 다양합니다.

이 절은 WebSphere Commerce에서 캐시 무효화를 수행하는 방법의 설명에만 초점을 맞추며 일반 캐시 작성 측면을 다루려는 것이 아닙니다. WebSphere Application Server 동적 캐시가 제공하는 무효화 메커니즘에 대해 설명합니다.

- cachespec.xml에 무효화 정책 정의
- WebSphere Application Server가 제공하는 Cache Monitor를 통한 캐시 항목 제거
- WebSphere Commerce 응용프로그램 안에서 프로그램적으로 WebSphere Application Server 동적 캐시 무효화 API 호출

### cachespec.xml에 무효화 정책 정의

WebSphere Application Server 동적 캐시 작성은 cachespec.xml 파일 사용을 통해 캐시 무효화의 구성을 지원합니다. 무효화 정책은 웹 프로그램 아카이브(WAR) WEB-INF 또는 엔터프라이즈 bean WEB-INF 디렉토리에 있고 캐시할 수 있는 오브젝트 구성에 사용되는 동일한 cachespec.xml에 동적으로 정의할 수 있습니다. cachespec.xml 구문의 자세한 정보에 대해서는 WebSphere Application Server 정보 센터 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "cachespec.xml 파일" 주제를 참조하십시오.

대응하는 캐시 항목을 무효화해야 하는 이벤트를 정의하는 <invalidation> 태그 또는 캐시 항목을 보존할 지속 기간을 구성하는 <timeout> 태그로 cache-entry 안에 무효화 규칙을 지정할 수 있습니다. 무효화 규칙은 종속성 ID와 정확하게 동일한 방식으로 정의되며 무효화 규칙에 의해 작성되는 ID는 해당되는 동일한 종속성 ID를 갖는 캐시 항목을 무효화하는 데 사용됩니다. cache-entry당 복수 무효화 규칙이 존재할 수 있습니다. 모든 무효화 규칙은 개별적으로 실행합니다.

다음 절은 cachespec.xml에서 이들 무효화 규칙의 구문에 대해 설명하고 다음에 대해 WebSphere Commerce에서 규칙을 적용하는 방법의 예를 보여줍니다.

- Servlet 기반 무효화 규칙
- 명령 기반 무효화 규칙
- 시간 종료 및 우선순위 무효화 규칙

## Servlet 기반 무효화 규칙

WebSphere Application Server 동적 캐시 작성이 제공하는 이 무효화 방법은 Servlet 요청 실행 시에 다른 캐시 항목을 무효화하는 것입니다.

### cacheSpec.xml의 구문:

```
<cache-entry>
  <class>servlet</class>
  <name>servlet_name</name>
  <invalidation>invalidation_id
    <component id="" type="pathinfo">
      ...
    </component>
  </invalidation>
</cache-entry>
```

주:

#### <class>

"servlet" 값은 무효화가 Servlet에 의해 트리거됨을 표시합니다.

#### <invalidation>

무효화를 트리거한 이벤트와 무효화할 지정된 ID를 갖는 캐시 항목을 식별하는 데 사용되는 요소.

#### pathinfo

WebSphere Commerce에서 모든 Servlet 요청은 com.ibm.commerce.server.RequestServlet.class 또는 com.ibm.commerce.tools.common.ToolsRequestServlet.class 중 하나인 단일 컨트롤러 Servlet을 거칩니다.

고유한 요청 URI를 식별하기 위해, "pathinfo" 구성요소 유형을 사용하여 요청에서 pathinfo 이름을 필터링할 수 있습니다.

**cacheSpec.xml에 정의되는 정책의 예:** 목록에서 카탈로그 항목을 삭제하기 위해 InterestItemDelete URL을 발행할 때 참조 번호로 식별되는 관심 항목 목록 캐시 페이지를 무효화하려면 다음을 수행하십시오.

```
<cache-entry>
  <class>servlet</class>
  <name>com.ibm.commerce.server.RequestServlet.class</name>
  <invalidation>listId
    <component id="" type="pathinfo" ignore-value="true">
      <required>true</required>
      <value>/InterestItemDelete</value>
    </component>
    <component id="listId" type="parameter">
```



```

        <required>true</required>
      </component>
    </invalidation>
  </cache-entry>

```

## 명령 기반 무효화 규칙

이 방법은 명령 실행의 부가 효과로서 다른 캐시 항목을 무효화합니다. WebSphere Commerce의 명령 레벨에서 무효화를 트리거하는 이벤트 구동 메커니즘을 제공합니다. 사용자는 명령에 작성된 비즈니스 로직을 검사하고 명령 실행 시 영향을 받을 수 있는 특정 캐시 오브젝트를 식별한 후 그에 따라 무효화 규칙을 정의할 수 있습니다. 그런 명령의 한 기준은 명령이 WebSphere Commerce 프로그래밍 모델을 구현해야 하고 무효화 ID가 해당 입력 매개변수를 바탕으로 작성될 경우 입력 매개변수를 처리하는 명령의 메소드와 필드가 외부화되어야 한다는 것입니다.

이 지원의 정보에 대해서는 WebSphere Application Server 문서를 참조하십시오. 또한 100 페이지의 『명령 기반 무효화를 위한 명령 쓰기』를 참조하십시오.

## cachespec.xml의 구문:

```

<cache-entry>
  <class>command </class>
  <name>fully_qualified_class_name_of_the_command</name>
  <invalidation>invalidation_rules</invalidation>
</cache-entry>

```

주:

### <class>

command 값은 무효화가 명령에 의해 트리거됨을 표시합니다.

### <name>

명령의 완전한 경로가 들어있습니다.

**cachespec.xml에 정의되는 정책의 예:** 카탈로그 항목을 변경하기 위해 CatalogEntryUpdate 명령이 호출될 때 참조 번호에 의해 캐시되는 상품 표시 페이지를 무효화하려면 다음 구문을 사용하십시오.

```

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.catalogmanagement.commands.CatalogEntryUpdateCmdImpl</name>

  <invalidation>productId
    <component id="getCatentryId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>

```

## 시간 종료 및 우선순위 무효화 규칙

정책에서 <timeout> 태그를 사용한 시간 종료 메커니즘은 사전설정된 지속 기간 이후의 모든 캐시 항목을 제거하는 간단한 방법을 제공합니다. 특정 무효화 규칙이나 캐시 항목을 제거할 시간을 지정하는 트리거 이벤트를 설정하는 것이 불가능할 때 유용합니다.



WebSphere Application Server 동적 캐시 작성 서비스는 가장 오래전에 사용(LRU) 알고리즘을 바탕으로 캐시가 가득 찰 때 자동으로 캐시 항목을 제거하는데 이 알고리즘에서 <priority> 태그를 사용한 우선순위 가중치 부여가 캐시에서 제거할 항목을 결정하는 데 사용됩니다. 다른 캐시 항목과 관련하여 특정 캐시 항목에 대한 우선순위 값이 높을 수록 이 캐시 항목이 다른 항목과 관련된 캐시에 더 오래 상주하는 경향이 있습니다.

캐시가 가득 차게될 때 메모리에서 항목을 제거하는 대신 디스크 오프로드를 구성하고 나중에 사용할 수 있도록 항목을 파일 시스템에 복사할 수 있습니다(위치는 구성 가능합니다). 캐시 항목은 또한 서버 종료 시에도 디스크에 오프로드되며 서버가 다시 시작될 때 재사용할 수 있습니다. 또한, 다음 동적 캐시 서비스 설정도 무효화 프로세스에 영향을 줍니다.

#### 캐시 크기

캐시가 보유하는 최대 항목 수를 판별합니다.

#### 기본 우선순위

기본값에 의해 항목이 가득찬 캐시에서 머무는 기간을 판별합니다.

#### 디스크 오프로드

디스크 오프로드가 사용되는 여부와 디스크 오프로드 위치를 지정합니다.

### cachspec.xml의 구문:

```
<cache-entry>
  .....
  <cache-id>
    <component >.....</component>
    <priority>priority_value</priority>
    <timeout>time_in_cache</timeout>
  </cache-id>
</cache-entry>
```

**cachspec.xml에 정의되는 정책의 예:** 캐시가 작성되거나 사용된 후 3600초 후에 장바구니 캐시 콘텐츠를 제거하려면 다음 구문을 사용하십시오.

```
<cache-id>
  <component id="" type="pathinfo">
    <required>true</required>
    <value>/OrderItemDisplay</value>
  </component>
  <component id="orderId" type="parameter">
    <required>true</required>
  </component>
  <timeout>3600</timeout>
</cache-id>
```

## Cache Monitor를 통한 캐시 항목 제거

WebSphere Application Server Cache Monitor는 수동으로 캐시 항목을 무효화하는 옵션을 제공합니다. 전체 캐시 콘텐츠를 비우는 캐시 지우기 버튼이 있습니다. 각 캐시 항목은 개별적으로 무효화될 수 있으며 캐시 그룹은 템플릿(예를 들면

ProductDisplay.jsp와 연관된 모든 항목)로 무효화될 수 있습니다. 종속성 ID로 식별되는 캐시 항목의 그룹도 모니터에서 블록으로 무효화할 수 있습니다.

## 동적 캐시 무효화 API 호출

WebSphere Application Server 동적 캐시 작성은 프로그램적 무효화를 지원하는 다음 API를 제공합니다.

- `com.ibm.websphere.cache.invalidateById`
- `com.ibm.websphere.cache.invalidateByTemplate`

이들 API 사용에 대한 정보는 WebSphere Application Server 문서를 참조하십시오.

WebSphere Commerce는 이들 API를 활용하고 직접 만든 무효화 래퍼인 `DynaCacheInvalidation` 명령을 사용하여 WebSphere Commerce 안에서 테이블 구동 무효화를 지원합니다. 이 래퍼는 `CACHEIVL` 테이블의 레코드를 처리하여 캐시 오브젝트를 무효화합니다. 추가 정보는

105 페이지의 『캐시 무효화 API(DynaCacheInvalidation)』의 내용을 참조하십시오.

---

## 명령 기반 무효화를 위한 명령 쓰기

동적 캐시가 명령 호출을 인터셉트하기 위해서는 명령이 `CacheableCommandImpl`(`com.ibm.websphere.command` 패키지에 포함)에서 확장하는 구현 클래스를 사용하여 WebSphere 명령 프레임워크에 작성되어야 합니다. 명령 기반 무효화를 위한 명령 쓰기를 단순화하기 위해 WebSphere Commerce는 `CacheableCommandImpl`에서 확장하기 위해 추상 클래스인 `ControllerCommandImpl`과 `TaskCommandImpl`을 갱신했으며 따라서 이들 추상 클래스에서 확장된 모든 명령이 `CacheableCommandImpl`에서도 확장되며 따라서 명령 기반 무효화에 적합합니다.

이런 명령을 작성할 때 무효화 ID가 무엇인지 알고 명령에 대한 호출을 인터셉트하는 무효화 규칙을 이해하는 것도 중요합니다. 무효화 ID가 명령에 입력 매개변수로서 존재하는 메소드 및 필드를 바탕으로 작성되므로 무효화 ID를 구성하는 데 필요한 모든 메소드가 명령 인터페이스에 제공되고 구현되어야 합니다.

## WebSphere Commerce에서 명령 무효화 사용의 예

다음 예는 WebSphere Commerce가 명령 무효화를 사용하는 방법을 보여줍니다. 특정 구성원 그룹에 속한 특정 구성원을 삭제하는 `DeleteMemberGroupMemberCmdImpl` 명령이 실행되면 동적 캐시는 무효화 규칙에 정의된 캐시 항목을 무효화합니다. 이 예에서는 `"DC_userId:userId"`로 정의되어 있으며 여기서 `userId`는 `getMemberId` 메소드에서 리턴된 값입니다(WebSphere Commerce의 구성원 그룹 정보 캐시에 대한 자세한 내용은 93 페이지의 『캐시 Servlet 필터링』을 참조하십시오). 예를 들면, `DC_userId:-1000`, `DC_userId:-1001`, 등. 이 명령은 삭제될 사용자 ID를 검색하는 `get` 메소드인 `getMemberId()`를 갖고 있으며 이 메소드는 사용자 ID를 바탕으로 중

속성 ID를 갖는 캐시 항목 중 삭제할 항목을 계산할 때 사용됩니다. 동일한 로직이 또한 get 메소드 getMemberId()를 갖는 명령 AddMemberGroupMemberCmdImpl에도 적용됩니다.

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.membergroup.commands.AddMemberGroupMemberCmdImpl</name>
  <name>com.ibm.commerce.membergroup.commands.DeleteMemberGroupMemberCmdImpl</name>

  <invalidation>DC_userId
    <component type="method" id="getMemberId">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>
```

주: 위의 모든 무효화 규칙이 샘플 캐시 정책에서 WebSphere Commerce와 함께 제공됩니다. /samples/dynacache/invalidation 서브디렉토리의 WebSphere Commerce 설치 디렉토리에서 샘플 무효화 규칙을 찾을 수 있습니다. cachespec.xml 파일에 무효화 규칙을 통합하는 방법에 대한 추가 정보에 대해서는 "Dynacache에 대한 샘플 무효화 캐시 정책"이라는 README 파일을 참조하십시오.

---

## 캐시 무효화 예

다음 예는 WebSphere Commerce의 Consumer Direct 비즈니스 모델에 대해 ProductDisplay JSP 페이지를 캐시하기 위해 cachespec.xml 파일에 캐시 정책을 설정하는 방법과 동일한 XML 파일에 무효화 규칙을 정의하여 캐시 항목을 무효화하는 방법을 보여 줍니다. 예에서는 JSP 파일에 대한 캐시 ID 작성과 함께 복수의 종속성 ID를 정의합니다. 각 종속성 ID는 다른 상황에서 캐시 항목이 갱신될 때 캐시 항목을 무효화하는 데 사용됩니다. 이 예는 ProductDisplay JSP를 무효화하는 데 필요한 정책을 서브세트만을 보여줍니다. 전체 예와 자세한 정보는 WC\_installdir/samples/dynacache/invalidation 디렉토리에 있는 README 파일을 참조하십시오.

```
<cache>
<cache-entry>
  <class>servlet</class>
  <name>/FashionFlow/ShoppingArea/CatalogSection/
    CatalogEntrySubsection/ProductDisplay.jsp</name>
  <property name="save-attributes">>false</property>

  <!-- Cache ProductDisplay.jsp -->
  <cache-id>
<component id="storeId" type="parameter">
  <required>true</required>
</component>
```

```

        <component id="catalogId" type="parameter">
            <required>true</required>
        </component>
        <component id="productId" type="parameter">
            <required>true</required>
        </component>
        <component id="DC_lang" type="attribute">
            <required>true</required>
        </component>
        <component id="DC_curr" type="attribute">
            <required>true</required>
        </component>
        <component id="DC_porg" type="attribute">
            <required>true</required>
        </component>
        <component id="DC_cont" type="attribute">
            <required>true</required>
        </component>
        <component id="DC_mg" type="attribute">
            <required>true</required>
        </component>
    </cache-id>

    <!-- 특정 상점에 속한 상품 표시 캐시 항목 -->
    <!-- 무효화에 사용 -->
    <dependency-id>storeId
        <component id="storeId" type="parameter">
            <required>true</required>
        </component>
    </dependency-id>

    <!-- 특정 상품의 캐시 항목 무효화에 사용 -->
    <dependency-id>productId
        <component id="productId" type="parameter">
            <required>true</required>
        </component>
    </dependency-id>

    <!-- 상점의 특정 카탈로그에 속한 상품 표시 캐시 항목 -->
    <!-- 무효화에 사용 -->
    <dependency-id>storeId:catalogId
        <component id="storeId" type="parameter">
            <required>true</required>
        </component>
        <component id="catalogId" type="parameter">
            <required>true</required>
        </component>
    </dependency-id>

    <!-- 특정 장기 구매 계약을 한 상품 표시 -->
    <!-- 캐시 항목 무효화에 사용 -->
    <dependency-id>contractId
        <component id="DC_cont0" type="attribute">
            <required>true</required>
        </component>
    </dependency-id>

```

```

</cache-entry>

<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>
  <name>com.ibm.commerce.catalogmanagement.commands.AddCatalogDescCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.UpdateCatalogDescCmdImpl</name>

  <!-- ***** -->
  <!-- 카탈로그 설명이 변경될 때 영향을 받을 수 있는 -->
  <!-- 모든 상품 페이지 캐시 항목을 무효화 -->
  <!-- ***** -->
  <invalidation>storeId:catalogId
    <component id="getStoreId" type="method">
      <required>true</required>
    </component>
  <component id="getCatalogId" type="method">
    <required>true</required>
  </component>
</invalidation>

</cache-entry>

<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>
  <name>com.ibm.commerce.catalogmanagement.commands.ListpriceAddCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.ListpriceDeleteCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.ListpriceUpdateCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.OfferAddCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.OfferDeleteCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.OfferUpdateCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.ProductAttributeUpdateCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.AttributeValueUpdateCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.AddListpriceCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.DeleteListpriceCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.UpdateListpriceCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.AddOfferCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.DeleteOfferCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.UpdateOfferCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.UpdateAttributeCmdImpl</name>
  <name>com.ibm.commerce.catalogmanagement.commands.UpdateAttributeValueCmdImpl</name>

  <!-- ***** -->
  <!-- 상품 갱신 시 특정 상품 페이지 캐시 항목을 -->
  <!-- 무효화 -->
  <!-- ***** -->
  <invalidation>productId
    <component id="getCatentryId" type="method">
      <required>true</required>
    </component>
  </invalidation>

</cache-entry>

<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>

```

```

<name>com.ibm.commerce.contract.commands.ContractSuspendCmdImpl</name>
<name>com.ibm.commerce.contract.commands.ContractTCDeployCmdImpl</name>

<!-- ***** -->
<!-- 특정 장기 구매 계약을 한 모든 상품 페이지 캐시 -->
<!-- 항목을 무효화 -->
<!-- ***** -->
  <invalidation>contractId
    <component id="getContractId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreProfileUpdateCmdImpl</name>
  <name>com.ibm.commerce.tools.devtools.flexflow.ui.commands.impl.FlexflowUpdateCmdImpl</name>
  <name>com.ibm.commerce.store.commands.StoreOpenCmdImpl</name>
  <name>com.ibm.commerce.store.commands.StoreCloseCmdImpl</name>

  <!-- ***** -->
  <!-- 상점이 갱신될 때 상점에 속한 모든 상품 페이지 -->
  <!-- 캐시 항목을 무효화 -->
  <!-- ***** -->
  <invalidation>storeId
    <component id="getStoreId" type="method">
      <required>true</required>
    </component>
  </invalidation>

</cache-entry>

<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>
  <name>com.ibm.commerce.catalogimport.commands.CatalogImportJobAddCmd</name>

  <!-- ***** -->
  <!-- 상점 카탈로그가 갱신될 때 상점에 속한 -->
  <!-- 모든 상품 페이지 캐시 항목을 무효화 -->
  <!-- ***** -->
  <invalidation>storeId
    <component id="getStoreId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>

</cache>

```

---

## 캐시 무효화 API(DynaCacheInvalidation)

WebSphere Commerce 스케줄러는 주기적으로 DynaCacheInvalidation 명령을 호출하여 CACHEIVL 테이블의 레코드를 처리한 후 동적 캐시 무효화 API를 호출하여 캐시 항목을 무효화합니다. startTime이 체크포인트 파일에서 검색됩니다. 체크포인트 파일은 instance\_name/cache 디렉토리에 기록됩니다. 이 디렉토리는 처음 DynaCacheInvalidation이 실행될 때 작성됩니다. 마지막 레코드의 시간소인이 명령의 와인드업 단계 중에 체크포인트 파일에 기록됩니다.

---

## CACHEIVL 테이블

CACHEIVL 테이블을 처리할 때 DynaCacheInvalidation 명령에 대한 규칙은 다음과 같습니다.

- 템플릿 열이 설정되면 DynaCacheInvalidation 명령은 동적 캐시 무효화 API(invalidateByTemplate)를 호출하고 그 이름을 템플릿 ID로 사용합니다(예 : /webapp/wcs/stores/ToolTech/ShoppingArea/CatalogSection/CategorySubsection/StoreCatalogDisplay.jsp).
- DATA\_ID 열이 설정되고 템플릿 이름이 설정되지 않으면 DynaCacheInvalidation 명령이 동적 캐시 무효화 API(invalidateById)를 호출하고 DATA\_ID를 해당 ID로 사용합니다(예: StoreCatalogDisplay:storeId:10151).
- 동적 캐시 무효화 API는 호출될 때 캐시 항목을 무효화합니다.

---

## 외부 캐시에서 무효화

결과 캐시를 IBM HTTP Server 또는 Edge Server 같은 외부 캐시에 푸쉬할 수 있습니다. 다음 경우에 외부 캐시에서 무효화가 발생합니다.

- 캐시가 정체되는 경우
- 시간 종료가 발생하는 경우
- 무효화 메시지가 WebSphere Application Server에서 보내지는 경우

---

## 웹 서버 플러그인 캐시

WebSphere Application Server 5.0의 경우, 웹 서버 플러그인에 빌트인 ESI(Edge Side Include) 프로세서가 들어있습니다. ESI 프로세서는 단편뿐 아니라 전체 페이지를 캐시할 수 있어서 더 높은 캐시 적중율을 제공합니다. ESI 프로세서에 의해 구현되는 캐시는 디스크 캐시가 아니라 캐시 메모리입니다. 그러므로 웹 서버를 다시 시작할 때 캐시 항목이 저장되지 않습니다.

웹 서버 플러그인이 요청을 수신할 때 요청은 ESI 프로세서가 사용되는 한 ESI 프로세서로 보내집니다(기본적으로 사용됩니다). 캐시 누락이 발생하는 경우,



Surrogate-Capabilities 헤더가 요청에 추가되고 요청이 WebSphere Application Server 로 전달됩니다. 동적 Servlet 캐시를 Application Server에서 사용하고 응답이 전달 가능한 경우, Application Server는 WebSphere 플러그인에 대한 응답으로 Surrogate-Control 헤더를 리턴합니다.

ESI 프로세서는 WebSphere 웹 서버 플러그인 구성 파일인 plugin-cfg.xml을 통해 구성할 수 있습니다.

예:

```
<Property Name="esiEnable" Value="true"/>
<Property Name="esiMaxCacheSize" Value="1024"/>
<Property Name="esiInvalidationMonitor" Value="true" />
```

여기서,

- esiEnable은 값을 false로 설정하여 ESI 프로세서를 사용 불가능합니다. ESI는 기본적으로 사용됩니다.
- esiMaxCacheSize는 1K 바이트 단위의 최대 캐시 크기입니다. 캐시의 기본 최대 크기는 1MB입니다. 캐시가 가득 찬 경우, 캐시에서 제거될 첫 번째 항목은 만기에 가장 가까운 항목입니다. (ESI 캐시 항목에 대한 기본 만기 시간은 24시간입니다.)
- esiInvalidationMonitor는 ESI 프로세서가 응용프로그램 서버에서 무효화를 수 신할지 여부를 지정합니다.

ESI 캐시에서 캐시 항목을 지울 수 있는 세 가지 방법이 있습니다.

- 항목에 대한 만기 시간 종료를 호출합니다.
- 새 항목을 위한 공간을 작성하기 위해 기존 항목을 삭제합니다.
- Application Server가 항목 그룹에 대해 명시된 무효화를 보냅니다.

이러한 메커니즘을 사용하려면 esiInvalidationMonitor가 TRUE로 설정되어야 하고 installableApps 디렉토리에 있는 DynaCacheEsi.ear 응용프로그램이 Application Server에 설치되어야 합니다.

주: DynaCacheEsi.ear은 VH\_instance\_name을 가상 호스트로 사용해야 합니다.

ESI 프로세서에 대한 추가 정보에 대해서는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 『ESI(Edge Side Include) 캐시 작성 구성』을 참조하십시오.

---

## 단순 파일 Servlet

기본적으로 WebSphere Commerce는 이미지와 HTML 같은 정적 데이터를 캐시하지 않습니다. 이것은 시스템 특성 com.ibm.servlet.file.esi.timeOut을 값 0(영)으로 설정합니다. 이미지 및 HTML 캐시 작성을 사용 가능하게 하려면 다음 중 하나를 수행하십시오.



- cachespec.xml 파일에 SimpleFileServlet에 대한 캐시 항목을 설정하십시오. WebSphere Application Server에서 캐시될 것을 제어하면 필요할 때 캐시 항목을 무효화할 수 있기 때문에 cachespec.xml에 캐시 항목을 설정하는 것이 더 좋은 접근 방식입니다.

응용프로그램 서버에서 모든 이미지를 캐시 항목으로 저장하고 있을 수 있으며, 이것이 캐시 항목의 최대 수에 도달할 때 성능을 저하시킬 수 있음을 주의하십시오. SimpleFileServlet 캐시 항목에 대해서는 다른 캐시 항목에 비해 낮은 우선순위 값을 설정하도록 권장합니다.

- 시스템 특성 com.ibm.servlet.file.esi.timeOut 설정을 영이 아닌 시간 종료 값으로 수정하십시오.

일부 이미지 파일에 대해 무효화가 필요한 이유는 일부 비즈니스 모델에서 로고와 배너를 쉽게 수정할 수 있기 때문입니다.

예:

```
<cache-entry>
  <class>servlet</class>
  <name>com.ibm.ws.webcontainer.servlet.SimpleFileServlet.class</name>
  <property name="EdgeCacheable">true</property>
  <cache-id>
    <component id="" type="pathinfo">
      <required>true</required>
    </component>
  </cache-id>
  <dependency-id>
    <component id="" type="pathinfo">
      <required>true</required>
    </component>
  </dependency-id>
  <timeout>1800</timeout>
</cache-entry>
```

다음은 이미지 무효화를 위한 캐시 항목의 예입니다.

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreLogoUpdateCmdImpl</name>
  <invalidation>FashionFlow/images/logo.gif</invalidation>
</cache-entry>

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreBannerUpdateCmdImpl</name>
  <invalidation>FashionFlow/images/banner.gif</invalidation>
</cache-entry>
```

## 문제점 판별

동적 캐시 작성을 사용하여 페이지가 적절히 캐시되게 하는 데 문제점이 발생하는 경우, 다음 정보를 참조하십시오.

- 문제점: Cache Monitor에 액세스할 수 없습니다.

해결책: 웹 서버 플러그인을 재작성했고 웹 서버와 Application Server를 다시 시작해서 새 가상 호스트와 포트 번호가 로드되는지 확인하십시오.

- 문제점: 어떤 것이든 캐시되는 페이지가 없습니다.

해결책: cachespec.xml 파일이 WebSphere Application Server에 로드되었는지 검증하십시오. cachespec.xml이 로드되면 측면 표시줄에 있는 캐시 정책을 눌러 캐시 모니터에서 정책을 볼 수 있어야 합니다. 또한 WebSphere Application Server SystemOut.log 파일이 다음과 유사한 행을 표시합니다.

```
[6/6/03 21:21:12:635 EDT] 65c60609 ConfigManager I DYNA0047I: 성공적으로 로드되었습니다.  
cache configuration file D:\WebSphere\AppServer5\installedApps\buzz\WC_demo.ear\  
Stores.war\WEB-INF/cachespec.xml.
```

위의 행은 보기 편하게 하기 위해 분리되었습니다.

- 문제점: 페이지 중 일부만이 캐시되고 있습니다.

해결책: cachespec.xml 파일에 문제점이 있습니다. 추적을 켜고 조사할 페이지에 액세스할 수 있습니다. 추적이 문제점을 표시할 것입니다.

추적 사용에 대한 정보는 70 페이지의 『추적 구성요소 사용』을 참조하십시오.

동적 캐시 추적(WebSphere Commerce의 경우)을 켜기 위해서는 추적 스펙에 "com.ibm.websphere.commerce.WC\_CACHE=all=enabled"가 있어야 합니다.

동적 캐시 추적(WebSphere Application Server의 경우)을 켜기 위해서는 추적 스펙에 "com.ibm.ws.cache.\*=all=enabled"가 있어야 합니다.

- 문제점: DynaCacheInvalidation 명령으로 무효화되지 않습니다.

해결책: URL에서 명령이 가능하고 있는지 검증하십시오. URL에서 명령을 실행해야 하는 경우, cacheinvalidation.jsp 파일이 docpath에 있어야 합니다. 경로에 없는 경우에도, 명령은 실행하지만 완료를 보고하는 표시 페이지가 수신되지 않습니다. URL 호출이 올바른 경우, WebSphere Commerce 스케줄러를 통해 실행하도록 계획되었는지를 검증할 수 있습니다. 기본값은 무계획적으로 실행하는 것입니다.

- 문제점: 종속성 ID를 통한 무효화가 제대로 작동하지 않습니다.

해결책: 의도한 Servlet 또는 JSP 파일에 대한 종속성 ID가 동일한 cache-entry 블록에 있는지 검증하십시오. 이 경우라면 다른 캐시 항목 블록의 동일한 종속성 ID에 대한 복수 정의가 동일한 정의를 갖는지 확인해야 합니다. 즉, 다음과 같이 종속성 ID가 일부 cache-entry에 대해 정의되는 경우, 임의의 다른 cache-entry 블록에 있는 종속성 ID의 정의가 정확하게 동일해야 합니다.

```
<dependency-id>storeId  
  <component id="storeId" type="parameter">  
    <required>true</required>  
  </component>  
</dependency-id>
```



---

## 제 6 장 시스템 유지보수





이 장에서는 데이터베이스 정리 유틸리티 및 유틸리티를 사용하여 시스템의 데이터베이스를 유지보수하는 방법에 대해 설명합니다.





---

### 데이터베이스 정리 유틸리티

데이터베이스 정리 유틸리티를 사용하면 다수의 오브젝트를 동시에 데이터베이스에서 삭제할 수 있습니다. 데이터베이스의 많은 정보를 변경했고 사용하지 않는 테이블이나 행이 있는 경우에 이 작업을 수행합니다. 데이터베이스에서 삭제할 수 있는 여러 가지 오브젝트가 있습니다. 이 오브젝트에 관한 추가 정보는 122 페이지의 『데이터베이스 정리』를 참조하십시오.

데이터베이스 정리 유틸리티가 오브젝트를 삭제할 때 오브젝트 테이블의 레코드가 데이터베이스의 참조 무결성을 보존하기 위해 삭제됩니다. 데이터베이스 정리 유틸리티 명령은 하향식 또는 상향식 중 한 방법으로 데이터베이스를 정리합니다. 하향식은 계단식 삭제 방법을 사용하여 하위 테이블의 모든 행을 삭제합니다. 참조 무결성에 삭제 제한이 지정되어 있으면 계단식 삭제는 실패하므로 상향식 방법을 사용해야 합니다.

    상향식 방법을 사용하려면 명령 구문에서 FORCE 매개변수를 yes로 지정하십시오. 그러면 하위 테이블을 먼저 삭제한 다음 상위 테이블을 삭제합니다.

    상향식 방법을 트리거하는 또다른 방법은 명령 구문에서 LOGLEVEL 매개변수를 2로 지정하는 것입니다. 0을 지정하면 아무 것도 로그하지 않고 1을 지정하면 맨 위 테이블의 삭제 명령문만 로그합니다. LOGLEVEL 2는 각각 삭제된 하위 테이블에서 맨 위 테이블까지 명령문을 삭제합니다. 2를 선택하면 상향식 방법을 트리거하지만 참조 무결성에 삭제 제한이 있는 경우 삭제 성공을 보장할 수 없습니다. 삭제 제한이 있는 레코드를 삭제하려면 FORCE 매개변수를 yes로 지정하십시오.

테이블에 많은 하위 테이블이 있는 경우 상향식 방법으로 더 긴 응답 시간을 예상할 수 있습니다. 예를 들어, MEMBER 테이블은 500개 이상의 하위 테이블을 포함하고 있습니다. 성능상의 이유로 하향식 방법 사용을 권장합니다.

주: 로그 레벨 0 또는 로그 레벨 1이 실패하면 로그 레벨 2만을 사용하십시오. 아니오로 설정되는 FORCE 매개변수가 실패하면, 예로 설정되는 FORCE 매개변수만 사용하십시오.

+  
+  
+

데이터베이스 정리 유틸리티는 구성 가능하고 확장 가능하며 개조 가능합니다. 사전 설정된 정리 구성을 제외하고, 새 오브젝트를 데이터베이스 테이블에 추가하여 정리할 테이블 및 행을 정의할 수 있습니다. 새 구성을 데이터베이스 정리 유틸리티에 추가하는 법을 참조하십시오.

새 테이블을 작성하여 데이터베이스 스키마를 확장한 경우, 데이터베이스 정리 유틸리티를 사용하여 새 테이블을 정리할 수 있습니다. 데이터베이스 스키마를 변경한 경우(예 : 새 열을 테이블에 추가, foreign key 1차 키 관계 변경 또는 새 하위 테이블을 참조 무결성 경로에 추가), 데이터베이스 정리 유틸리티가 자동으로 변경 사항에 적용합니다. 열 이름을 변경하는 경우, CLEANCONF 테이블의 구성 데이터를 갱신하십시오.

데이터베이스 정리 유틸리티는 데이터베이스 스키마의 참조 무결성 정의의 삭제 규칙에 근거하여 하위 테이블의 레코드를 삭제합니다. 삭제 규칙을 계단식 삭제, 삭제 시 널 (null)값으로 설정 또는 삭제 제한으로 설정할 수 있습니다. 새 테이블을 추가하는 경우, 참조 무결성과 삭제 규칙을 올바르게 정의했는지 확인하십시오. 그렇지 않으면, 데이터베이스 정리 유틸리티가 새 테이블을 처리할 수 없습니다.

주: staglog 오브젝트를 정리하려면 스테이징 서버에서만 데이터베이스 정리 유틸리티를 실행해야 합니다. 스테이징 데이터베이스는 프로덕션 데이터베이스와 다릅니다. 스테이징 데이터베이스에는 조작 데이터는 없고 구성 데이터만 있습니다. 구성 데이터를 삭제하면 조작 데이터가 계단식 삭제될 수도 있습니다. 스테이징 전파 유틸리티가 프로덕션 데이터베이스로 삭제를 전파할 때, 조작 데이터(보관하려는 데이터)에 대해 계단식 삭제가 유발될 수 있습니다. 구성 데이터를 정리하려면, 프로덕션 데이터베이스에서 데이터베이스 정리 유틸리티를 실행하십시오.

## 데이터베이스 정리 유틸리티 오브젝트

데이터베이스 정리 유틸리티는 CLEANCONF 테이블을 참조하여 특정 오브젝트 및 오브젝트 유형이 지정될 때 삭제할 테이블 및 행을 판별합니다. 다음 테이블은 CLEANCONF 테이블의 사전 구성된 삭제 시나리오에 대해 설명합니다. CLEANCONF 테이블에 유사한 행을 추가하여 고유한 삭제 오브젝트를 구성할 수 있습니다.

오브젝트	유형	명령문
account	obsolete	delete from account where markfordelete = 1 and trdtype_id = 0 and trading_id not in (select account_id from trading) and trading_id not in (select distinct account_id from ordpymthd)
address	obsolete	delete from address where status = 'T' and (days(CURRENT_TIMESTAMP) - days(lastcreate)) >= ? and (address_id not in (select distinct address_id from orderitems where address_id is not null)) and (address_id not in (select distinct address_id from orders where address_id is not null)) and (address_id not in (select distinct allocaddress_id from orderitems where allocaddress_id is not null))

오브젝트	유형	명령문
atp_inventory	obsolete	delete from receipt where qtyonhand = 0 and qtyinkits = 0 and receipt_id not in (select distinct receipt_id from ordpickhst where receipt_id is not null) and receipt_id not in (select distinct receipt_id from ordshipst where receipt_id is not null)
attachment	obsolete	delete from attachment where days(current timestamp) - days(timeupdated) >=? and attachment_id not in (select attachment_id from trdattach)
auction	retracted	delete from auction where austatus = 'R' and (days(CURRENT TIMESTAMP) - days(updatetime)) >= ?
auction	settlement_closed	delete from auction where austatus = 'SC' and (days(CURRENT TIMESTAMP) - days(updatetime)) >= ?
auctionlog	obsolete	delete from auctionlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?
autobidlog	obsolete	delete from autobidlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?
baseitem	obsolete	delete from baseitem where markfordelete = 1 and baseitem_id not in (select baseitem_id from catentry) and baseitem_id not in (select distinct baseitem_id from itemspc where markfordelete = 0 and itemspc_id in (select distinct itemspc_id from orderitems) or itemspc_id in (select distinct itemspc_id from oicomplst) or itemspc_id in (select distinct itemspc_id from versionspc where versionspc_id in (select distinct versionspc_id from receipt)) or itemspc_id in (select distinct itemspc_id from radetail) or itemspc_id in (select distinct itemspc_id from bkordalloc) or itemspc_id in (select distinct itemspc_id from invreserve) or itemspc_id in (select distinct itemspc_id from rmaitem) or itemspc_id in (select distinct itemspc_id from rmaitemcmp) or itemspc_id in (select distinct itemspc_id from catentry))
bidlog	obsolete	delete from bidlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?
calculation_code	obsolete	delete from calcode where published = 2 and calcode_id not in (select distinct calcode_id from ordadjust where calcode_id is not null) and calcode_id not in (select distinct calcode_id from stencalug where calcode_id is not null) and calcode_id not in (select distinct calcode_id from ordcalcd where calcode_id is not null) and calcode_id not in (select distinct calcode_id from ordicalcd where calcode_id is not null)
catalog_group	obsolete	delete from catgroup where markfordelete = 1

오브젝트	유형	명령문
catentry	without_orderitems	delete from catentry where markfordelete = 1 and(days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and catentry_id not in (select distinct catentry_id from auction) and catentry_id not in (select distinct catentry_id from orderitems where catentry_id is not null) and catentry_id not in (select distinct catentry_id from oicomplst where catentry_id is not null) and catentry_id not in (select distinct catentry_id from rmaitem where catentry_id is not null) and catentry_id not in (select distinct catentry_id from offer where offer_id in (select distinct offer_id from orderitems ) )
catentry	without_orderitems-iitems	delete from catentry where markfordelete = 1 and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and catentry_id not in (select distinct catentry_id from auction) and catentry_id not in (select distinct catentry_id from orderitems) and catentry_id not in (select distinct catentry_id from oicomplst where catentry_id is not null)and catentry_id not in (select distinct catentry_id from iitem) and catentry_id not in (select distinct catentry_id from rmaitem where catentry_id is not null) and catentry_id not in (select distinct catentry_id from offer where offer_id in (select distinct offer_id from orderitems))
contract	obsolete	1. delete from trading where markfordelete = 1 and trdtype_id = 1 and trading_id not in (select distinct trading_id from orderitems) and trading_id not in (select distinct trading_id from rma) and trading_id not in (select distinct trading_id from ordpaymthd) and trading_id not in (select distinct account_id from ordpaymthd)  2. delete from productset where markfordelete = 1 and productset_id not in (select distinct productset_id from tradeposcn where tradeposcn_id in (select distinct tradeposcn_id from offer where offer_id in (select distinct offer_id from orderitems ) ))  3. delete from tradeposcn where markfordelete = 1 and tradeposcn_id not in (select distinct tradeposcn_id from offer where offer_id in (select distinct offer_id from orderitems))
coupon_promotion	expired	delete from cppmn where days(current timestamp) - days(enddate) >=?
cpnlog	obsolete	delete from cpnlog
cpnstats	obsolete	delete from cpnstats
expected_inventory_ records	obsolete	delete from ra where markfordelete = 1 and ra_id not in (select distinct ra_id from receipt, radetail where receipt.radetail_id = radetail.radetail_id)
expected_inventory_ record_details	obsolete	delete from radetail where markfordelete = 1 and radetail_id not in (select distinct radetail_id from receipt)



오브젝트	유형	명령문
forummsg	obsolete	delete from forummsg where msgstatus = 'D' or (days(CURRENT TIMESTAMP) - days(posttime)) >= ?
fulfillment_center	obsolete	delete from ffmcenter where markfordelate = 1 and ffmcenter_id not in (select distinct ffmcenter_id from radetail) and ffmcenter_id not in (select distinct ffmcenter_id from inventory) and ffmcenter_id not in (select distinct ffmcenter_id from rma ) and ffmcenter_id not in (select distinct ffmcenter_id from orderitems) and ffmcenter_id not in (select distinct allocffmc_id from orderitems) and ffmcenter_id not in (select distinct ffmcenter_id from store) and ffmcenter_id not in (select distinct rtnffmctr_id from store) and ffmcenter_id not in (select distinct ffmcenter_id from receipt) and ffmcenter_id not in (select distinct ffmcenter_id from auction) and ffmcenter_id not in (select distinct ffmcenter_id from auctionlog)
inventory_adjustments	obsolete	delete from invadjust where days(CURRENT TIMESTAMP) - days(adjustmentdate) >= ?
inventory_adjustment_codes	obsolete	delete from invadjcode where markfordelate = 1 and invadjcode_id not in (select distinct invadjcode_id from invadjust)
itemspecification	obsolete	delete from itemspc where markfordelate = 1 and itemspc_id not in (select distinct itemspc_id from orderitems) and itemspc_id not in (select distinct itemspc_id from oicomplis) and itemspc_id not in (select distinct itemspc_id from versionspc where versionspc_id in (select distinct versionspc_id from receipt)) and itemspc_id not in (select distinct itemspc_id from radetail) and itemspc_id not in (select distinct itemspc_id from bkordalloc) and itemspc_id not in (select distinct itemspc_id from invreserve) and itemspc_id not in (select distinct itemspc_id from rmaitem) and itemspc_id not in (select distinct itemspc_id from rmaitemcmp) and itemspc_id not in (select distinct itemspc_id from catentry)
message	obsolete	delete from message where message_id not in (select message_id from msgmemrel) or (days(CURRENT TIMESTAMP) - days(posttime)) >= ?
msgmemrel	obsolete	delete from msgmemrel where message_id in (select m.message_id from message ms, msgmemrel m where ms.message_id = m.message_id and (status = 'D' or ((status = 'O' or sendstat = 'S') and (days(CURRENT TIMESTAMP) - days(posttime)) >= ?))
order	canceled	delete from orders where status = 'X' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	completed	delete from orders where status = 'C' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	deposited	delete from orders where status = 'D' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?

오브젝트	유형	명령문
order	shipped	delete from orders where status = 'S' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	stale_guest	delete from orders where (status = 'P' or status = 'I' or status = 'W' or status = 'N') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and orders.member_id in (select distinct users_id from users where registertype = 'G') and (orders_id not in (select distinct orders_id from orderitems where inventorystatus != 'NALC' and inventorystatus is not null))
order	stale_non_guest	delete from orders where (status = 'P' or status = 'I' or status = 'W' or status = 'N') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and orders.member_id in (select distinct users_id from users where registertype != 'G') and (orders_id not in (select distinct orders_id from orderitems where inventorystatus != 'NALC' and inventorystatus is not null))
organization	obsolete	delete from member where member_id in (select orgentity_id from orgentity where orgentity_id = ?)
pastats	obsolete	delete from pastats
pcstats	obsolete	delete from pcstats
pestats	obsolete	delete from pestats
policy	obsolete	delete from policy where days(current timestamp) - days(endtime) > ? and policy_id not in (select distinct policy_id from ordpaymthd) and policy_id not in (select distinct policy_id from rma)
rfq	obsolete	delete from trading where markfordelete = 1 and trdtype_id in (2, 3, 4) and trading_id not in (select distinct trading_id from orderitems) and trading_id not in (select distinct trading_id from rma) and trading_id not in (select distinct trading_id from ordpaymthd) and trading_id not in (select distinct account_id from ordpaymthd)
rma	abandoned	delete from rma where status in ('PRC', 'EDT') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rmaitem where rmaitem.status in ( 'APP', 'MAN')) and rma_id not in (select rma_id from rtnreceipt)
rma	approved_or_partly_ approved	delete from rma where status = 'PND' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id in (select rma_id from rmaitem where rmaitem.status in ('APP', 'MAN') ) and rma_id not in (select rma_id from rtnreceipt)
rma	canceled	delete from rma where status = 'CAN' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
rma	completed	delete from rma where status = 'CLO' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rtnreceipt)

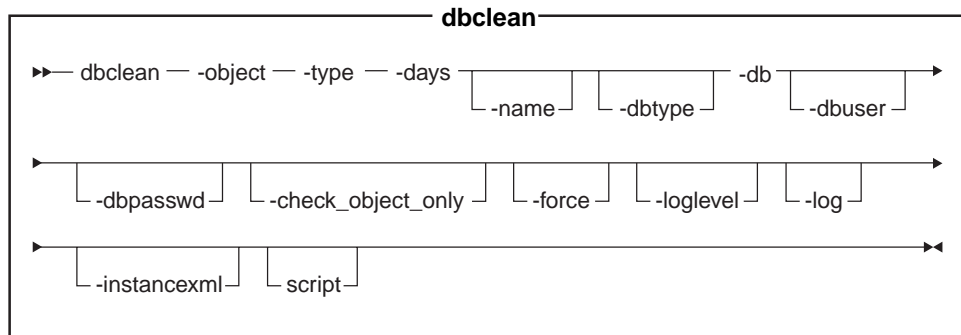
오브젝트	유형	명령문
rma	not_approved	delete from rma where status = 'PND' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rmaitem where rmaitem.status in ('APP', 'MAN') ) and rma_id not in (select rma_id from rtnreceipt)
rtnreasons	obsolete	delete from rtnreason where markfordelete = 1 and rtnreason_id not in (select distinct rtnreason_id from rtnrcptdsp) and rtnreason_id not in (select distinct rtnreason_id from rmaitem)
sastats	obsolete	delete from sastats
staglog	obsolete	delete from staglog where stgprocessed = 1 and (days(CURRENT TIMESTAMP) - days(stgstmp)) >= ?
store	obsolete	delete from storeent where storeent_id = ? and type='S'
users	guest	delete from member where member_id in (select users_id from users where registertype='G' and (days(CURRENT TIMESTAMP) - days(lastsession)) >= ? And (users_id not in (select member_id from orders where status != 'Q')) and (users_id > 0) and (users_id not in (select member_id from address where address_id in (select address_id from orderitems where address_id is not null and status != 'Q') or address_id in (select allocaddress_id from orderitems where allocaddress_id is not null and status != 'Q') or address_id in (select address_id from orders where address_id is not null and status !='Q'))))
users	registered	delete from member where member_id in (select users_id from users where registertype= 'R' and (days(CURRENT TIMESTAMP) - days(lastsession)) >= ? and (users_id not in (select member_id from orders where status != 'Q')) and (users_id > 0) and (users_id not in (select member_id from address where address_id in (select address_id from orderitems where address_id is not null and status != 'Q') or address_id in (select allocaddress_id from orderitems where allocaddress_id is not null and status != 'Q') or address_id in (select address_id from orders where address_id is not null and status !='Q'))))
usrtraffic	obsolete	delete from usrtraffic where (days(CURRENT TIMESTAMP) - days(stmp)) >= ?
vendor	obsolete	delete from vendor where markfordelete = 1 and vendor_id not in (select distinct vendor_id from ra) and vendor_id not in (select distinct vendor_id from receipt where vendor_id is not null)
productset	obsolete	delete from productset where markfordelete = 1 and productset_id not in (select productset_id from tradeposcn where productset_id is not null)
productset	obsolete	delete from productset where productset_id in (select productset_id from tradeposcn where productset_id is not NULL and markfordelete = 1 and type = 'C')

오브젝트	유형	명령문
tradeposcn	obsolete	delete from tradeposcn where markfordelete = 1 and type = 'S'

## 데이터베이스 정리 명령

### 데이터베이스 정리 유틸리티 명령(AIX, iSeries, Linux, Solaris 및 Windows 2000)

데이터베이스 정리 유틸리티는 데이터베이스에서 오브젝트를 제거합니다. 데이터베이스 정리 유틸리티를 실행하려면 명령행에서 다음을 입력하십시오. 전체 명령을 한 행에 입력하십시오.



▶ 400 iSeries는 이제 dbclean.sh를 사용합니다. iSeries에서 셸 스크립트를 실행하려면 다음을 수행하십시오.

1. 65535 외의 다른 CCSID를 가진 사용자 프로파일로 로그인하십시오.
2. OS/400 명령행에서 STRQSH 명령을 입력하여 QSHELL 명령창을 여십시오.
3. 다음과 같이 명령을 실행하십시오. /QIBM/ProdData/CommerceServer55/bin/dbclean.sh (매개변수 . . .)

주: 매개변수 -dbuser를 항상 지정해야 합니다.

▶ Oracle 주:

- 동일한 사용자 ID를 사용하여 현재 이 유틸리티를 실행 중인 경우에도 명령에 선택적 매개변수, 로그인 사용자 ID 및 암호를 포함시켜야 합니다.
- ▶ Linux ▶ 400 Oracle 참조는 iSeries 또는 Linux 플랫폼에 적용되지 않습니다.

매개변수 값:

**object** 쓸모없게 된 레코드를 삭제하는 오브젝트의 이름입니다. 다음 오브젝트 이름 중 하나를 입력하십시오.

- account: 계정 오브젝트를 삭제합니다.

- address: 주소 오브젝트를 삭제합니다.
- atp\_inventory: 수령 정보 오브젝트를 삭제합니다.
- attachment: 첨부 오브젝트를 삭제합니다.
- auction: 경매 오브젝트를 삭제합니다.
- auctionlog: 경매 로그 오브젝트를 삭제합니다.
- autobidlog: 자동 입찰 오브젝트를 삭제합니다.
- baseitem: 상품 정보 오브젝트를 삭제합니다.
- bidlog: 입찰 로그 오브젝트를 삭제합니다.
- calculation\_code: 계산 코드 오브젝트를 삭제합니다.
- catentry: 카탈로그 항목 오브젝트를 삭제합니다.
- catalog\_group: 카탈로그 그룹 오브젝트를 삭제합니다.
- contract: 장기 구매 계약 오브젝트를 삭제합니다.
- coupon\_promotion: 쿠폰 오브젝트를 삭제합니다.
- cpnlog: 캠페인 오브젝트를 삭제합니다.
- cpnstats: 캠페인 통계 오브젝트를 삭제합니다.
- expected\_inventory\_records: 재고 오브젝트를 삭제합니다.
- expected\_inventory\_records\_details: 재고 정보 오브젝트를 삭제합니다.
- forummsg: 사이트 운영자와 고객 사이의 메시지를 삭제합니다.
- fulfillment\_center: 서비스 센터 오브젝트를 삭제합니다.
- inventory\_adjustments: 재고 오브젝트를 삭제합니다.
- inventory\_adjustment\_codes: 재고 코드 오브젝트를 삭제합니다.
- itemspecification: 지정된 항목 오브젝트를 삭제합니다.
- jdbccustomizer: dbclean과 함께 사용할 customizer 매개변수의 위치를 지정합니다. 도구상자 jdbc 드라이버를 iSeries와 함께 사용하도록 dbclean을 구성하려면 DB2/iSeries를 dbtype으로 지정하고 customizer 정의 파일에는  
jdbcTemplate=com.ibm.as400.access.AS400JDBCdriver  
jdbcTemplatePrefix=jdbc:as400://이 들어 있어야 합니다. iSeries에서 DB2에 대한 기본값은 jdbcTemplate=com.ibm.db2.jdbc.app.DB2driver  
jdbcTemplatePrefix=jdbc:db2://입니다. 기타 플랫폼에서 DB2에 대한 기본값은 jdbcTemplate=COM.ibm.db2.jdbc.app.DB2driver  
jdbcTemplatePrefix=jdbc:db2입니다.

주: 기타 조합도 작동할 수 있지만 테스트되지는 않았습니다.

- message: 경매 관련 메시지 오브젝트를 삭제합니다.
- msgmemrel: 메시지 구성원 관계 오브젝트를 삭제합니다.

- orders: 주문 오브젝트를 삭제합니다.
- organization: 조직 오브젝트를 삭제합니다.
- pastats: 상품 어드바이저 통계 오브젝트를 삭제합니다.
- pcstats: 상품 비교 통계 오브젝트를 삭제합니다.
- pestats: 상품 탐색기 통계 오브젝트를 삭제합니다.
- policy: 정책 오브젝트를 삭제합니다.
- product\_sets: 상품 설정 오브젝트를 삭제합니다.
- rfq: RFQ 오브젝트를 삭제합니다.
- rma: 반품 항목 오브젝트를 삭제합니다.
- rtnreasons: rtnreason(반품 이유) 오브젝트를 삭제합니다.
- sastats: 판매 보조 기능 통계 오브젝트를 삭제합니다.
- staglog: 스테이징된 오브젝트를 삭제합니다.
- store: 상점 오브젝트를 삭제합니다.
- user: 사용자 오브젝트를 삭제합니다.
- usrtraffic: 사용자 트래픽 로그 오브젝트를 삭제합니다.
- vendor: 공급업체 오브젝트를 삭제합니다.

**type** 삭제하려는 오브젝트의 유형입니다. 데이터베이스 정리 절에 있는 개별 명령을 참조하십시오.

**days** 레코드를 삭제하기 전까지 보관할 최소 일입니다.

**name** (선택) 삭제할 오브젝트의 ID입니다. 이 매개변수는 member가 organization 매개변수의 값으로 지정되고 organization이 type의 값으로 지정된 경우 필수입니다.

#### dbtype

+ (선택) 데이터베이스 유형은 DB2, DB2/iSeries 또는 Oracle입니다. 기본값은 DB2입니다. ▶ 400 기본값은 DB2/iSeries입니다.

**db** 데이터베이스의 이름입니다. ▶ 400 관계형 데이터베이스 디렉토리에 표시된 데이터베이스 이름입니다.


주: 도구상자 JDBC 드라이버를 사용하도록 jdbccustomizer 매개변수를 구성한 경우, 데이터베이스 이름 대신 데이터베이스가 상주하는 시스템의 호스트 이름을 지정하십시오(예: -db hostname.ibm.com).

▶ Oracle

host:port:sid를 사용하십시오. 예: myhost:1521:mydb

#### dbuser

(선택) 스키마를 작성한 관리자 또는 데이터베이스 사이트 운영자의 로그인 ID

입니다. 이 매개변수를 지정하지 않은 경우, 유틸리티를 호출한 사용자의 ID가 사용됩니다.  (필수) WebSphere Commerce 인스턴스와 연관된 사용자 프로파일. 또한 스키마 이름입니다.

### dbpasswd

(선택) DB2 데이터베이스의 dbuser 매개변수가 지정하는 로그인 ID의 암호. 암호를 지정하지 않으면 시스템이 암호를 입력하도록 프롬프트합니다.

### check\_object\_only

(선택) yes를 지정하면, 데이터베이스 정리 유틸리티는 데이터베이스 정리에 의해 영향을 받을 수 있는 모든 하위 테이블과 삭제 제한을 표시합니다. 매개변수를 no(기본값)로 두면 정리 유틸리티는 확인을 수행하지 않습니다.

**force** (선택) force 옵션은 yes 또는 no로 지정할 수 있습니다. 예를 지정하는 경우, 정리 유틸리티는 하위 테이블을 삭제한 다음 상위 테이블을 삭제합니다.


### loglevel


(선택) 데이터베이스 정리 중에 수행할 로그 작성 레벨입니다. 값을 지정하지 않은 경우, 기본 로그 레벨은 0입니다.

- 로그 활동이 기록되지 않도록 지정하려면 0을 입력하십시오. 이것이 기본값입니다.
- 지정된 테이블에 대해서만 DELETE 명령문이 기록되도록 지정하려면 1을 입력하십시오.
- 지정된 테이블 및 모든 하위 테이블에 대해 DELETE 명령문이 기록되도록 지정하려면 2를 입력하십시오. Loglevel 2는 데이터베이스 정리 유틸리티가 상향식 방법을 사용하도록 강제합니다.

+ **log** (선택) 유틸리티가 해당 활동을 기록하는 로그 파일의 경로 및 이름입니다. 이 명령의 발행자는 지정된 경로에 대한 쓰기 권한이 있어야 하며 해당 경로가 이미 있어야 합니다. 이 매개변수를 지정하지 않으면, 다음 log 디렉토리에 dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 로그 파일이 작성됩니다.

+    WC\_installdir/logs

+  WC\_installdir\logs

+  이 매개변수를 지정하지 않으면, dbclean\_dbuser\_yyyy.mm.dd\_hh.mm.ss.zzz.log 로그 파일이 WC\_userdir/instances 디렉토리에 작성됩니다.

### instancexml

상점 삭제 시 사용합니다. instancexml은 installdir/instances/instancexml/xml/instancexml.xml 양식의 절대 파일 이름 값을 가집니다.

주: 이 매개변수는 사용자, 조직, 상점 및 첨부과 같은 파일 자원을 소유하는 일부 오브젝트에 필요합니다.

+ **script** (선택) 저장될 출력 스크립트의 경로 및 이름을 지정합니다. 기본값으로는  
 + `Windows WC_installdir\logs\assetclean_userName_timestamp_.cmd`  
 + `400 /WC_userdir/instances/assetclean_userName_timestamp_.sh`  
 + 및 기타 다른 플랫폼용으로는 `WC_installdir /logs/`  
 + `assetclean_userName_timestamp_.sh`로 저장됩니다.




---

## 데이터베이스 정리

데이터베이스 정리 유틸리티를 사용하여 WebSphere Commerce 데이터베이스에서 사용하지 않거나 쓸모없게 된 오브젝트를 삭제하려면 다음 중 하나를 선택하십시오.

- 계정 오브젝트
- 주소 오브젝트
- ATP 재고(Available to promise inventory) 오브젝트
- 첨부 오브젝트
- Professional 경매 오브젝트
- Professional 경매 로그 오브젝트
- Professional 자동 입찰 로그 오브젝트
- 기본 항목 오브젝트
- Professional 입찰 로그 오브젝트
- 계산 코드 오브젝트
- 캠페인 오브젝트
- 캠페인 통계 오브젝트
- 카탈로그 항목 오브젝트
- 카탈로그 그룹 오브젝트
- 장기 구매 계약 오브젝트
- 쿠폰 오브젝트
- 예상 재고 레코드 오브젝트
- 예상 재고 레코드 정보 오브젝트
- 포럼 메시지 오브젝트
- 서비스 센터 오브젝트
- 재고 조정 코드 오브젝트
- 재고 조정 오브젝트
- 항목 스펙 오브젝트
- 구성원 메시지 오브젝트
- 메시지 오브젝트



- 주문 오브젝트
- 조직 오브젝트
-  상품 어드바이저 통계 오브젝트
-  상품 비교 통계 오브젝트
-  상품 탐색기 통계 오브젝트
- 정책 오브젝트
- 상품 세트 오브젝트
- RFQ 오브젝트
- 반품 항목 오브젝트
- rtnreason(반품 이유) 오브젝트
-  판매 보조 기능 통계 오브젝트
- 스테이징된 오브젝트
- 상점 오브젝트
- 사용자 오브젝트
- 사용자 트래픽 로그 오브젝트
- 공급업체 오브젝트



주: 데이터베이스를 구성하고 PATH 환경 변수를 설정해야 데이터베이스 정리 유틸리티를 사용할 수 있습니다.









---

## WebSphere Commerce 유틸리티에 대한 PATH 환경 변수 설정

주:  이 절은 iSeries 플랫폼에 적용되지 않습니다.

데이터베이스 정리 유틸리티 또는 스테이징 서버를 사용하기 전에 다음 PATH 환경 변수를 설정했는지 확인하십시오.

-    
`WC_installdir/bin` 및 `WC_installdir\sqllib\bin`.
-    
`WC_installdir\bin` 및 `WC_installdir\ora9i\bin`
-    
`WC_installdir\bin` 및 `WC_installdir\sqllib\bin`
-    
`WC_installdir\bin` 및 `WC_installdir\ora9i\bin`
-    
`WC_installdir/bin` 및 `WC_installdir/lpp/db2_08_01`

-    
WC\_installdir/bin 및 /usr/ora9i/bin
-    
WC\_installdir/bin 및 WC\_installdir/IBMdb2/V8.1
-    
WC\_installdir/bin 및 opt/IBM/db2/V8.1
-    
WC\_installdir/bin 및 WC\_installdir/ora9i/bin













## 오브젝트 삭제

이 절에서는 데이터베이스 정리 유틸리티를 사용하여 WebSphere Commerce 데이터베이스에서 사용하지 않거나 쓸모없게 된 특정 오브젝트를 삭제하는 것을 다룹니다.

### 계정 오브젝트 삭제

계정 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
dbclean -object account -type obsolete -db *dbname* -days *daysold* -loglevel *loglevel*
-    
dbclean -object account -type obsolete -db *dbname* -days *daysold* -loglevel *loglevel* -dbtype oracle -dbuser *user* -dbpasswd *password*
- +       
+ . dbclean.sh -object account -type obsolete -db *dbname* -dbuser *user*  
+ -days *daysold* -loglevel *loglevel*
-     
. dbclean.sh -object account -type obsolete -db *dbname* -days *daysold* -loglevel *loglevel* -dbtype oracle -dbuser *user* -dbpasswd *password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

계정 오브젝트에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 주소 오브젝트 삭제

주소 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- ▶ Windows ▶ DB2

```
dbclean -object address -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- ▶ Windows ▶ Oracle

```
dbclean -object address -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2

```
. dbclean.sh -object address -type obsolete -db dbname -dbuser user  
-days daysold -loglevel loglevel
```

- ▶ AIX ▶ Solaris ▶ Oracle

```
. dbclean.sh -object address -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

주소 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## ATP 재고(Available to promise inventory) 오브젝트 삭제

ATP 재고(Available to promise inventory) 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- ▶ Windows ▶ DB2

```
dbclean -object atp_inventory -type typename -db dbname -loglevel  
loglevel
```

- ▶ Windows ▶ Oracle

```
dbclean -object atp_inventory -type typename -db dbname -logLevel  
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2

```
. dbclean.sh -object atp_inventory -type typename -db dbname  
-dbuser user -loglevel loglevel
```

- ▶ AIX ▶ Solaris ▶ Oracle  
`. dbclean.sh -object atp_inventory -type typename -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

ATP 재고 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 첨부 오브젝트 삭제

첨부 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- ▶ Windows ▶ DB2  
`dbclean -object attachment -type obsolete -db dbname -days daysold  
 -loglevel loglevel`

- ▶ Windows ▶ Oracle  
`dbclean -object attachment -type obsolete -db dbname -days daysold  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2  
`. dbclean.sh -object attachment -type obsolete -db dbname -dbuser  
 user -days daysold -loglevel loglevel`

- ▶ AIX ▶ Solaris ▶ Oracle  
`. dbclean.sh -object attachment -type obsolete -db dbname -days  
 daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd  
 password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

주: 첨부의 `-type`는 또한 `markfordelete`도 될 수 있습니다. 이 경우에는 `-instancexml`이 필요합니다. 예를 들면,

```
dbclean.sh -object attachment -type markfordelete -instancexml  

  /WC_userdir/instances/instance/xml/instance.xml  

  -db dbname -dbuser instance -loglevel
```

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

첨부 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 경매 오브젝트 삭제

경매 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- Windows DB2

```
dbclean -object auction -type typename -db dbname -days daysold  
-loglevel loglevel
```

- Windows Oracle

```
dbclean -object auction -type typename -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+

- AIX 400 Solaris Linux DB2

```
. dbclean.sh -object auction -type typename -db dbname -dbuser user  
-days daysold -loglevel loglevel
```

- AIX Solaris Oracle

```
. dbclean.sh -object auction -type typename -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb 주: -type 매개변수의 경우 *settlement\_closed*을 지정하여 경매 레코드가 완료되었음을 표시하거나, *retracted*를 지정하여 경매가 유찰되었음을 표시할 수 있습니다.

2. *dbclean.log\_yyyy.mm.dd\_hh.mm.ss.zzz* 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

경매 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 경매 로그 오브젝트 삭제

경매 로그 오브젝트를 삭제하려면 다음을 수행하십시오.









1. 다음을 입력하십시오.

- Windows DB2

```
dbclean -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- Windows Oracle

```
dbclean -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- + •       
 + . dbclean.sh -object auctionlog -type obsolete -db *dbname* -dbuser  
 + *user* *days* *daysold* -loglevel *loglevel*
-     
 . dbclean.sh -object auctionlog -type obsolete -db *dbname* -dbuser  
*user* -days *daysold* -loglevel *loglevel* -dbtype oracle -dbuser *user*  
 -dbpasswd *password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. dbclean\_YYYY.MM.DD\_HH.MM.SS.ZZZ.LOG 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

경매 로그 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## autobid 로그 오브젝트 삭제

자동 입찰 로그 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
 dbclean -object autobidlog -type obsolete -db *dbname* -days *daysold*  
 -loglevel *loglevel*
-    
 dbclean -object autobidlog -type obsolete -db *dbname* -days *daysold*  
 -loglevel *loglevel* -dbtype oracle -dbuser *user* -dbpasswd *password*
- + •       
 + . dbclean.sh -object autobidlog -type obsolete -db *dbname* -dbuser  
 + *user* -days *daysold* -loglevel *loglevel*
-     
 . dbclean.sh -object autobidlog -type obsolete -db *dbname* -days  
*daysold* -loglevel *loglevel* -dbtype oracle -dbuser *user* -dbpasswd  
*password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. dbclean\_YYYY.MM.DD\_HH.MM.SS.ZZZ.LOG 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

autobid 로그 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 기본 항목 오브젝트 삭제

동일한 이름과 설명을 가진 상품의 일반 계열에 대한 정보를 담고 있는 기본 항목 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object baseitem -type obsolete -db dbname -loglevel  
loglevel
```

-  

```
dbclean -object baseitem -type obsolete -db dbname -loglevel  
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+

-     

```
. dbclean.sh -object baseitem -type obsolete -db dbname -dbuser  
user -loglevel loglevel
```

-   

```
. dbclean.sh -object baseitem -type obsolete -db dbname -loglevel  
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

기본 항목 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 입찰 로그 오브젝트 삭제

입찰 로그 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object bidlog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```




-  

```
dbclean -object bidlog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+

-     

```
. dbclean.sh -object bidlog -type obsolete -db dbname -dbuser user  
-days daysold -loglevel loglevel
```

-     
`. dbclean.sh -object bidlog -type obsolete -db dbname -days daysold  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb



2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.



bid 로그 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.






## 계산 코드 오브젝트 삭제




계산 코드 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object calculation_code -type obsolete -db dbname  
 -loglevel loglevel`

-    
`dbclean -object calculation_code -type obsolete -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

- +       
 + `. dbclean.sh -object calculation_code -type obsolete -db dbname  
 + -dbuser user -loglevel loglevel`

-     
`. dbclean.sh -object calculation_code -type obsolete -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.













계산 코드 오브젝트에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 캠페인 오브젝트 삭제

캠페인 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.



-    
`dbclean -object cpnlog -type all -db dbname -loglevel loglevel`
-    
`dbclean -object cpnlog -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object cpnlog -type all -db dbname -dbuser user  
 -loglevel loglevel`
- +     
`. dbclean.sh -object cpnlog -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

캠페인 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 캠페인 통계 오브젝트 삭제

캠페인 통계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -table cpnstats -type all -db dbname -loglevel loglevel`
-    
`dbclean -table cpnstats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object cpnstats -type all -db dbname -dbuser user  
 -loglevel loglevel`
- +     
`. dbclean.sh -object cpnstats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

캠페인 통계 레코드를 CPGNSTATS 테이블에서 삭제하는 것에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 카탈로그 항목 오브젝트 삭제

카탈로그 항목 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object catentry -type typename -db dbname -days daysold -loglevel loglevel`
-    
`dbclean -object catentry -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-       
`. dbclean.sh -object catentry -type typename -db dbname -dbuser user -days daysold -loglevel loglevel`
-     
`. dbclean.sh -object catentry -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

주: -type 매개변수의 경우 *without\_orderitems*을 지정하여 주문 항목을 참조하지 않는 카탈로그 항목을 표시하거나 *without\_orderitems\_iitems*을 지정하여 주문 항목 또는 관심 목록 항목을 참조하지 않는 카탈로그 항목을 표시할 수 있습니다.



2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.











카탈로그 항목 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 카탈로그 그룹 오브젝트 삭제

카탈로그 그룹 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object catalog_group -type obsolete -db dbname -loglevel loglevel`

-    
`dbclean -object catalog_group -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object catalog_group -type obsolete -db dbname -dbuser user -loglevel loglevel`
- +     
`. dbclean.sh -object catalog_group -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

카탈로그 그룹 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 장기 구매 계약 오브젝트 삭제






장기 구매 계약 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object contract -type obsolete -db dbname -loglevel loglevel`
-    
`dbclean -object contract -type obsolete -db database -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object contract -type obsolete -db dbname -dbuser user -loglevel loglevel`
- +     
`. dbclean.sh -object contract -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

- 주:      `-dbor DATABASE` 매개 변수의 경우, 세 개의 데이터베이스 테이블(`productset`, `tradeposn` 또는 `trading`) 중 하나를 지정해야 합니다.

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

장기 구매 계약 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 쿠폰 오브젝트 삭제

쿠폰 오브젝트를 삭제하려면 다음을 수행하십시오.






1. 다음을 입력하십시오.




-  

```
dbclean -object coupon_promotion -type expired -db dbname -days daysold -loglevel loglevel
```

-  

```
dbclean -object coupon_promotion -type expired -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+  
•       
. `dbclean.sh -object coupon_promotion -type expired -db dbname -dbuser user -days daysold -loglevel loglevel`

-   

```
. dbclean.sh -object coupon_promotion -type expired -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

쿠폰 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.











## 예상 재고 레코드 오브젝트 삭제

예상 재고 레코드 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object expected_inventory_records -type obsolete -db dbname -days daysold -loglevel loglevel
```

-    
`dbclean -object expected_inventory_records -type obsolete -db  
 dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user  
 -dbpasswd password`
- +  
+  
+  
•       
`. dbclean.sh -object expected_inventory_records -type obsolete -db  
 dbname -dbuser user -days daysold -loglevel loglevel`
-     
`. dbclean.sh -object expected_inventory_records -type obsolete -db  
 dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user  
 -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`












2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

예상 재고 레코드 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 예상 재고 레코드 오브젝트 정보 삭제

예상 재고 레코드 오브젝트 정보를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object expected_inventory_record_details -type obsolete  
 -db dbname -loglevel loglevel`
-    
`dbclean -object expected_inventory_record_details -type obsolete  
 -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd  
 password`
- +  
+  
+  
•       
`. dbclean.sh -object expected_inventory_record_details -type  
 obsolete -db dbname -dbuser user -loglevel loglevel`
-     
`. dbclean.sh -object expected_inventory_record_details -type  
 obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user  
 -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

예상 재고 레코드 오브젝트에 대한 정보 삭제에 관한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 포럼 메시지 오브젝트 삭제

포럼 메시지 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object forummsg -type obsolete -db dbname -days daysold
-loglevel loglevel
```

-  

```
dbclean -object forummsg -type obsolete -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-     

```
. dbclean.sh -object forummsg -type obsolete -db dbname -dbuser
user -days daysold -loglevel loglevel
```

-   

```
. dbclean.sh -object forummsg -type obsolete -db dbname -days
daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd
password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

포럼 메시지 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.











## 서비스 센터 오브젝트 삭제

서비스 센터 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object fulfillment_center -type obsolete -db dbname
-loglevel loglevel
```

-    
`dbclean -object fulfillment_center -type obsolete -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +  
 +  
 +  
-       
`. dbclean.sh -object fulfillment_center -type obsolete -db dbname  
 -dbuser user -loglevel loglevel`
-     
`. dbclean.sh -object fulfillment_center -type obsolete -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

서비스 센터 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 재고 코드 조정 오브젝트 삭제

재고 코드 조정 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object inventory_adjustment_codes -type obsolete -db  
 dbname -loglevel loglevel`
-    
`dbclean -object inventory_adjustment_codes -type obsolete -db  
 dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd  
 password`
- +  
 +  
 +  
-       
`. dbclean.sh -object inventory_adjustment_codes -type obsolete -db  
 dbname -dbuser user -loglevel loglevel`
-     
`. dbclean.sh -object inventory_adjustment_codes -type obsolete -db  
 dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd  
 password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

재고 코드 조정 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 재고 조정 오브젝트 삭제

재고 조정 오브젝트를 삭제하려면 다음을 수행하십시오.






1. 다음을 입력하십시오.




-  

```
dbclean -object inventory_adjustments -type obsolete -db dbname
-days daysold -loglevel loglevel
```

-  

```
dbclean -object inventory_adjustments -type obsolete -db dbname
-days daysold -loglevel loglevel -dbtype oracle -dbuser user
-dbpaswd password
```

+  
+  
+  
•       
. `dbclean.sh -object inventory_adjustments -type obsolete -db`  
`dbname -dbuser user -days daysold -loglevel loglevel`

-   

```
. dbclean.sh -object inventory_adjustments -type obsolete -db  
dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user  
-dbpaswd password
```

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

재고 조정 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 지정된 항목 정보 오브젝트 삭제











지정된 항목 정보 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-  

```
dbclean -object itemspecification -type obsolete -db dbname
-loglevel loglevel
```



-    
`dbclean -object itemspecification -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object itemspecification -type obsolete -db dbname -dbuser user -loglevel loglevel`
- +     
`. dbclean.sh -object itemspecification -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

지정된 항목 정보 오브젝트에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 구성원 메시지 관계 오브젝트 삭제

구성원 메시지 관계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel`
-    
`dbclean -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
`. dbclean.sh -object msgmemrel -type obsolete -db dbname -dbuser user -days daysold -loglevel loglevel`
- +     
`. dbclean.sh -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

메시지 구성원 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 메시지 오브젝트 삭제

메시지 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object message -type obsolete -db dbname -days daysold -loglevel loglevel`
-    
`dbclean -object message -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-       
`. dbclean.sh -object message -type obsolete -db dbname -dbuser user -days daysold -loglevel loglevel`
-     
`. dbclean.sh -object message -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb










2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

메시지 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 주문 오브젝트 삭제

주문 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object order -type typename -db dbname -days daysold -loglevel loglevel`
-    
`dbclean -object order -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-       
`. dbclean.sh -object order -type typename -db dbname -dbuser user -days daysold -loglevel loglevel`

- ▶ AIX ▶ Solaris ▶ Oracle

```
. dbclean.sh -object order -type typename -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

주: -type 매개변수에 대해서는, 다음 여섯 가지 completed 유형 중 하나를 지정하여 완료된 주문을 표시할 수 있습니다. canceled는 취소된 주문을 표시하고, shipped는 운송된 주문을 표시하며, deposited는 예치된 주문을 표시하며, stale\_guest는 게스트 고객이 실효(stale)한 주문을 표시하며, stale\_non\_guest는 게스트 아닌 고객이 실효한 주문을 표시합니다.

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

주문 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 조직 오브젝트 삭제

조직 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- ▶ Windows ▶ DB2

```
dbclean -object organization -type specified -db dbname -loglevel
loglevel -name organizationid -instancexml WC_installdir/
instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

- ▶ Windows ▶ Oracle

```
dbclean -object organization -type specified -db dbname -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password -name
organizationid -instancexml WC_installdir/instances/
INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

+  
+  
+  
+

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2

```
. dbclean.sh -object organization -type specified -db dbname
-dbuser user -loglevel loglevel -name organizationid -instancexml
WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

- ▶ AIX ▶ Solaris ▶ Oracle

```
. dbclean.sh -object organization -type specified -db dbname
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
-name organizationid -instancexml WC_installdir/instances/
INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb 주: -type 매개변수의 경우 organization을 지정하여 모든 조직 레코드를 표시할 수 있습니다.

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.



조직 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.






## 상품 어드바이저 통계 오브젝트 삭제




상품 어드바이저 통계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
dbclean -object pastats -type all -db *dbname* -loglevel *loglevel*

-    
dbclean -object pastats -type all -db *dbname* -loglevel *logLevel*  
-dbtype oracle -dbuser *user* -dbpasswd *password*

-       
+ . dbclean.sh -object pastats -type all -db *dbname* -dbuser *user*  
+ -loglevel *loglevel*  
+

-     
. dbclean.sh -object pastats -type all -db *dbname* -loglevel *loglevel*  
-dbtype oracle -dbuser *user* -dbpasswd *password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.











상품 어드바이저 통계 오브젝트에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 상품 비교 통계 오브젝트 삭제

상품 비교(상품 어드바이저) 통계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
dbclean -object pcstats -type all -db *dbname* -loglevel *loglevel*

-    
`dbclean -object pcstats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`
- +  
+  
+
  -       
`. dbclean.sh -object pcstats -type all -db dbname -dbuser user  
 -loglevel loglevel`
  -     
`. dbclean.sh -object pcstats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

상품 비교(상품 어드바이저) 통계 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 상품 탐색기 통계 오브젝트 삭제

상품 탐색기(상품 어드바이저) 통계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object pestats -type all -db dbname -loglevel loglevel`
-    
`dbclean -object pestats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`
- +  
+  
+
  -       
`. dbclean.sh -object pestats -type all -db dbname -dbuser user  
 -loglevel loglevel`
  -     
`. dbclean.sh -object pestats -type all -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb







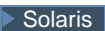





2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

상품 탐색기(상품 어드바이저) 통계 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 정책 오브젝트 삭제

정책 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object policy -type obsolete -db dbname -days daysold -loglevel loglevel`
-    
`dbclean -object policy -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
+ `. dbclean.sh -object policy -type obsolete -db dbname -dbuser user -days daysold -loglevel loglevel`  
+
-     
`. dbclean.sh -object policy -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`










2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.




정책 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 상품 세트 오브젝트 삭제

상품 세트 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object product_sets -type obsolete -db dbname -loglevel loglevel`
-    
`dbclean -object product_sets -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +       
+ `. dbclean.sh -object product_sets -type obsolete -db dbname -dbuser user -loglevel loglevel`  
+

-     
`. dbclean.sh -object product_sets -type obsolete -db dbname  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.



상품 세트 오브젝트에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## RFQ 오브젝트 삭제






RFQ 오브젝트를 삭제하려면 다음을 수행하십시오.




1. 다음을 입력하십시오.

-    
`dbclean -table rfq -type obsolete -db dbname -loglevel loglevel`

-    
`dbclean -object rfq -type obsolete -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`

+  
+  
+

-       
`. dbclean.sh -object rfq -type obsolete -db dbname -dbuser user  
 -loglevel loglevel`

-     
`. dbclean.sh -object rfq -type obsolete -db dbname -loglevel  
 loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

RFQ 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 반품 항목 오브젝트 삭제

반품 항목 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object rma -type typename -db dbname -days daysold  
 -loglevel loglevel`
-    
`dbclean -object rma -type typename -db dbname -days daysold  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
- +  
+  
+  
-       
`. dbclean.sh -object rma -type typename -db dbname -dbuser user  
 -days daysold -loglevel loglevel`
-     
`. dbclean.sh -object rma -type typename -db dbname -days daysold  
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`

주: `-type` 매개변수에는, `abandoned`를 지정하여 포기(`abandon`)된 레코드를 표시하거나, `canceled`를 지정하여 취소된 레코드를 표시하거나, `not_approved`를 지정하여 거부된 레코드를 표시하거나, `approved_or_partly_approved`를 지정하여 승인 또는 부분 승인된 레코드를 표시하거나, `completed`을 지정하여 완료된 레코드를 표시할 수 있습니다.





2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

반품 항목 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.









## 반품 이유 오브젝트 삭제

상품에 대한 고객의 불만족 이유 또는 단순한 반품 이유 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object rtnreason -type obsolete -db dbname -loglevel  
 loglevel`
-    
`dbclean -object rtnreason -type obsolete -db dbname -loglevel  
 loglevel -dbtype oracle -dbuser user -dbpasswd password`



- + •       
 + . dbclean.sh -object rtnreason -type obsolete -db *dbname* -dbuser  
 + *user* -loglevel *loglevel*
-     
 . dbclean.sh -object rtnreason -type obsolete -db *dbname* -loglevel  
*loglevel* -dbtype oracle -dbuser *user* -dbpasswd *password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. dbclean\_YYYY.MM.DD\_HH.MM.SS.ZZZ.LOG 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

반품 이유 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 판매 보조 기능 통계 오브젝트 삭제

판매 보조 기능(상품 어드바이저) 통계 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
 dbclean -object sastats -type all -db *dbname* -loglevel *loglevel*
-    
 dbclean -object sastats -type all -db *dbname* -loglevel *loglevel*  
 -dbtype oracle -dbuser *user* -dbpasswd *password*
- + •       
 + . dbclean.sh -object sastats -type all -db *dbname* -dbuser *user*  
 + -loglevel *loglevel*
-     
 . dbclean.sh -object sastats -type all -db *dbname* -loglevel *loglevel*  
 -dbtype oracle -dbuser *user* -dbpasswd *password*

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. dbclean\_YYYY.MM.DD\_HH.MM.SS.ZZZ.LOG 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

판매 보조 기능(상품 어드바이저) 통계 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 스태이징된 오브젝트 삭제

스태이징된 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object staglog -type obsolete -db dbname -days daysold  
-loglevel loglevel`
-    
`dbclean -object staglog -type obsolete -db dbname -dbuser user -days  
daysold -loglevel loglevel -dbtype oracle -dbuser <user> -dbpasswd  
<password>`
-       
`. dbclean.sh -object staglog -type obsolete -db dbname -dbuser user  
-days daysold -loglevel loglevel`
-     
`. dbclean.sh -object staglog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb





2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.






스태이징된 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.




## 상점 오브젝트 삭제

상점 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object store -type specified -db dbname -loglevel loglevel  
-name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/  
xml/INSTANCE_NAME.xml`
-    
`dbclean -object store -type specified -db dbname -loglevel loglevel  
-dbtype oracle -dbuser user -dbpasswd password -name storeid  
-instancexml WC_installdir/instances/INSTANCE_NAME/xml/  
INSTANCE_NAME.xml`

- + •     

```
. dbclean.sh -object store -type specified -db dbname -dbuser user
-loglevel loglevel -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- + •   

```
. dbclean.sh -object store -type specified -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

2. *dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log* 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

상점 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 사용자 오브젝트 삭제


사용자 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- + •  

```
dbclean -object user -type typename -db dbname -days daysold
-loglevel loglevel -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- + •  

```
dbclean -object user -type typename -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
-instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- + •     

```
. dbclean.sh -object user -type typename -db dbname -dbuser user
-days daysold -loglevel loglevel -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- + •   

```
. dbclean.sh -object user -type typename -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
-instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb

주: -type 매개변수에는 guest를 지정하여 게스트 고객을 표시하거나 registered를 지정하여 등록 고객을 표시할 수 있습니다.

2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

사용자 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 사용자 트래픽 로그 오브젝트 삭제

사용자 트래픽 로그 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

- Windows DB2

```
dbclean -object usrtraffic -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- Windows Oracle

```
dbclean -object usrtraffic -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

+  
+  
+  
• AIX 400 Solaris Linux DB2  
. dbclean.sh -object usrtraffic -type obsolete -db dbname -dbuser  
user -days daysold -loglevel loglevel

- AIX Solaris Oracle

```
. dbclean.sh -object usrtraffic -type obsolete -db dbname -days  
daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd  
password
```

Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: myhost:1521:mydb













2. dbclean\_yyyy.mm.dd\_hh.mm.ss.zzz.log 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

사용자 트래픽 로그 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 공급업체 오브젝트 삭제

공급업체 오브젝트를 삭제하려면 다음을 수행하십시오.

1. 다음을 입력하십시오.

-    
`dbclean -object vendor -type obsolete -db dbname -loglevel loglevel`
-    
`dbclean -object vendor -type obsolete -db dbname -loglevel loglevel  
 -dbtype oracle -dbuser user -dbpasswd password`
-       
`. dbclean.sh -object vendor -type obsolete -db dbname -dbuser user  
 -loglevel loglevel`
-     
`. dbclean.sh -object vendor -type obsolete -db dbname -loglevel  
 loglevel -dbtype oracle -dbuser user -dbpasswd password`

Oracle 데이터베이스 이름으로 `host:port:sid`를 사용하십시오.

예: `myhost:1521:mydb`






2. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

공급업체 오브젝트 삭제에 대한 추가적인 예는 152 페이지의 『오브젝트 삭제 예』를 참조하십시오.

## 데이터베이스 정리 유틸리티에 새 구성 추가하는 법

새 구성을 데이터베이스 정리 유틸리티에 추가하려면 다음 구문을 참조로 사용하십시오. 예를 들어, `o1` 오브젝트는 `R1` 테이블로 구성되어 있고, `R1` 테이블은 `col1`, `col2`, `lastupdate` 및 `col3` 열을 포함하고 있습니다. `col1 > 10`이고 `lastupdate`가 `n`일 이전인 오브젝트를 모두 삭제하도록 데이터베이스 정리 유틸리티를 구성하려면, 다음을 수행하십시오.



1.     DB2 명령 프롬프트를 여십시오.
2. 다음을 입력하십시오.  
`db2 insert into cleanconf (objectname, type, statement, namearg, sequence, daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (days(CURRENT TIMESTAMP) - days(lastupdate)) > ?', 'no', 1, 'yes')`
1.  다음 SQL 명령문을 실행하십시오.  
`insert into cleanconf (objectname, type, statement, namearg, sequence, daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (days(CURRENT TIMESTAMP) - days(lastupdate)) > ?', 'no', 1, 'yes')`













## Oracle

1. SQLPlus 명령창을 여십시오.
2. 다음을 입력하십시오.  
`insert into cleanconf (objectname, type, statement, namearg, sequence, daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (sysdate - lastupdate) > ?', 'no', 1, 'yes')`

여기에서 ?는 다음 명령행에서 `-days` 매개변수로 바뀝니다. 'no'는 명령문에 이름 매개변수를 사용하지 않음을 표시합니다. 'yes'는 명령문에 `-days` 매개변수를 사용함을 표시합니다. 'obsolete'는 o1 오브젝트에 대한 정리 유형을 나타냅니다. 다른 단어를 사용할 수도 있지만 데이터베이스 정리 유틸리티를 호출할 때 `-type` 인수에서 동일한 단어를 사용해야 합니다.

## yes

데이터베이스 정리 유틸리티를 호출하여 새 테이블에서 2일 동안 존재한 레코드를 정리하려면, 다음을 입력하십시오.

-    
`dbclean -object o1 -db dbname -type obsolete -days 2 -loglevel 1`
-    
`dbclean -object o1 -db dbname -type obsolete -days 2 -loglevel 1 -dbtype oracle -dbuser user -dbpasswd password`
-       
`. dbclean.sh -object o1 -db dbname -dbuser user -type obsolete -days 2 -loglevel 1`
-     
`. dbclean.sh -object o1 -db dbname -type obsolete -days 2 -loglevel 1 -dbtype oracle -dbuser user -dbpasswd password`

주: Oracle `dbname` 매개변수로는 `host:port:sid`를 사용하십시오.













예: `myhost:1521:mydb`

---













## 오브젝트 삭제 예

다음은 데이터베이스 정리 유틸리티에서 선택 매개변수를 사용하여 데이터베이스 테이블에서 오브젝트를 삭제하는 예입니다. 매개변수에 대한 자세한 정보는 데이터베이스 정리 유틸리티 명령을 참조하십시오.





예 1: 삭제 제한 매개변수를 지정하는 테이블을 검증하려면 다음을 입력하십시오.









-    
`dbclean -object objectname -type typename -db database -check_object_only yes`
-    
`dbclean -object objectname -type typename -db host:port:sid -check_object_only yes -dbtype oracle -dbuser user -dbpasswd <password>`
- +       
`. dbclean.sh -object objectname -type typename -db database -dbuser user -check_object_only yes`
- +     
`. dbclean.sh -object objectname -type typename -db host:port:sid -check_object_only yes -dbtype oracle -dbuser user -dbpasswd password`

예 2 삭제 제한 매개변수를 지정한 테이블에 대해 force 옵션을 사용하려면 다음을 입력하십시오.

-    
`dbclean -object objectname -type typename -db database -days daysold -loglevel loglevel -force yes`
-    
`dbclean -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -force yes -dbtype oracle -dbuser user -dbpasswd <password>`
- +       
`. dbclean.sh -object objectname -type typename -db database -dbuser user -days daysold -loglevel loglevel -force yes`
- +     
`. dbclean.sh -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -force yes -dbtype oracle -dbuser user -dbpasswd <password>`

예 3 기본 로그 파일 이름은 항상 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log`의 변형입니다. 로그 파일 이름을 지정하려면 다음을 입력하십시오.

-    
`dbclean -object objectname -type typename -db database -days daysold -loglevel loglevel -log logfilename`
-    
`dbclean -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -log logfilename -dbtype oracle -dbuser user -dbpasswd <password>`

- + •       
+ . dbclean.sh -object *objectname* -type *typename* -db *database* -dbuser  
+ *user* -days *daysold* -loglevel *loglevel* -log *logfile*
-     
. dbclean.sh -object *objectname* -type *typename* -db *host:port:sid* -days  
*daysold* -loglevel *loglevel* -log *logfile* -dbtype *oracle* -dbuser  
*user* -dbpasswd <password>



---

## 제 7 장 성능

WebSphere Commerce는 여러 제품 간의 복잡한 상호작용 역할을 수행합니다. 각 제품에는 제품의 고유 성능 특성이 있고 여러 구성요소의 상호작용 내에는 올바르지 않은 구성이나 부족한 자원으로 인해 성능이 영향을 받는 여러 경우가 있습니다. 성능은 적절한 방식으로 다음과 같은 유형의 요청을 처리하는 것을 목표로 합니다.

- 복수 고객 요청 처리
- WebSphere Commerce 데이터베이스의 데이터 액세스
- 데이터를 웹 페이지 형태로 포맷팅
- 구매자 브라우저로 응답 리턴

WebSphere Commerce를 최적화하려면 다음 구성요소를 고려하십시오.

- 하드웨어

시스템이 *WebSphere Commerce* 설치 안내서에 설명되어 있는 최소 시스템 요구사항을 만족하는지 확인하십시오.

다수의 동시 사용자를 갖는 프로덕션 환경에서는 복수 CPU가 성능을 향상시키는데 도움이 됩니다. 빠른 CPU를 사용하면 대부분의 작업 속도를 높일 수 있습니다.

- 데이터베이스

DB2 Database 성능 조정에 대해서는 Redbook *DB2/UDB WebSphere* 성능 조정 안내서(<http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg246417.html?Open>에서 사용 가능)를 참조하십시오.

최대 데이터베이스 연결 풀 크기가 모든 동시 태스크(예: HTTP 연결, 스케줄러 스레드 등)를 모두 처리하기에 충분한지 확인하십시오.

- WebSphere Commerce

서버가 I/O 바인드되어 있는지 확인하십시오. 많은 파일 액세스 또는 네트워크 액세스가 발생하는 경우 WebSphere Commerce 시스템 성능에 나쁜 영향을 끼칠 수 있습니다. 예를 들어, 모든 로그 작성 및 추적 기능이 켜져 있는 경우, 시스템은 대부분의 시간을 워크로드를 처리하는 대신 데이터를 디스크에 쓰는데 소비하게 됩니다.

87 페이지의 제 5 장 『동적 캐시 작성』에 설명되어 있는 동적 캐시를 사용하십시오.

서버 기반 세션 관리를 사용하고 있는 경우 *WebSphere Application Server 5.0* 성능 조정 안내서의 지침을 참조하십시오. 이 안내서는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에서 찾을 수 있습니다. 모니터 및 문제점 해결 --> 성능 --> 성능 조정을 펼치십시오.

- WebSphere Application Server
  - 시스템 성능 조정 방법에 관한 지침으로는 *WebSphere Application Server 5.0* 성능 조정 안내서를 사용하십시오. 이 안내서는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에서 찾을 수 있습니다. 모니터 및 문제점 해결 --> 성능 --> 성능 조정을 펼치십시오.
- 기타 고려사항
  - WebSphere Datasource(최소 및 최대 연결 풀 크기, 명령문 캐시 크기)
  - 웹 사이트 설계
  - 보안(구성, 시간 종료, 인증 및 액세스 제어)
  - 웹 서버 관련사항(프로세스 처리, 자원 사용법, 빠른 응답 캐시 액셀러레이터)
  - WebSphere 엔진 관련사항(Java Virtual Machine 또는 JVM, 전송 큐, JSP 파일의 캐시, EJB 컨테이너)
  - WebSphere Commerce 세션 관리(캐시, 메모리에 세션 저장 또는 데이터베이스에 세션 저장)
  - WebSphere Application Server 세션 관리(인메모리 세션 계수 설정, 오버플로우 허용, 시간 종료 간격 및 분산 환경 설정)
  - NFS(Network File System) 성능 조정(파일 서버 성능 조정)

---

## WebSphere Commerce PMI 모듈을 사용한 성능 모니터링

WebSphere Application Server PMI(Performance Monitoring Infrastructure)를 사용하여 WebSphere Commerce 시스템 성능을 모니터링할 수 있습니다. 이전 WebSphere Commerce 성능 모니터의 기능이 이제는 WebSphere Commerce PMI 모듈로 대체되었습니다. 따라서 WebSphere Application Server에 있는 다양한 도구를 이용하여 성능 데이터를 모니터링하고 분석할 수 있습니다. 사이트 운영자는 도구를 통해 수집된 정보를 사용하여 성능 문제점을 발견하고 성능 경향을 분석할 수 있습니다. 로컬 또는 원격 시스템에서 도구를 사용하여 WebSphere Commerce Application Server의 성능을 측정할 수 있습니다.

WebSphere Commerce 응용프로그램 서버는 URL, 태스크 및 JSP에 대한 통계를 수집합니다. 각 데이터 키마다 다음 정보를 제공하는 카운터 세트가 연관되어 있습니다.

- 평균 응답 시간
- 마지막 응답 시간
- 최소 응답 시간
- 최대 응답 시간
- 히트 수
- 총 응답 시간

- 표준 편차

PMI를 설정하려면 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "성능 모니터링" 절을 참조하십시오.

## PMI 모듈 설정

PMI(Performance Monitoring Infrastructure)는 클라이언트-서버 아키텍처를 사용합니다. 서버는 다양한 WebSphere Application Server 구성요소에서 성능 데이터를 수집합니다. 클라이언트는 하나 이상의 서버에서 성능 데이터를 검색하고 처리합니다.

WebSphere Application Server 서버는 PMI 데이터를 메모리에 수집합니다. 이 데이터는 Servlet 응답 시간 및 데이터 연결 풀 사용량과 같은 카운터로 구성되어 있습니다. 그러면 웹 클라이언트, Java 클라이언트 또는 JMX 클라이언트를 사용하여 데이터 위치를 검색합니다. WebSphere Application Server는 성능 데이터를 표시하고 모니터링하는 Java 클라이언트인 Tivoli® Performance Viewer를 제공합니다.

성능 모니터링 도구를 사용하려면 다음과 같이 WebSphere Commerce PMI 모듈을 설정해야 합니다.

1. PMI 인터페이스를 통해 성능 데이터를 모니터링하려면 먼저 WebSphere Application Server 관리 콘솔에서 성능 모니터링 서비스를 사용 가능 상태로 만든 다음 서버를 다시 시작해야 합니다. 자세한 방법은 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "관리 콘솔을 통해 응용프로그램 서버에서 PMI 서비스 사용"을 참조하십시오.
2. 관리 콘솔을 통해 NodeAgent의 성능 모니터링 서비스를 사용 가능 상태로 만들어도 PMI 인터페이스를 통한 성능 데이터를 모니터링할 수 있습니다. 자세한 방법은 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "관리 콘솔을 통해 NodeAgent에서 PMI 서비스 사용"을 참조하십시오.
3. 위의 설정을 마치면 데이터를 수집할 수 있습니다.

사용 가능한 데이터 카운터를 판별하는 모니터링 레벨은 서버를 다시 시작하지 않고 동적으로 설정할 수 있습니다. 이것은 다음 세 가지 방법 중 하나로 수행할 수 있습니다.

- WebSphere Application Server 관리 콘솔을 통한 데이터 수집을 사용하십시오.
- Tivoli Performance Viewer(이전의 자원 분석기)를 통한 성능 모니터링 서비스를 사용합니다.
- 명령행을 이용하여 성능 모니터링 서비스를 사용합니다.

자세한 단계는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에서 위의 테스트에 해당하는 절을 참조하십시오.

#### 4. 성능 데이터를 모니터링하고 분석하십시오.

다음의 여러 도구를 사용하여 데이터를 모니터링하고 분석할 수 있습니다.

- a. Tivoli Performance Viewer를 사용하여 성능 데이터를 모니터링합니다. 이 도구는 WebSphere Application Server에 포함되어 있습니다.
- b. 다른 Tivoli 모니터링 도구를 사용하여 성능 데이터를 모니터링하십시오.
- c. 사용자 개발 모니터링 도구를 사용하여 성능 데이터를 모니터링하십시오. 사용자 고유의 응용프로그램을 작성하여 성능 데이터를 모니터링하십시오.
- d. 타사 모니터링 도구를 사용하여 성능 데이터를 모니터링하십시오.

자세한 정보는 WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)에서 위의 태스크에 해당하는 절을 참조하십시오.

## Tivoli Performance Viewer 시작

Tivoli Performance Viewer는 WebSphere Application Server 5.0과 함께 제공되는 WebSphere Application Server용 성능 모니터입니다. 이 표시기는 관리 서버를 정기적으로 폴링함으로써 성능 데이터를 검색합니다. 데이터는 계속 수집되며 필요에 따라 표시기 내에서 검색됩니다. 자원 분석기를 실행하고 WebSphere Application Server 관리 콘솔에서 데이터를 수집할 데이터 레벨을 지정하십시오. 그래픽 인터페이스를 사용하여 테이블 또는 도표로 데이터를 검색하여 보거나 데이터를 로그 파일에 저장하십시오. 표시기를 사용하여 Commerce PMI 뿐만 아니라 WebSphere Commerce와 관련된 다른 일반 WebSphere Application Server 성능 데이터를 모니터링할 수 있습니다.

추가 정보는 WebSphere Application Server 정보 센터의 "Tivoli Performance Viewer(이전의 자원 분석기)를 이용한 성능 모니터링" 주제를 참조하십시오.

WebSphere Commerce에 대해 Tivoli Performance Viewer를 시작할 때 WebSphere Commerce 관련 카운터를 측정하기 위해 `WC_instance_name` Application Server에 대한 SOAP 또는 RMI 포트 번호를 지정해야 합니다. 그렇지 않으면, 기본적으로 Tivoli Performance Viewer가 대신 `server1` Application Server에 연결을 시도합니다. `WC_instance_name` 응용프로그램 서버의 SOAP 또는 RMI 포트 번호를 결정하는 한 가지 방법은 `WC_instance_name`의 `System.Out` 로그 파일을 조사하는 것입니다. 서버가 시작할 때 SOAP 또는 RMI 포트 번호가 표시됩니다.

다음은 Tivoli Performance Viewer를 시작하는 단계입니다.

1. 관리 콘솔을 통한 PMI 서비스를 사용 가능하게 하십시오.

2. Tivoli Performance Viewer를 시작하십시오. 다음 두 가지 방법으로 수행할 수 있습니다.

a. 명령행에서 성능 모니터링을 시작하십시오. *WAS\_installdir/bin* 디렉토리로 이동하여 *tperfviewer* 스크립트를 실행하십시오.

Windows 2000 환경에서는 다음과 같이 호스트와 포트를 지정할 수 있습니다.

```
tperfviewer.bat host_name port_number connector_type
```

AIX 및 기타 UNIX® 플랫폼에서는 다음을 사용하십시오.

```
tperfviewer.sh host_name port_number connector_type
```

예를 들면, `tperfviewer.bat localhost 8879 SOAP`

*connector\_type*은 SOAP 또는 RMI 중 하나입니다. SOAP 커넥터에 대한 기본 ND 포트는 8879입니다. RMI 커넥터에 대한 기본 ND 포트는 9809입니다.

b. 시작 --> 프로그램 --> **IBM WebSphere** --> **Application Server v.50** > **Tivoli Performance Viewer**를 누르십시오.

Tivoli Performance Viewer는 사용 중인 WebSphere Application Server 패키지를 검출하고 기본 RMI(Remote Method Invocation) 커넥터 포트를 사용하여 you 연결합니다. 연결에 실패하는 경우, 새 연결 매개변수를 제공하기 위해 대화 상자가 표시됩니다. 원격 호스트 또는 다른 포트 번호로 연결한 다음 명령행을 사용하여 성능 표시기를 시작할 수 있습니다.

3. 데이터 컬렉션 설정을 조정하십시오.

WebSphere Application Server 정보 센터(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)의 "성능 모니터링 레벨 설정" 주제에 있는 지시사항을 참조하십시오.

## WebSphere Commerce PMI 모듈 보고서 값

Tivoli Performance Viewer를 시작한 후, 자원 선택 패널에서 **Commerce** 카운터 그룹을 펼치십시오. 데이터는 상점 ID 아래에 태스크, URL 및 뷰에 따라 분류됩니다. 모든 태스크 명령은 `StoreId=0`에서만 찾을 수 있음을 주의하십시오. 자원 선택 패널에서 특정 URL, 뷰 또는 태스크 명령을 선택하면 카운터 선택 패널이 WebSphere Commerce 카운터를 표시합니다.

다음은 각 태스크, URL 또는 뷰에 포함되는 카운터입니다. 모든 시간 값은 경과 시간입니다.

카운터 이름

정의

평균 응답 시간

태스크의 평균 응답 시간

#### 마지막 응답 시간

태스크의 마지막 응답 시간입니다.

#### 최소 응답 시간

태스크의 최소 응답 시간입니다.

#### 최대 응답 시간

태스크의 최대 응답 시간입니다.

#### 히트 수

태스크가 호출된 총 횟수입니다.

#### 총 응답 시간

태스크의 총 응답 시간입니다.

#### 표준 편차

응답 시간의 표준 편차입니다. 표준 편차 계산 공식에서는 데이터가 표준 분산 규칙을 따른다고 가정합니다.

---

## 기타 성능 분석 도구

사이트 운영자는 다음 도구를 사용할 수도 있습니다.

- IBM Tivoli Web Site Analyzer

IBMTivoli Web Site Analyzer는 웹 사이트 사용량, 상태, 무결성 및 사이트 콘텐츠 캡처, 분석, 저장 및 보고서를 작성하여 방문자 사이트 상호작용과 사이트의 전반적인 성능을 명백히 할 수 있습니다. 이 결과를 이용하여 사이트를 최적화하여 고객 충실도 및 e-business 실효성을 증가시킬 수 있습니다. Web Site Analyzer는 페이지 콘텐츠 및 대상 지정 판매 설정 또는 특정 방문자나 고객 세그먼트에 대한 캠페인에 대한 상품 구매를 추적할 수 있습니다. 이는 적게 방문하는 웹 페이지나 상품 페이지에 대한 투자 감소 또는 웹 탐색의 변화를 나타낼 수 있습니다.

Web Site Analyzer에 대한 추가 정보는 다음 웹 주소를 참조하십시오.

<http://www.ibm.com/software/sysmgmt/products/web-site-analyzer.html>

- Commerce Studio Page Detailer

웹 페이지를 분석하고 동일성, 크기, 소스 및 각 항목을 전달하는 데 걸리는 시간을 페이지에 표시하려면 이 도구를 사용하십시오. 이러한 정보를 사용하여 일반 사용자의 경험을 늘리기 위해 성능을 향상시킬 수 있는 영역을 식별할 수 있습니다. Commerce Studio의 다른 구성요소를 실행하지 않고도 Page Detailer를 실행할 수 있습니다.

---

## 스케줄러 유지보수

적절한 성능을 유지하려면 스케줄러를 적절하게 유지보수해야 합니다. 스케줄러를 일상적으로 유지보수하여 제대로 실행되도록 하는 것은 사이트 운영자의 책임입니다.



SCHSTATUS 테이블에는 SCHCONFIG 테이블에 있는 각 작업의 실행에 해당하는 한 항목이 들어 있습니다. 스케줄러가 실행하는 작업의 볼륨이 크므로 SCHSTATUS 테이블에 보관되는 작업 상태 레코드는 매우 많아집니다. WebSphere Commerce Server의 성능에 역효과를 미칠 수 있습니다. 따라서 스케줄러 상태 표시 페이지를 사용하여 SCHSTATUS 테이블을 정기적으로 정리하도록 권장됩니다. 이 페이지는 관리 콘솔의 구성 메뉴에 있습니다. 작업 상태 레코드를 삭제하면 시간 소인 및 작업 참조 번호에 기초하여 작업을 정리함으로써 SCHSTATUS 테이블의 크기를 줄일 수 있습니다.

또는 CleanJob 작업을 실행하여 SCHSTATUS 테이블의 크기를 자동으로 자르도록 스케줄할 수 있습니다. endTime 매개변수를 사용하여 이전 주 또는 이전 달에 대한 작업 상태 레코드를 제거할 것인지 여부를 지정하십시오. CleanJob 명령의 구문 및 스케줄에 대해서는 해당 플랫폼용 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오.

주: CleanJob을 실행할 때는 상점 공개 상태가 상점 공개 페이지에서 제거됩니다.

작업 히스토리를 보관할 필요가 없는 경우에는 스케줄러의 autoClean 특성을 항상 "ON"으로 설정할 수 있습니다. 그러면 작업이 실행을 완료한 후 SCHSTATUS 테이블에서 작업이 자동으로 제거됩니다.

스케줄러에 대한 추가 정보는 26 페이지의 『스케줄 서비스』 또는 WebSphere Commerce 온라인 도움말을 참조하십시오.

---

## LDAP에 대한 복제

WebSphere Commerce는 LDAP를 사용하여 인증하고 인증 및 프로파일 데이터를 저장할 수 있도록 합니다. 일부 데이터는 WebSphere Commerce 데이터베이스와 LDAP 서버 간에 복제됩니다. 대부분의 복제는 ldapentry.xml 파일을 사용하여 구성할 수 있습니다. 특정 구성원 서브시스템 명령에 의해서만 복제 로직이 트리거되는 이전 버전의 WebSphere Commerce와는 달리 현재 버전의 WebSphere Commerce는 오브젝트 레벨에서 복제를 수행합니다. 이는 필요할 때마다 복제가 수행됨을 의미합니다. 따라서 LDAP 서버가 작동 중지되면 오류가 작성됩니다.

---

## 스테이징 서버에서 사이트 테스트

스테이징 서버에서 사이트를 테스트하려면 다음을 수행하십시오.

1. 스테이징 서버를 구성하십시오.
2. 사용자 정의 테이블에 대한 트리거를 작성하십시오.
3. 원격 데이터베이스를 구성(적용 가능한 경우)하십시오.
4. 스테이징 데이터베이스로 데이터를 복사하십시오.
5. 사이트를 테스트하십시오.

6. 스테이징 확인 명령을 실행하여 고유 색인 키 충돌이 없는지 확인하십시오.
7. 프로덕션 데이터베이스로 데이터를 전달하십시오.
8. 프로덕션 서버로 파일을 복사하십시오.
9. STAGLOG 테이블에서 스테이징된 오브젝트를 삭제하십시오.

## 스테이징 서버 구성

WebSphere Commerce 시스템을 스테이징 서버로 설정할 수 있습니다. 스테이징 서버는 설치 중 또는 설치 후에 구성할 수 있습니다. WebSphere Commerce 설치 중에 스테이징 서버를 설정하는 방법은 *WebSphere Commerce 설치 안내서*에 설명되어 있습니다. 설치 후에 스테이징 서버를 설정하려면 다음을 수행하십시오.

1. 구성 관리자를 사용하여 스테이징 서버로 사용할 별도의 WebSphere Commerce 인스턴스를 작성하고 구성하십시오.
2. 인스턴스를 스테이징 서버로 구성하려면 데이터베이스 패널에서 스테이징 서버 사용 선택란을 선택해야 합니다.
3. 구성 관리자의 캐시 패널에서 캐시가 사용 불가능으로 설정되었는지 확인하십시오.


## 사용자 정의 테이블의 트리거 작성

트리거는 데이터베이스 레코드의 변경사항을 식별하는 STAGLOG 테이블에 항목을 작성합니다. 테이블에 같은 데이터 범위 및 키 특성이 있는 경우, 기존 트리거의 설정을 새 테이블로 수정할 수 있습니다.

새 테이블을 작성하지 않은 경우, 이 단계를 수행할 필요가 없습니다. 새 테이블을 작성한 경우, 사용자 정의 테이블에 맞게 스테이징 서버 구성에 있는 지시사항을 참조하십시오.

## 원격 데이터베이스 구성

프로덕션 서버가 아닌 시스템에 스테이징 서버를 설치한 경우, 원격 데이터베이스를 구성해야 합니다. 스테이징 서버에서 스테이징 유틸리티를 실행하려면 스테이징 서버에 프로덕션 데이터베이스를 원격 데이터베이스로 구성해야 합니다. 프로덕션 서버에서 스테이징 유틸리티를 실행하려면 프로덕션 서버에 스테이징 데이터베이스를 원격 데이터베이스로 구성해야 합니다.

 DB2 데이터베이스의 경우, *DB2 관리 안내서*를 참조하십시오.

 Oracle 데이터베이스의 경우, 제품 문서를 참조하십시오.

## 스테이징 데이터베이스로 데이터 복사

프로덕션 데이터베이스에서 스테이징 데이터베이스로 데이터를 복사하려면 다음을 수행하십시오.

1. PATH 환경 변수를 설정하십시오.



2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-  

```
stagingcopy -scope _all_ -sourcdb production_database_name
             -destdb staging_database_name
```



-  

```
stagingcopy -scope _all_ -sourcdb production_database_name
             -destdb staging_database_name -dbtype oracle -sourcdb_user
             user -sourcdb_passwd password -destdb_user
             user -destdb_passwd password
```

+  
+  
+  
+  
+

-     

```
. stagingcopy.sh -scope _all_ -sourcdb production_database_name
                -destdb staging_database_name -sourcdb_user
                user -destdb_user user
```

-   

```
. stagingcopy.sh -scope _all_ -sourcdb production_database_name
                -destdb staging_database_name dbtype oracle -sourcdb_user
                user -sourcdb_passwd password
                -destdb_user user -destdb_passwd password
```

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.  
예: *myhost:1521:mydb*

5. *stagingcopy\_yyyy.mm.dd\_hh.mm.ss.zzz.log* 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.
6. 스테이징 서버 인스턴스를 중지하고 다시 시작하십시오.

## Stage Check 명령 실행

프로덕션 데이터베이스에 대해 스테이징 데이터베이스 간에 고유 색인 키 충돌 가능성이 있는지를 확인하려면 다음을 수행하십시오.

고유 색인 키의 충돌이 없는지 확인하기 위해 Stage Check 명령을 실행하려면 다음을 수행하십시오.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

-  

```
stagingcheck -scope _unique_index_ -sourcdb staging_database_name
             -destdb production_database_name
```

- ▶ Windows ▶ Oracle

```
stagingcheck -scope _unique_index_ -sourcedb staging_database_name
              -destdb production_database_name -dbtype oracle -sourcedb_user
              user -sourcedb_passwd password -destdb_user
              user -destdb_passwd password
```

+  
+  
+  
+  
+

- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2

```
. stagingcheck.sh -scope _unique_index_
-sourcedb staging_database_name
  -destdb production_database_name -sourcedb_user
  user -destdb_user user
```

- ▶ AIX ▶ Solaris ▶ Oracle

```
. stagingcheck.sh -scope _unique_index_
-sourcedb staging_database_name
  -destdb production_database_name dbtype oracle -sourcedb_user
  user -sourcedb_passwd password -destdb_user
  user -destdb_passwd password
```

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: *myhost:1521:mydb*

5. *stagingcheck\_yyyy.mm.dd\_hh.mm.ss.zzz.log* 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

## 프로덕션 데이터베이스로 데이터 전달

스테이징 데이터베이스에서 프로덕션 데이터베이스로 데이터를 전달하려면 다음을 수행하십시오.

1. PATH 환경 변수를 설정하십시오.
2. 데이터베이스를 구성하십시오.
3. 로그 파일을 기록할 디렉토리로 변경하십시오.
4. 다음을 입력하십시오.

- ▶ Windows ▶ DB2

```
stagingprop -scope _all_ -sourcedb staging_database_name -destdb
            production_database_name
```

- ▶ Windows ▶ Oracle

```
stagingprop -scope _all_ -sourcedb staging_database_name
            -destdb production_database_name -dbtype oracle -sourcedb_user
            user -sourcedb_passwd password -destdb_user
            user -destdb_passwd password
```

+  
+  
+  
+  
+

- ▶ AIX ▶ 400 ▶ DB2 ▶ Solaris ▶ Linux ▶ DB2

```
. stagingprop.sh -scope _all_ -sourcedb staging_database_name
  -destdb production_database_name -sourcedb_user
  user -destdb_user user
```

- ▶ AIX
  - ▶ Solaris
  - ▶ Oracle

```
. stagingprop.sh -scope _all_ -sourcedb staging_database_name
  -destdb production_database_name dbtype oracle -sourcedb_user
  user -sourcedb_passwd password -destdb_user
  user -destdb_passwd password
```

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: *myhost:1521:mydb*

5. *stagingprop\_yyyy.mm.dd\_hh.mm.ss.zzz.log* 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

프로덕션 데이터베이스로 데이터 전달에 대한 자세한 정보는 예를 참조하십시오.

## 프로덕션 서버로 파일 복사

스테이징 서버에서 이미지 또는 HTML 파일(데이터베이스의 데이터가 아닌)을 추가, 변경 또는 삭제할 경우, 이 파일을 프로덕션 서버로 직접 복사해야 합니다.

프로덕션 서버로 파일을 복사하려면 다음을 수행하십시오.

1. 적합한 디렉토리에서 프로덕션 서버로 파일을 복사하십시오.
  - a. 모든 정적 HTML 파일, 관련 이미지 파일 및 기타 삽입된 파일이 들어 있는 하나의 포함된 상자를 작성하십시오. 이 파일에는 새 파일, 갱신된 파일 및 변경되지 않은 파일이 포함되어야 합니다.
  - b. 압축 파일을 프로덕션 서버로 전송하십시오.
  - c. 압축 파일을 프로덕션 서버 디렉토리 구조에서 해당 디렉토리에 압축해제 하십시오. HTML 파일을 이동한 경우, 새 디렉토리의 파일을 가리키도록 WebSphere Commerce 구성을 편집하십시오.
2. 프로덕션 서버에서 사용하지 않은 디렉토리를 삭제하십시오.

## 스테이징된 오브젝트 삭제

스테이징된 오브젝트를 삭제하려면 다음을 수행하십시오.









1. PATH 환경 변수를 설정하십시오.
2. 로그 파일을 기록할 디렉토리로 변경하십시오.
3. 다음을 입력하십시오.

- ▶ Windows
  - ▶ DB2

```
dbclean -object staglog -type obsolete -db dbname
  -days daysold -loglevel loglevel
```
- ▶ Windows
  - ▶ Oracle

```
dbclean -object staglog -type obsolete -db dbname
  -days daysold -loglevel loglevel -dbtype oracle
  -dbuser user -dbpasswd password
```

+  
+  
+

-       
• `. dbclean.sh -object staglog -type obsolete -db dbname  
-dbuser user -days daysold -loglevel loglevel`
-     
• `. dbclean.sh -object staglog -type obsolete -db dbname  
-days daysold -loglevel loglevel -dbtype oracle  
-dbuser user -dbpasswd password`

주: Oracle 데이터베이스 이름으로 *host:port:sid*를 사용하십시오.

예: *myhost:1521:mydb*

4. `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 파일을 검토하여 명령이 성공적으로 실행되었는지 확인하십시오.

스테이징된 오브젝트 삭제에 대한 추가의 예는 오브젝트 삭제 예를 참조하십시오.

---

## WebSphere Commerce Payments 성능 조정 매개변수

WebSphere Commerce Payments에는 WebSphere Commerce Payments 내부 자원 할당을 제어할 수 있는 조정 매개변수가 포함되어 있습니다.

주의: WebSphere Commerce Payments에 익숙한 운영자만 이 매개변수를 수정해야 합니다. 이 매개변수 값을 너무 높게 설정하면 성능 저하 또는 시작 시 WebSphere Commerce Payments의 실패를 초래할 수 있습니다.


다음은 수행할 것을 강력히 권장합니다.

- 작은 증가 단위로만 변경하십시오.
- 한번에 한 매개변수만 변경하십시오. 효과를 관찰한 다음 다른 변경을 수행하십시오.
- 프로덕션 시스템에서 사용하기 전에 중요하지 않은 시스템에서 변경을 철저히 테스트하십시오.

WebSphere Commerce Payments는 다음 스레드 풀 설정을 관리합니다.

### 프로토콜 스레드 풀

이 풀의 스레드는 카세트 내에서 서버측 프로토콜 특정 메시지를 처리하는 태스크가 지정됩니다. 고유한 ComPoint를 정의하는 각 카세트에 대해 그런 풀이 하나씩 있습니다.

 특성 `wpm.poolsize`는 프로토콜 스레드 풀 크기를 변경합니다.

WebSphere Application Server 관리 콘솔에서 다음 단계를 완료하여 풀 크기를 변경할 수 있습니다.

1. **Servers**를 펼치십시오.

2. **Application Servers -> wpm Commerce Payments Server -> 프로세스 정의 -> Java Virtual Machine -> 사용자 정의 특성 -> wpm.poolsize**를 누르십시오.
3. 원하는 값을 입력하십시오. 기본값은 8입니다.
4. 적용을 누르십시오.

일반적으로 서버측 프로토콜 메시지와 관련된 처리의 특성에 따라(네트워크 통신의 증가된 레벨), 이들 스레드는 일반적으로 주어진 각 요청에 대해 오랫동안 유지됩니다. 따라서 프로토콜 메시지를 처리하기 위해 프로토콜 스레드에 의존하는 카세트는 중요한 프로토콜 메시지를 포함하는 전형적인 동시 트랜잭션 수를 수용하기에 충분한 프로토콜 스레드 풀을 사용하여 구성해야 합니다.

### 서비스 스레드 풀

이 풀의 스레드는 현재 또는 미래의 태스크를 배경에서 수행하기 위해 카세트뿐만 아니라 프레임워크에 의해 사용됩니다. 이 풀 내의 기본 스레드 수는 6입니다.

▶ 400 특성 wpm.spoolsize는 서비스 스레드 풀 크기를 변경합니다.

WebSphere Application Server 관리 콘솔에서 다음을 수행하여 풀 크기를 변경할 수 있습니다.

1. **Servers**를 펼치십시오.
2. **Application Servers -> wpm Commerce Payments Server -> 프로세스 정의 -> Java Virtual Machine -> 사용자 정의 특성 -> wpm.spoolsize**를 누르십시오.
3. 값 필드에 원하는 풀 크기를 지정하십시오. 기본값은 6입니다.
4. 적용을 누르십시오.

원래 가장 일반적인 서비스 스레드 사용은 미래의 어떤 시점에서 실행되도록 계획된 태스크를 처리하는 것이었습니다. 이러한 태스크의 예로는 정기적인 유지 보수 태스크뿐 아니라 카세트의 백엔드 프로세서에 대한 재시도 트랜잭션이 있습니다.

### wpm.disableDuplicateOrderCheck

이 매개변수는 WebSphere Commerce Payments가 새 주문을 처리할 때 주문 확인 복제 작성을 생략하고 데이터베이스 액세스를 제거하도록 지시합니다. 이렇게 하면 보다 높은 트랜잭션 처리량을 얻을 수 있습니다. 판매자가 주문 번호 복제 작성을 작성하지 않는 경우에만 이 매개변수를 사용하는 것이 바람직합니다. 그렇지 않으면 주문에 실패하게 됩니다. 이 매개변수를 사용하려면, WebSphere Application Server 관리 콘솔을 통해 다음 단계를 완료하십시오.

1. **Servers**를 펼치십시오.

2. **Application Servers -> wpm Commerce Payments Server -> 프로세스 정의 -> Java Virtual Machine -> 사용자 정의 특성**을 누르십시오.
3. **새로 만들기를** 누르십시오.
4. 이름 필드에 매개변수 이름 `wpm.spoolsize`를 입력하십시오.
5. 값에 **1**을 입력하십시오.
6. **적용을** 누르십시오.

**추가 정보:** 사용자의 데이터베이스 제품이 일정한 시간에 데이터베이스에 설정할 수 있는 동시 연결 수를 제한할 수 있습니다. 이것은 기술적인 한계이거나 데이터베이스 제품을 구매한 사용권 계약에 의해 부과된 한계일 수 있습니다. 위의 매개변수를 조정할 때 이들 요소를 고려해야 합니다. 매개변수들 중 하나라도 너무 높게 설정하여 그런 한계를 초과하는 경우 Payment Servlet 초기화에 실패합니다.

---

## iSeries에서 Payments 인스턴스 데이터베이스 관리

새 주문 및 지불을 WebSphere Commerce Payments로 처리할 때, Payments 인스턴스 데이터베이스 콜렉션이 증가됩니다. iSeries 스토리지 공간이 가득 차기 시작하면 시스템 관리자는 CPI099C - Critical storage lower limit reached와 같은 경고 메시지를 수신할 수 있습니다. 그 경우 WebSphere Commerce Payments 인스턴스 데이터베이스의 공간을 비우십시오.

WebSphere Commerce Payments 인스턴스 데이터베이스 콜렉션은 저널링이 \*YES로 설정되어 작성됩니다. 이것은 인스턴스 라이브러리가 데이터베이스 저널 리시버(데이터베이스 파일에 대한 변경 같은 이벤트가 발생하는 시기를 표시하는 항목)를 포함함을 의미합니다. iSeries에서는 저널 QSQRN이 WebSphere Commerce Payments 인스턴스 라이브러리에 빌드되어 있으며 QSQRNXXXX라는 이름의 저널 리시버가 인스턴스 라이브러리에 작성됩니다. WebSphere Commerce Payments 인스턴스가 너무 커지면 인스턴스 라이브러리의 오래된 저널 리시버(\*JRNRCV 유형의 QSQRNXXXX)를 삭제하여 공간을 비울 수 있습니다. 가장 최근의 저널 리시버는 계속 데이터베이스에 첨부되므로 삭제해서는 안 됩니다. 또한, IBM 서비스에서 사용할 수 있는 Commerce Payments Pruneorders 유틸리티를 사용하여 데이터베이스 테이블을 정리할 수도 있습니다.

스토리지 정리에 대한 추가한 정보는 *AS/400 System Operation* 매뉴얼(SC41-4203)을 참조하십시오.

---

## 제 8 장 문제점 해결

사이트 운영자는 문제점 해결 및 시스템 문제의 솔루션 찾기를 담당합니다. 이 장에서는 Adapter for Crossworlds<sup>®</sup> 및 WebSphere Commerce Payments에 대한 문제점 해결을 다룹니다. 다음에 대한 문제점 해결 정보는 *WebSphere Commerce 설치 안내서*를 참조하십시오.

- 다운로드 가능 도구
- 로그 파일
- WebSphere Application Server 문제점
- 웹 서버 문제점
- WebSphere Commerce 문제점
- 데이터베이스 문제점

---

### Adapter for CrossWorlds<sup>®</sup>

이 절에는 어댑터를 구성할 때 발생할 수 있는 잠재적인 문제점과 이들 문제점을 해결하기 위한 조치가 표시됩니다.

- **문제점:** 한 시스템에 ICS(InterChange Server) 및 저장소가 있고 다른 시스템에 WebSphere Commerce Server가 있습니다. Visigenic ORB 에이전트가 서브네트를 넘어 작업하기 위해 어떤 구성 정보가 필요합니까? 증상은 연결하려 시도할 때 팝업되는 대화 상자로 서버가 실행 중임을 독립적으로 검증할 수 있는 경우에도 WebSphere Commerce Server "Can't locate the InterChange Server '<server name>'. The server is probably not running."를 표시합니다.

#### 해결책 1:

1. Visibroker 설치의 adm 디렉토리(즉, d:\inprise\vbroker\adm)를 가리키는 VBROKER\_ADM이라는 사용자 레벨 환경 변수를 작성하십시오.
2. adm 디렉토리에서 이 ICS에 연결하기 위해 필요한 클라이언트 시스템의 IP 주소와 호스트 이름을 포함하는 agentaddr라는 텍스트 파일을 작성하십시오.
3. osagent 프로세스를 다시 시작하고 VBROKER\_ADM이 이 프로세스를 시작하는 사용자 또는 셸의 환경에 있는지 확인하십시오. 이제 WebSphere Commerce Server를 ICS에 연결할 수 있을 것입니다. 각 osagent가 WebSphere Commerce Server의 원격 IP를 포함하는 고유한 agentaddr 텍스트 파일을 가져야 함을 주의하십시오.

주: agentaddr에는 확장자가 없으며 실행 중이어야 하는 유일한 osagent는 ICS 측에 있습니다.



**해결책 2:** 다른 사람들은 대체 환경 변수인 OSAGENT\_ADDR\_FILE로 더 쉽게 성공했는데, 이 변수는 agentaddr 파일의 경로와 파일 이름을 지정해야 합니다.

ORB가 몇 분의 검색 시간이 필요하기 때문에 임의의 특정 방법을 포기하기 전에 10 분 동안 기다린 후 다시 연결을 시도하십시오.

- **문제점:** WebSphere Commerce Server가 둘 이상의 InterChange Server에 연결할 수 없습니다. OSAGENT\_ADDR 환경 변수가 WebSphere Commerce Server 시스템에 설정되었지만 지정된 시스템에서 실행 중인 InterChange Server에만 연결할 수 있습니다. 두 개의 상이한 서브네트에서 두 InterChange Server를 실행 중이고 세 번째 시스템에서 WebSphere Commerce Server를 실행 중입니다. osagent는 각 InterChange Server 시스템에서 실행 중이고 각 시스템에 VBROKER\_ADM 환경 변수가 각 InterChange Server/osagent 시스템의 IP 주소를 포함하는 agentaddr 파일이 있는 디렉토리를 가리키도록 제대로 구성되었습니다.

**해결책:** 각 osagent 시스템의 VBROKER\_ADM 환경 변수에 지정된 디렉토리에 localaddr 텍스트 파일을 작성하고, 다른 osagent 시스템의 IP 주소 또는 DNS 별명을 지정하십시오.

- **문제점:** NT 시스템의 WebSphere Commerce Server를 Solaris에서 실행 중인 ICS에 연결하려 시도 중입니다.

**해결책:** WebSphere Commerce Server 및 ICS 위치 모두에 대한 agentaddr 파일에 연결될 시스템의 IP 주소뿐 아니라 시스템 자체의 IP 주소도 있어야 합니다. 그러므로 NT 시스템의 agentaddr 파일에는 NT 시스템 자신의 IP 주소와 Solaris 시스템의 IP 주소가 둘다 있어야 합니다.

**주:**

1. 이 절에서 다루지 않는 다른 문제점에 대해서는 IBM 지원 영업대표에 문의하십시오.
2. Adapter for CrossWorlds<sup>®</sup>는 iSeries, Linux, IBM eServer<sup>™</sup> zSeries<sup>®</sup> 또는 S/390<sup>®</sup> Linux에서 지원되지 않습니다.

**주:** 이 절에서 다루지 않는 다른 문제점에 대해서는 IBM 지원 영업대표에 문의하십시오.

---

## WebSphere Commerce Payments

각 비즈니스 모델은 고유한 요구사항을 가지며 따라서 WebSphere Commerce Payments 기능을 다소 다르게 이용합니다. 종종 주어진 비즈니스 환경에 대해 성능과 기능성을 극대화하기 위해 WebSphere Commerce Payments를 조정할 수 있습니다.



## 고성능 환경을 위한 성능 조정

### TCP/IP 소켓에 대한 대기 시간

WebSphere Commerce Payments에 대한 각 요청 시에 TCP 소켓이 TIME\_WAIT 상태가 되고 수 분 동안 해당 상태에 머물게 됩니다. 대량의 요청을 서비스하는 시스템의 경우, 많은 수의 소켓이 TIME\_WAIT 또는 TIME\_CLOSED 상태에 있을 수 있어서 요청 거부, 즉 Cannot connect to WebSphere Commerce Payments 리턴 코드가 발생합니다. 이 행위는 모든 TCP 연결에 대해 예상되며 또한 필요합니다.

TCP 소켓은 소켓에서의 모든 후속 통신이 새로 바운드되는 소켓에 대한 새 통신으로 오해되지 않도록 보장하기 위해 일정한 시간 동안 TIME\_WAIT 상태로 이동합니다. 이 시간은 이론적으로 2 MSL(즉, 최대 세그먼트 수명의 두 배)입니다. 실제로는 TIME\_WAIT 기본값이 Windows 운영체제 및 Solaris에서는 4분이고 AIX에서는 2분입니다. iSeries 시스템에서 기본값은 약 2분입니다. 사용자 운영체제에서 TIME\_WAIT 값을 변경하여 대용량 사용자의 이러한 문제점을 줄일 수 있습니다. 다음은 Windows, AIX 및 Solaris 운영체제에서 간격을 재구성하는 방법의 예입니다.

**예외:** 대체 스택이 사용되고 있는 경우, 다른 방법이 필요할 수 있습니다.

### Windows:

1. 레지스트리에서 다음을 찾으십시오.

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\tcpip\Parameters  
\TcpTimeWaitDelay
```

이 항목이 Windows 레지스트리에 없는 경우, 아래 2단계에 따라서 항목을 작성해야 합니다.

2. 이 항목을 새 DWORD 항목으로 편집하십시오.
3. 항목을 30 - 300(값은 초 수를 표시함) 범위의 임의의 값을 설정하십시오. 이 값을 30으로 설정할 것을 권장합니다.

### AIX:

```
no -o tcp_timewait=1
```

값(1)의 증가 단위는 15초입니다. 예를 들어, 1은 15초이고, 2는 30초에 해당합니다. 값을 1 또는 2로 설정하는 것이 바람직합니다.

### Solaris:

```
ndd -set /dev/tcp tcp_close_wait_interval 30000
```

값(30000)의 단위는 밀리초입니다.



---

## 부록 A. LDAP 시나리오

---

### LDAP 시나리오: LDAP 서버를 구성원 저장소로 사용

LDAP 서버를 구성원 저장소로 사용하는 시나리오는 WebSphere Commerce의 새 인스턴스를 작성하고 이 인스턴스가 LDAP 서버를 구성원 저장소로 사용하도록 지정하는 경우입니다. 이 시나리오에서 WebSphere Commerce 데이터베이스는 Root Organization, Default Organization 및 'wcsadmin' 사용자 항목의 부트스트랩 데이터로 채워집니다.

이 시나리오에서는 다음을 수행해야 합니다.

1. 디렉토리 서버에 필요한 접미부를 작성합니다. WebSphere Commerce가 사용할 사용자 및 조직 엔티티 항목은 이 접미부 아래에 존재합니다.
2. 디렉토리 서버에 '루트 조직', '기본 조직', 'wcsadmin'에 대한 항목을 작성하고 '기본 조직'과 'wcsadmin'을 모두 '루트 조직' 아래 둡니다. 견본 LDIF 파일은 아래의 "LDAP 서버에서 부트스트랩 항목 작성" 관련 링크를 참조하십시오.
3. LDAP 서버에 'wcsadmin' 사용자 항목에 대한 암호를 작성합니다.
4. 디렉토리 서버에 항목이 작성되고 난 후, ORGENTITY 테이블의 DN 열을 루트 조직(MEMBER\_ID -2001) 및 기본 조직(MEMBER\_ID -2000)의 올바른 인식 이름으로 갱신해야 합니다.
5. WebSphere Commerce 속성을 LDAP 속성에 맵핑하도록 ldapentry.xml 파일을 설정하십시오. 'wcsadmin' 사용자를 찾을 수 있도록 ldapentry.xml 파일에 사용자 검색 기준이 지정되었는지 확인하십시오.
6. 'wcsadmin'으로 로그인하고 암호를 제공하십시오.

WebSphere Commerce 또는 다른 응용프로그램을 통해 디렉토리 서버에 추가 사용자를 작성할 때, 사용자는 사용자의 RDN 값이나 DN 값을 사용하여 WebSphere Commerce에 로그인할 수 있습니다. 사용자가 RDN을 사용하여 로그인하는 경우, ldapentry.xml 파일에 검색 기준이 제대로 지정되었는지 확인하십시오.

주: WebSphere Commerce의 여러 인스턴스를 작성하고 이러한 인스턴스가 같은 LDAP 서버를 공유하는 경우, 각 WebSphere Commerce 인스턴스 내의 LDAP 서버 구성이 동일한지 확인해야 합니다.



---

## 부록 B. LDAP 파일

이 부록에서는 다음 LDAP 파일에 대해 자세히 설명합니다.

- ldapmap.dtd
- ldapentry.xml

---

### ldapmap.dtd

ldapentry.dtd는 이전 버전의 WebSphere Commerce에서 사용되는 ldapmap.dtd 파일을 대체합니다. 다음은 LDAP에 사용되는 DTD입니다. 이 DTD를 수정해서는 안됩니다.

```
<!-- new mapping file -->
<!ELEMENT ldapentry (entry+)>
<!ELEMENT entry (ldapsetting,ldapmap)>
<!ELEMENT ldapsetting (ldaprtn,ldapocs,ldapbase)>
<!ELEMENT ldaprtn EMPTY>
<!ELEMENT ldapocs EMPTY>
<!ELEMENT ldapbase EMPTY>
<!ELEMENT ldapmap (map+)>
<!ELEMENT map (objectAttribute+,objectSeparator?,ldapAttribute)>
<!ELEMENT objectAttribute EMPTY>
<!ELEMENT objectSeparator EMPTY>
<!ELEMENT ldapAttribute EMPTY>

<!ATTLIST entry
entryName      (User|Organization|OrganizationalUnit) #REQUIRED>

<!ATTLIST ldapbase
defaultBase    CDATA      #REQUIRED
searchBase    CDATA      #REQUIRED>

<!ATTLIST ldaprtn
rdnName        CDATA      #REQUIRED  keyAttrName    CDATA      #REQUIRED
keyObjName     CDATA      #REQUIRED>

<!ATTLIST ldapocs
objClass       CDATA      #REQUIRED>

<!ATTLIST objectAttribute
attrName       CDATA      #REQUIRED>

<!ATTLIST objectSeparator
attrSeparator  CDATA      #IMPLIED>
<!ATTLIST ldapAttribute
name           CDATA      #REQUIRED
operation      (replace|add) #REQUIRED
flow           (ldapToWcs|wcsToLdap|bothDirections) #REQUIRED>
<!-- End of DTD -->
```

---

## ldapentry.xml

ldapentry.xml 파일은 이전 버전의 WebSphere Commerce에서 사용되는 ldapmap.xml 파일을 대체합니다. 이전에 사용된 ldapmap.xml 파일의 내용이 ldapentry.xml 파일로 전송되었는지 확인하십시오. ldapentry.xml 파일은 가장 공통적으로 사용되는 필드에 대한 기본 매핑을 제공합니다. 기본 스키마 매핑에서 LDAP 속성은 기본 사용자 오브젝트 클래스(person, organizationalPerson, inetOrgPerson 및 ePerson) 및 기본 조직 엔티티 오브젝트 클래스(organization 및 organizationalUnit)에 기초합니다. 이러한 objectclasses는 WebSphere Commerce에서 지원하는 모든 LDAP 서버에 대해 정의됩니다. 기본 스키마 매핑을 이러한 오브젝트 클래스 이상으로 확장하려면 먼저 LDAP 서버에서 스키마 확장을 수행하십시오. 그러면 새 속성에 대한 매핑이 ldapentry.xml 파일에 삽입됩니다. WebSphere Commerce에서 내부적으로 작성된 키, USERS 테이블의 Lastsession과 같이 자주 변경되는 데이터의 속성은 LDAP에 대해 존속시키지 않는 것이 좋습니다.

LDAP 속성에 대한 정보는 다음 사이트를 참조하십시오.

[www.as400.ibm.com/ldap/schema](http://www.as400.ibm.com/ldap/schema)

```
<?xml version="1.0"?>
<!DOCTYPE ldapentry SYSTEM "ldapentry.dtd">
<ldapentry>
  <entry entryName="User">
    <ldapsetting>

      <ldaprdn rdnName="uid"
        keyAttrName="logonId" keyObjName="UserRegistry"/>
      <ldapocs
        objClass="top;person;organizationalPerson;inetOrgPerson"/>
      <ldapbase
        defaultBase="o=Default Organization,o=Root Organization"
        searchBase="o=Root Organization"/>
      </ldapsetting>
      <ldapmap>
        <map>

          <
            objectAttribute attrName="logonPassword"/>

          <ldapAttribute name="userPassword"
            operation="replace" flow="wcsToLdap"/>
          </map>
          <map>

            <objectAttribute attrName="lastName"/>

            <objectAttribute attrName="firstName"/>

            <objectSeparator attrSeparator="/"/>
          </map>
        </map>
      </ldapmap>
    </entry>
  </ldapsetting>
</ldapentry>
```

```

<ldapAttribute name="cn" operation="replace"
  flow="wcsToLdap"/>

</map>

<map>

  <objectAttribute attrName="lastName"/>

  <ldapAttribute name="sn" operation="replace
" flow="bothDirections"/>

  </map>
    <map>

      <objectAttribute attrName="firstName"/>

      <ldapAttribute name="givenName" operation="replace
" flow="bothDirections"/>
        </map>
        <map>

          <objectAttribute attrName="phone1"/>

          <ldapAttribute name="homePhone" operation="replace
" flow="bothDirections"/>
            </map>
            <map>

              <objectAttribute attrName="zipCode"/>

              <ldapAttribute name="postalCode" operation="replace
" flow="bothDirections"/>
                </map>
                <map>

                  <objectAttribute attrName="address1"/>

                  <objectAttribute attrName="address2"/>

                  <objectAttribute attrName="address3"/>

                  <objectSeparator attrSeparator="/"/>

                  <ldapAttribute name="postalAddress" operation="replace
" flow="bothDirections"/>
                    </map>
                    </ldapmap>
                    </entry>

                entry entryName="Organization">
                  <ldapsetting>

```

```

<ldaprdn rdnName="o" keyAttrName="orgEntityName"
keyObjName="Organization"/>
<
ldapocs objClass="top;organization"/>
<
ldapbase defaultBase="o=Root Organization" searchBase="
o=Root Organization"/>
</ldapsetting>
<ldapmap>
<map>

<objectAttribute attrName="businessCategory"/>

<ldapAttribute name="businessCategory" operation="
replace" flow="bothDirections"/>
</map>
<map>

<objectAttribute attrName="description"/>

<ldapAttribute name="description" operation="
replace" flow="bothDirections"/>
</map>
<map>

<objectAttribute attrName="address1"/>

<objectAttribute attrName="address2"/>

<objectAttribute attrName="address3"/>

<objectSeparator attrSeparator=""/>

<ldapAttribute name="postalAddress" operation="replace
" flow="bothDirections"/>
</map>
<map>

<objectAttribute attrName="phone1"/>

<ldapAttribute name="telephoneNumber" operation="replace
" flow="bothDirections"/>
</map>
</ldapmap>
</entry>
<entry entryName=
"OrganizationalUnit">
<ldapsetting>

<ldaprdn rdnName="ou" keyAttrName="orgEntityName"
keyObjName="Organization"/>

<ldapocs objClass="top;organizationalUnit"/>

<ldapbase defaultBase="o=Root Organization"
searchBase="o=Root Organization"/>
</ldapsetting>

```



```

<ldapmap>
  <map>

<objectAttribute attrName="businessCategory"/>

<ldapAttribute name="businessCategory" operation="replace
" flow="bothDirections"/>
  </map>
  <map>

<objectAttribute attrName="description"/>

<ldapAttribute name="description" operation="replace
e" flow="bothDirections"/>
  </map>
  <map>

<objectAttribute attrName="address1"/>

<objectAttribute attrName="address2"/>

<objectAttribute attrName="address3"/>

<objectSeparator attrSeparator="/" />

<ldapAttribute name="postalAddress" operation="replace
" flow="bothDirections"/>
  </map>
  <map>

<objectAttribute attrName="phone1"/>

<ldapAttribute name="telephoneNumber" operation="replace
" flow="bothDirections"/>
  </map>
</ldapmap>
</entry>
</ldapentry>

```

ldapentry.xml 파일의 형식은 다음과 같습니다.

**entry** 항목 요소는 맵핑이 어느 구성원 유형에 대한 것인지를 식별합니다. 올바른 값은 User, Organization 또는 OrganizationalUnit입니다.

#### **ldapsetting - ldaprdn - rdnName**

어느 LDAP 속성이 RDN 속성인지를 지정합니다.

#### **ldapsetting - ldaprdn - keyAttrName**

어느 WebSphere Commerce 속성이 RDN 속성에 맵핑되는지를 지정합니다. 사용자인 경우, 속성 이름은 UserRegistrationAdd 명령의 구문에 문서화되어 있는 특성 이름과 상응합니다. 속성 이름은 대소문자가 구분됩니다. OrgEntityAdd 명령이 사용되어야 하는 조직 및 조직 단위의 경우와 유사합니다.

**ldapsetting - ldaprdn - keyObjName**

keyAttrName에 지정된 WebSphere Commerce 속성이 발견될 수 있는 WebSphere Commerce 액세스 bean을 지정합니다.

**ldapsetting - ldapocs - objClass**

LDAP 항목을 작성하는 데 사용되는 LDAP 오브젝트 클래스를 지정합니다.

**ldapsetting - ldapbase - defaultBase**

RDN을 로그온에 사용하는 경우 WebSphere Commerce 가 로그온 중에 사용자를 검색하는 첫 번째 검색 기준 DN을 지정합니다. 조직 엔티티의 경우, defaultBase는 WebSphere Commerce 가 필요할 때마다 조직 엔티티를 검색하는 첫 번째 검색 기준 DN을 지정합니다.

**ldapsetting - ldapbase - searchBase**

WebSphere Commerce 사용자 및 조직 엔티티를 검색하는 나머지 기본 DN을 지정합니다.

**map -objectAttribute - attrName**

WebSphere Commerce 속성 이름 사용자인 경우, 속성 이름은 UserRegistrationAdd 명령의 구문에 문서화되어 있는 특성 이름에 해당합니다. 속성 이름은 대소문자가 구분됩니다. OrgEntityAdd 명령이 사용되어야 하는 조직 및 조직 단위의 경우와 유사합니다.

**map -ldapAttribute - name**

attrName에 지정된 WebSphere Commerce 속성에 맵핑될 LDAP 속성 이름

**map -ldapAttribute - flow**

속성값을 LDAP에서 읽어오는지, LDAP에 작성되는지 또는 그 둘다인지를 지정합니다. 올바른 값은 ldapToWcs, wcsToLdap 또는 bothDirections입니다.

**map -ldapAttribute - operation**

LDAP의 속성값을 어떻게 수정할 것인지 여부를 지정합니다. 올바른 값은 replace 또는 add입니다. replace 값은 현재 사용자 정보는 제공된 새 정보로 갱신합니다(예: 기존 전화 번호를 새 번호로 바꾸기). add 값에는 사용자에게 대한 다른 항목이 포함됩니다(예를 들어, 현재 사용자 정보에 새 전화 번호 추가, 사용자의 전화 번호 목록 작성).

**map - objectSeparator - attrSeparator**

WebSphere Commerce 속성을 LDAP 속성에 저장하거나 LDAP 속성에서 검색할 때 사용하는 분리 문자

---

## 부록 C. WebSphere Commerce Payments 학습서

이 학습서는 OfflineCard Cassette를 사용한 최초 WebSphere Commerce Payments 설치를 단계별로 안내합니다. 이 초기 설치의 일부로서 그리고 가장 공통적인 관리 및 지불 기능을 보여주기 위해 WebSphere Commerce Payments는 OfflineCard Cassette 및 Sample Checkout을 사용한 학습 지원을 제공합니다. 관리, 구성 및 지불 기능에 대한 자세한 정보는 WebSphere Commerce Payments 사용자 인터페이스에 대한 온라인 도움말을 참조하십시오.

다음은 OfflineCard Cassette와 함께 동작하는 WebSphere Commerce Payments를 설정하기 위한 여섯 가지 꼭 수행해야 하는 구성 태스크입니다.

1. WebSphere Commerce Payments 사용자 인터페이스 액세스
2. WebSphere Commerce Payments 판매자 작성 및 지불 카세트 권한 부여
3. WebSphere Commerce Payments 사용자 정의
4. 사용자 역할 지정
5. 계정 작성
6. 브랜드 작성

구성 태스크가 완료되면 주문을 하고 판매자가 일반적으로 매일 매일 수행하는 다음 지불 처리 태스크를 시작할 준비가 되었습니다.

7. 주문 작성
8. 주문 승인
9. 지불 예치
10. 일괄처리 계산
11. 대변 발행
12. 매일의 결산 일괄처리 총계 보기

---

### 1단계: WebSphere Commerce Payments 사용자 인터페이스 액세스

첫 번째 단계는 지불 관리자로 Payments에 로그인하는 것입니다.

WebSphere Commerce Payments 사용자 인터페이스에 로그인하려면 다음을 수행하십시오.

1. 웹 브라우저에서 `http://host_name:port/webapp/PaymentManager`를 지정하십시오. 여기서 `host_name`은 Payments용 웹 서버를 실행하는 호스트 이름이고 `port`는 Payments가 실행 중인 포트 번호를 의미합니다.
2. WebSphere Commerce 사이트 운영자 ID를 입력하십시오.

3. WebSphere Commerce 사이트 운영자 암호를 입력하십시오.
4. 로그인 버튼을 누르십시오.

**중요:** WebSphere Commerce Payments 인스턴스가 사용 중인 HTTP 서버가 기본값이 아닌 포트 번호에 대해 구성되는 경우, 이 학습 전반에 걸쳐서 WebSphere Commerce Payments 웹 주소 링크의 호스트 이름 뒤에 포트 번호를 포함하십시오.

사용자 인터페이스의 오른쪽 상단에 있는 아이콘은 다음과 같이 사용됩니다.

- 페이지를 최신 정보로 고치려면 다방향 화살표를 누르십시오.
- 방문한 마지막 페이지로 돌아가려면 왼쪽 화살표를 누르십시오.
- 페이지에 대한 컨텍스트별 온라인 도움말에 액세스하려면 물음표를 누르십시오.

## 2단계: WebSphere Commerce Payments 판매자 작성 및 카세트 권한 부여

아직 수행하지 않았다면 지불 관리자로서 WebSphere Commerce Payments에 로그인하십시오. 이제 전체적인 열람 및 액세스 권한이 생깁니다. WebSphere Commerce Payments 구성의 첫 번째 단계는 판매자를 작성하고 해당 판매자에게 지불 카세트를 사용할 권한을 부여하는 것입니다. 판매자를 작성하고 카세트를 권한 부여하려면 다음을 수행하십시오.

1. 탐색 프레임에서 판매자 설정을 누르십시오.
2. 판매자 설정 페이지에서 판매자 추가를 누르십시오.
3. 판매자 설정 페이지에서 다음 정보를 입력하십시오(모노체로 된 텍스트는 학습 과정에서 반드시 입력해야 합니다).

표 11. 판매자 작성 필드

필드 이름	설명
판매자 이름	Test Store를 입력하십시오. 이것이 판매자에게 지정하는 이름입니다. 사용자 인터페이스에 표시하는 역할만 합니다.
판매자 번호	123456789를 입력하십시오. 이것은 사용자가 지정하여 모든 트랜잭션 데이터에서 판매자를 고유하게 식별하는 번호입니다.
권한부여된 카세트	<b>OfflineCard</b> 에 대한 상자를 체크하십시오. 이 상자를 체크하면 판매자에게 이 지불 카세트를 사용할 권한을 부여합니다.

4. 판매자 작성을 눌러서 판매자 구성을 저장하십시오.

이 카세트에 대한 사용 권한을 부여하려는 판매자를 이미 작성한 경우 다음을 수행하십시오.

1. 판매자 설정을 누르십시오.
2. 판매자 이름을 누르십시오.
3. **OfflineCard**에 대한 상자를 선택하십시오.

4. 판매자 작성을 누르십시오.

이제 판매자가 이 카세트를 사용할 권한을 가집니다.

---

### 3단계: WebSphere Commerce Payments 사용자 정의

이 학습에서는 다음 사용자에게 대해 작업합니다.

- 설치 중에 사용되는 기본 사용자(추가 정보에 대해서는 *WebSphere Commerce 설치 안내서*를 참조하십시오.)
- 정의하려는 사용자 - *Pat*

WebSphere Commerce 조직 관리 콘솔을 사용하여 사용자 정의 및 관리와 같은 태스크를 수행하게 됩니다. WebSphere Commerce Payments에서 사용자 정의는 2 파트로 된 프로세스입니다. 예를 들어, 사용자 *Pat*을 정의하려면 WebSphere Commerce 조직 관리 콘솔을 사용하여 *Pat*에게 사이트 운영자 역할을 지정해야 합니다. 그런 다음 Payments UI에서 직접 또는 조직 관리 콘솔을 통해 *Pat*의 사용자 역할을 판매자 관리자로 지정할 수 있습니다. 판매자를 작성해야 사용자에게 액세스를 지정할 수 있음을 주의하십시오.

Payments 사용자를 구성하려면 다음을 수행하십시오.

1. 웹 브라우저에서 `https://host_name:port/adminconsole`을 지정하십시오.
2. 액세스 관리 > 사용자를 누르십시오.
3. 새로 만들기를 누르십시오.
4. 새 사용자 마법사를 사용하여 새 사용자인 *Pat*을 작성하십시오.
5. 액세스 페이지에서 *Pat*에게 사이트 운영자의 역할을 지정하십시오.
6. **Payments** > 사용자를 누르십시오.
7. 사용자 이름 *Pat*을 입력하십시오.
8. 검색을 누르십시오.
9. *Pat*의 이름을 누르십시오. 사용자의 구성 페이지가 표시됩니다.
10. 원하는 판매자 이름을 선택하십시오.
11. 원하는 역할에 대한 라디오 버튼(이 경우에는 판매자 관리자)을 선택하십시오.
12. 갱신을 눌러서 스펙을 저장하십시오.

WebSphere Commerce 제품에서 원격으로 시스템에 WebSphere Commerce Payments를 설치하는 중이고 WebSphere Commerce Payments 사용자 인터페이스에서 WebSphere Commerce 스타일시트를 사용하려는 경우, `Payments_installdir\samples\wcs\PMCustomUI.properties` 디렉토리의 `PMCustomUI.properties` 파일을 WebSphere Commerce Payments 설치 디렉토리에 복사해야 합니다.

## 4단계: 사용자 역할 지정

사용자는 다음의 WebSphere Commerce Payments 역할 중 하나에 지정되어야 합니다. 이 역할은 해당 WebSphere Commerce 역할에 대한 상대적인 맵핑을 갖습니다.

표 12. 역할 맵핑

Payments 역할	WebSphere Commerce 역할
지불 관리자	사이트 운영자
판매자 관리자	사이트 운영자
감독자	작업 또는 판매 관리자
직원	고객 서비스 대표

다음 사용자를 작성한 후,

- 사용자 *Pat*
- 판매자 *Test Store*

WebSphere Commerce Payments 구성에서 *Pat* 역할을 지정할 준비가 되었습니다.

**예외:** 또한 *WebSphere Commerce Payments 액세스 없음*의 역할을 지정하여 WebSphere Commerce Payments에 대한 사용자 액세스를 거부할 수 있습니다. 이것은 임시로 사용자 액세스를 거부해야 하는 경우(예를 들어 휴가 중인 사용자) 유용합니다. WebSphere Commerce Payments 역할 맵핑에 대한 추가 정보는 *WebSphere Commerce Payments 프로그래밍 안내서* 및 참조서의 역할 맵핑 테이블을 참조하십시오.

*Pat*에게 *Test Store*에 대한 판매자 관리자 역할을 지정하려면 Payments UI에서 다음을 수행하십시오.

1. 탐색 프레임에서 사용자를 누르십시오.
2. 사용자 검색 페이지에 사용자 이름 *Pat*을 입력하고 검색을 누르십시오.
3. 사용자 페이지에서 사용자 이름 **Pat**을 누르십시오.
4. 판매자 화면이동 상자에서 **Test Store**를 누르십시오.
5. 판매자 관리자에 대한 라디오 버튼을 선택하십시오.
6. 갱신을 눌러서 사용자 구성을 저장하십시오.

이 시점에서 WebSphere Commerce Payments 사용자 인터페이스에서 로그인 후 이번에는 판매자 관리자, *Pat*으로 다시 로그인하십시오.

**중요:** Commerce Service Provider(CSP)가 여러 판매자에게 원격으로 서비스하는 WebSphere Commerce Payments를 설정하도록 호스팅된 환경에서 CSP는 지불 관리자이고 각 판매자의 제공업체와의 장기 구매 계약당으로 작용합니다. 이 시나리오에서 판매자는 CSP(지불 관리자)가 부여하는 판매자 관리자 권한으로 자신의 고유한 판매자 설정을 구성합니다.

## 판매자 관리자로서 로그인

로그오프하고 다시 로그인하려면 다음을 수행하십시오.

1. 탐색 프레임에서 관리자 로그오프를 눌러서 기본 WebSphere Commerce Payments 로그인 페이지로 리턴하십시오.
2. 사용자 ID Pat을 입력하십시오.
3. Pat에 대해 정의된 암호를 입력하십시오.
4. 확인을 누르십시오.

이 학습의 나머지 부분에서 여러분의 역할은 Test Store에 대한 판매자 관리자 권한을 갖는 사용자 Pat이 됩니다. WebSphere Commerce Payments 사용자 인터페이스의 뷰가 판매자 관리 기능으로 제한됩니다. 이전에는 지불 관리자로서는 판매자 및 지불 관리자 기능에 대한 글로벌 뷰를 가지고 있었습니다.

---

## 5단계: 계정 작성

지금까지는 하나의 판매자인 Test Store를 정의하고 하나의 지불 카세트인 OfflineCard Cassette를 사용했습니다. 판매자 관리자로서의 첫 번째 태스크는 OfflineCard Cassette에 대한 계정을 설정하는 것입니다.

계정은 판매자와 해당 판매자에 대한 트랜잭션을 처리하는 금융 기관 간의 관계입니다. 각 지불 카세트에 대해 복수 계정이 있을 수 있습니다. 이 학습에서는 OfflineCard Cassette에 대해 하나의 계정을 작성합니다.

계정을 작성하려면 다음을 수행하십시오.

1. 탐색 프레임에서 판매자 설정을 누르십시오.
2. 판매자 설정 페이지에서 Test Store에 대한 OfflineCard Cassette 아이콘을 누르십시오.
3. OfflineCard Cassette 페이지에서 계정을 누르십시오.
4. 계정 페이지에서 계정 추가를 누르십시오.
5. 다음 필드를 완료하십시오(모노체로 된 텍스트는 학습 과정에서 반드시 입력해야 합니다).

표 13. 계정 추가 필드

필드 이름	설명
계정 이름	Test Account를 입력하십시오. 이것은 사용자가 계정에 지정하는 이름입니다. 이는 사용자 인터페이스에 표시하는 역할만 합니다.
계정 번호	111111111을 입력하십시오. 이것은 호스팅 서비스 제공업체나 판매자 관리자가 모든 트랜잭션 데이터에서 계정을 고유하게 식별하기 위해 지정하는 번호입니다.



표 13. 계정 추가 필드 (계속)

필드 이름	설명
금융 기관 이름	Test Bank를 입력하십시오. 이것은 이 계정을 소유하는 금융 기관의 이름입니다. 이는 사용자 인터페이스에 정보를 표시하는 역할만 합니다.
통화	이 계정이 수집할 통화를 선택하십시오.
일괄처리 종료 시간	카세트가 이 계정에 대한 일괄처리를 자동으로 닫는 자정 이후의 시간(분). 값 0(영)은 자정을 의미합니다. 1439가 허용되는 최대값입니다. 널(Null)값은 자동 일괄처리 종료를 사용 불가능하게 합니다.

6. 계정 작성을 눌러서 OfflineCard Cassette에 대한 계정을 작성하십시오.

## 계정 설정 지정

학습을 사용하기 위해 카세트에 대해 방금 작성한 계정 및 브랜드에 대해 임의의 특정 기본 또는 고급 설정을 지정할 필요는 없습니다. 그러나 계정에 대해 특정 옵션을 지정할 수 있음을 인식해야 합니다. 예를 들어, 판매자 관리자는 계정이 지불 승인과 연관된 만기 기간을 갖는지 여부를 정의할 수 있습니다.

OfflineCard Cassette에 대한 WebSphere Commerce Payments 사용자 인터페이스에서 계정 설정을 보려면 다음을 수행하십시오.

1. 탐색 프레임에서 판매자 설정을 선택하십시오.
2. 판매자 설정 창에서 Test Store 창의 OfflineCard Cassette 아이콘을 누르십시오.
3. OfflineCard Cassette 창에서 계정을 누르십시오.
4. 계정 창에서 **Test Account**를 선택하십시오.
5. Test Account 창에서 계정 설정을 선택하십시오.
6. 해당 선택을 눌러서 기본 또는 고급 설정을 보거나 변경할 수 있습니다. 예를 들어 자동 지불 처리 옵션에 대한 승인 만기 설정을 보려면 고급 설정을 선택하십시오.
7. 해당 옵션을 입력하고 갱신을 선택하여 계정 설정을 갱신하십시오(취소하려면 계정 설정 창의 왼쪽 화살표를 선택하십시오).

처리 옵션에 대한 설정의 추가 정보에 대해 온라인 도움말을 참조할 수 있습니다. 그러나 지금은 그 중 하나인 승인 만기 옵션에 대해 설명합니다.

승인 만기를 설정하여 지불 승인이 유효한 기간을 제어할 수 있습니다. 승인이 만기된 후 자금 예치를 위해 금융 기관이 부과하는 비용을 피하기 위해 승인 만기 기간을 설정하는 것이 유용할 수 있습니다. 예를 들어, 신용 카드의 경우, 지불 승인은 신용 카드 발행자가 설정한 특정 일 수 동안만 지속됩니다. 승인이 만기된 후 주문 지불을 위해 자금을 예치하려 시도하는 경우, 금융 기관이 자금 예치를 거부하거나 예치하기 위한 추가 비용을 부과할 수 있습니다. WebSphere Commerce Payments에서 승인 만기 기



간을 설정하여 WebSphere Commerce Payments가 승인 권한부여가 만기한 자금의 캡처를 막기 때문에 이 상황을 피할 수 있습니다.

지불 승인이 만기된 후에 자금을 예치하려는 경우에는 지불이 승인 상태에 있지 않음을 나타내는 메시지를 수신하게 됩니다. WebSphere Commerce Payments 사용자 인터페이스에서 금액이 0인 지불을 취소할 수 있습니다. 지불이 취소된 후에는 승인을 수행하여 지불을 다시 승인 상태에 둘 수 있습니다.

승인 만기 옵션을 사용하려면 사용 중인 카세트가 승인 만기를 지원해야 합니다. WebSphere Commerce Payments에서 제공되는 OfflineCard, VisaNet, Paymentech 및 CustomOffline 카세트는 승인 만기를 지원합니다.

**중요:** 승인 만기 기간으로 입력하는 값이 계정과 연관된 모든 브랜드에 적용됩니다. 각 브랜드가 상이한 지불 승인 만기 규칙을 갖는 경우 가장 낮은 공통 분모에 대해 승인 만기 값을 설정하거나 각 브랜드에 대해 별도의 계정을 작성하십시오. 각 브랜드에 대해 별도의 계정을 작성하는 경우 각 계정에 대해 개별적으로 일괄처리 지불을 결산해야 합니다.

주문 승인, 지불 예치 및 결산 일괄처리에 대한 추가 정보가 이 장의 뒤에서 제공됩니다.

---

## 6단계: 브랜드 작성

OfflineCard Cassette를 사용하려면 먼저 상점에서 이용할 수 있는 신용 카드 회사의 브랜드 이름을 구성해야 합니다. 이것은 구매자가 주문 과정에서 적당한 신용 카드를 목록에서 선택할 수 있습니다. OfflineCard Cassette에 대한 브랜드를 작성하려면 다음을 수행하십시오(이 학습의 경우에는 ROBO가 브랜드로 사용됩니다).

1. 탐색 프레임에서 판매자 설정을 누르십시오.
2. 판매자 설정 페이지에서 Test Store에 대한 OfflineCard Cassette 아이콘을 누르십시오.
3. OfflineCard Cassette 페이지에서 계정을 누르십시오.
4. 계정 페이지에서 계정 이름인 Offline Account를 누르십시오.
5. Offline Account 페이지에서 브랜드를 누르십시오.
6. 브랜드 페이지에서 브랜드 추가를 누르십시오.
7. 브랜드 이름 필드에 ROBO를 입력하십시오.
8. 브랜드 작성을 누르십시오.

## 7단계: Sample Checkout을 사용하여 주문 작성

판매자 관리자는 글로벌 판매자 권한을 가지며, 이는 다음을 수행할 수 있음을 의미합니다.

- 판매자별 관리 기능
- 모든 지불 처리 기능

실제 비즈니스 시나리오에서는 제한된 지불 처리 권한을 가지는 판매자가 정의한 다른 사용자(예: 감독자 및 점원)에게 지불 처리 태스크를 위임할 수 있습니다. 이 학습에서는 판매자 관리자인 여러분들이 이러한 태스크를 수행합니다. 지불 처리를 시작하는 데 필요한 모든 WebSphere Commerce Payments 및 판매자 관리 태스크를 완료했으므로 이제 다음을 시작할 수 있습니다.

- 주문 승인
- 지불 예치
- 결산 일괄처리
- 대변 발행
- 매일의 결산 일괄처리 총계 보기

이 학습의 경우에는 Sample Checkout 도구를 사용하여 지불 처리를 위한 세 주문을 작성합니다. Sample Checkout 도구는 카세트 구현을 테스트하기 위해 견본 주문을 작성할 수 있는 사용자 인터페이스를 제공합니다. Sample Checkout에 액세스하려면 먼저 아래에 설명한 것처럼 기본 사용자 ID 및 암호를 변경해야 합니다.

WebSphere Commerce Payments Sample Checkout에 액세스하여 주문을 작성하려면 다음을 수행하십시오.

1. 다음 디렉토리에서

```
WAS_installdir/installedApps/host_name/  
payments_instance_Commerce_Payments_App.ear/SampleCheckout.war
```

구성 파일 SampleCheckout.xml을 여십시오.

2. SampleCheckout 요소에서 다음 속성값을 변경하십시오.

```
pmHostName="fully_qualified_host_name "  
pmPort="port "  
default_userid="wc_userid "  
password="wc_password "
```

3. 파일을 저장하십시오. 브라우저에서 `http://host_name:port/webapp/SampleCheckout` 을 지정하십시오. 여기서 `host_name`은 Payments용 웹 서버를 실행 중인 시스템의 호스트 이름이고, `port`는 Payments가 실행 중인 포트를 의미합니다.
4. Sample Checkout 페이지에서 다음 정보를 입력하십시오(모노체로 된 텍스트는 학습 과정에서 반드시 입력해야 합니다).

표 14. Sample Checkout 필드

이름	설명
판매자 번호	OfflineCard Sample Checkout 판매자 번호인 987654321을 입력하십시오.
주문 번호	주문 번호를 의미하는 임의의 번호를 입력하십시오.
총계	주문의 총액을 표시하는 임의의 금액을 입력하십시오.
통화	미국 달러를 입력하십시오. 이 주문에 사용되는 통화입니다.
지불 방법	지불 유형으로 <b>OfflineCard</b> 를 선택하십시오.
브랜드	사용하려는 OfflineCard의 브랜드를 선택하십시오.
신용 카드 번호	4111111111111111을 입력하십시오.
만기 날짜	신용 카드에 대한 만기 월 및 연도를 선택하십시오.

▶ 400

iSeries 시스템의 경우, 사용 중인 신용 카드의 브랜드를 지정해야 합니다.

#### 5. 구매를 누르십시오.

이들 단계를 두 번 더 반복하여 지불을 처리할 세 개의 주문을 작성하십시오.

## 8단계: 주문 승인

Sample Checkout을 사용하여 세 개의 주문을 작성했다면 해당 주문을 승인할 수 있습니다. 주문을 승인하려면 다음 단계를 따르십시오.

1. 브라우저에서 `http://host_name:port/webapp/PaymentManager`를 지정하고(여기서 `host_name`은 Payments용 웹 서버를 실행 중인 시스템의 호스트 이름이고, `port`는 Payments가 실행 중인 포트 번호임) Pat으로 로그인하십시오.
2. 탐색 프레임에서 승인을 누르십시오.
3. 승인 페이지에서 사용자의 주문 중 하나에 대한 상자를 선택하십시오.
4. 선택 승인을 누르십시오. 승인 결과 페이지가 승인 요청의 상태를 표시합니다.
5. 승인이 완료될 때 승인 화면으로 리턴을 누르십시오.

두 개의 주문이 여전히 사용자의 승인을 기다리고 있습니다. 모두 승인을 눌러서 총액에 대한 승인 페이지에서 동시에 승인할 수도 있습니다. 그러나 승인 기능을 보다 잘 보여주기 위해 이 학습은 각 주문을 개별적으로 작업하는 방법에 대해 설명합니다.

### 주문 페이지에서 주문 승인

이 절에서는 주문 페이지에서 주문의 총 주문액의 일부만을 승인합니다. 주문과 연관된 상품의 일부가 주문 처리 시에 운송이 불가능한 경우, 예를 들어 상품이 이월 주문되는 경우 주문의 일부만 승인하는 것이 유용할 수 있습니다.

1. 승인 페이지에서 승인을 기다리는 나머지 주문 중 하나에 대한 주문 번호를 누르십시오.
2. 주문 번호 페이지에서 주문 정보를 볼 수 있습니다. 승인을 눌러서 이 주문을 승인하십시오.
3. 주문 번호 승인 페이지에 다음 필드가 표시됩니다.

표 15. 주문 번호 승인 필드

필드 이름	설명
통화	이 주문에 사용되는 통화의 유형입니다. 이것은 읽기 전용 필드입니다.
주문 총계	주문에 사용되는 통화로 표현되는 주문의 총액입니다. 이것은 읽기 전용 필드입니다.
승인 총계	이 읽기 전용 필드는 주문의 총계가 아직 승인되지 않았으므로 0을 표시합니다.
예치금 총계	이 읽기 전용 필드는 총계가 아직 승인되거나 예치되지 않았으므로 0을 표시합니다.
승인 총계	이것은 주문의 총액입니다.
승인 코드	수동 오프라인 승인 요청 프로세스에서 리턴되는 승인 코드입니다. 지불 상태가 승인됨으로 변경됩니다.
거부 이유	수동 오프라인 승인 요청 프로세스에서 리턴되는 거부 이유입니다. 지불 상태가 거부로 변경됩니다.
AVS 결과 코드	주소 확인 시스템(AVS: Address Verification System) 결과 코드입니다.

승인 총계를 3.00으로 변경하십시오. 선택적으로 총계를 승인하는 승인 이유 또는 승인이 거부되었음을 표시하는 거부 이유를 지정하십시오.

4. 승인을 눌러서 3달러에 대해 이 주문을 승인하십시오. 승인 처리가 완료되면 주문 페이지가 최신 정보로 고쳐지고 승인 상태를 표시합니다.

## 판매 기능으로 주문 승인

앞에서 다른 주문의 일부만을 승인했기 때문에 여전히 승인 페이지에는 두 개의 주문 항목이 있습니다. 이 단계에서는 판매 기능을 사용하여 나머지 주문을 승인합니다.

판매 기능을 사용하면 주문을 승인하고 승인됨 상태를 생략하여 직접 예치된 상태로 이동시킬 수 있습니다. 판매 기능은 주문 지불을 자동으로 승인하고 예치합니다. 따라서 판매를 자동 예치를 통한 승인으로 생각할 수 있습니다. 예를 들어 다운로드 가능한 소프트웨어나 전자 정보를 판매하고 있을 때 구매자에게 물품을 신속히 운송하고 자금의 캡처를 확실히 하려면 판매 기능을 사용하십시오. 그러나 판매 기능을 사용하면 승인에 대한 승인 또는 거부 이유를 설정할 수 없습니다. 판매가 승인과 예치를 하나의 트랜잭션으로 병합하기 때문에 트랜잭션별로 비용이 부과될 때도 유용합니다.

판매 기능으로 주문을 승인하려면 다음을 수행하십시오.

1. 탐색 프레임에서 승인을 누르십시오.

- 승인 페이지에서 모두 판매를 누르십시오. 처리가 완료될 때 판매를 위해 제출된 각 주문에 대한 승인 상태가 표시됩니다.
- 판매가 완료될 때 승인 화면으로 리턴을 누르십시오.

## 9단계: 지불 예치

예치금 기능을 사용하면 주문 지불을 예치할 수 있습니다. 하나의 주문 번호에 연관된 복수 지불을 가질 수 있습니다. 동일한 목록에 서로 다른 지불 정보를 가지는 동일한 주문 번호가 여러 번 표시될 수 있습니다.

지불을 예치하려면 다음을 수행하십시오.

- 탐색 프레임에서 예치를 누르십시오.
- 표시된 지불 중 하나에 대한 상자를 선택하고 선택된 주문 지불 예치를 누르십시오. 처리가 완료되면 예치를 위해 제출한 지불의 예치 상태가 표시됩니다.
- 예치가 완료되면 예치 화면으로 리턴을 누르십시오.

또한 지불의 일부를 예치할 수 있습니다. 일부 지불을 예치하려면 다음을 수행하십시오.

- 예치 페이지에서 예치를 기다리는 지불 중 하나의 지불 번호를 누르십시오.
- 지불 페이지에서 예치를 누르십시오.
- 주문 지불 페이지에서 예치 금액을 2.00으로 변경하고 예치를 누르십시오.

## 10단계: 결산 일괄처리

일괄처리는 금융 기관에서 하나의 단위로 처리하는 지불 및 대변의 콜렉션입니다. 일괄 처리는 하나의 판매자 및 하나의 계정과 연관됩니다. 이전 단계에서 예치한 지불이 일괄처리에 표시됩니다. 금융 기관에 의한 처리를 시작하려면 이 일괄처리를 결산해야 합니다. 결산이 완료된 이후에는 금융 기관이 자금 이체를 책임집니다.

일괄처리를 결산하려면 다음을 수행하십시오.

- 탐색 프레임에서 일괄처리 검색을 누르십시오. 또는 결산을 눌러도 됩니다.
- 일괄처리 검색 페이지에서 다음 정보를 입력하여 검색을 좁힐 수 있습니다(이 학습의 경우에는 이들 필드를 완료할 필요가 없습니다).

표 16. 일괄처리 검색 필드

필드 이름	설명
판매자	일괄처리를 검색하려는 판매자의 이름입니다. WebSphere Commerce Payments 데이터베이스에 500개 미만의 판매자가 있는 경우, 드롭 다운 목록에서 판매자 이름을 선택하십시오. WebSphere Commerce Payments 데이터베이스에 500개 이상의 판매자가 있는 경우 판매자의 이름을 입력하십시오.
일괄처리 번호	판매자 안에서 일괄처리를 고유하게 식별하는 번호입니다. 지불이 예치될 때 지정됩니다.

표 16. 일괄처리 검색 필드 (계속)

필드 이름	설명
상태	일괄처리의 상태: <ul style="list-style-type: none"> <li>• 열림</li> <li>• 종료</li> </ul>
대차 대조 상태	이 일괄처리의 대차 대조 상태: <ul style="list-style-type: none"> <li>• 대차 대조 일치 일괄처리가 성공적으로 대차 대조되었습니다. 즉, 모든 총계가 일치합니다.</li> <li>• 대차 대조 불일치 이 일괄처리를 일괄처리하려는 시도가 실패했습니다. 즉, 모든 총계가 일치하지 않습니다.</li> </ul>
지불 유형	주문하는 데 사용되는 지불 유형 또는 프로토콜을 식별합니다(예: OfflineCard).
일괄처리 시작 날짜	지정된 시간 범위 중에 시작된 일괄처리를 검색하려면 이후 및 이전 필드를 사용하십시오. <ul style="list-style-type: none"> <li>• 이후 해당 날짜 및 그 이후에 열린 모든 일괄처리를 검색할 날짜를 지정하십시오.</li> <li>• 이전 해당 날짜 및 그 이전에 열린 모든 일괄처리를 검색할 날짜를 지정하십시오.</li> </ul>
일괄처리 종료 날짜	지정된 시간 범위 중에 종료된 일괄처리를 검색하려면 이후 및 이전 필드를 사용하십시오. <ul style="list-style-type: none"> <li>• 이후 해당 날짜 및 그 이후에 종료된 모든 일괄처리를 검색할 날짜를 지정하십시오.</li> <li>• 이전 해당 날짜 및 그 이전에 종료된 모든 일괄처리를 검색할 날짜를 지정하십시오.</li> </ul>
계정	이 주문이 처리될 계정입니다. 500개 이상의 계정이 정의된 경우 입력 필드에 계정 번호를 입력하십시오.

### 3. 검색을 누르십시오.

**추가정보:** 이전 및 이후 필드를 사용하여 검색에서 특정 일괄처리를 제외해서 검색 결과를 좁힐 수도 있습니다. 예를 들어, 08/01/2003 이전 및 08/15/2003 이후에 시작된 모든 일괄처리를 검색하여 08/02/2003 및 08/14/2003 사이에 열린 일괄처리를 제외할 수 있습니다.

4. 제품 정보를 보려면 일괄처리 번호를 누르십시오.
5. 이 일괄처리의 모든 지불 및 대변의 자세한 목록을 보려면 일괄처리 정보를 누르십시오.
6. 결산을 눌러서 일괄처리를 결산하십시오. 처리가 완료될 때 결산 결과 페이지에 결산 상태가 표시됩니다.

시간이 지난 정보를 삭제하려면 결산 결과 페이지에서 삭제를 눌러서 결산된 일괄처리를 삭제할 수 있습니다. 일괄처리가 삭제될 때, 해당 일괄처리에 대한 모든 부수적인 정

보, 즉 지불, 대변 및 카세트 특정 데이터도 삭제됩니다. 모든 지불 데이터를 보유해야 하는 경우(감사에 대비한 경우와 같이) 일괄처리를 삭제하면 안됩니다.

## 11단계: 대변 발행

대변은 주문에 대해 발행되며 주문의 총액 한도 내에서 금액에 제한 없이 대변이 주어질 수 있습니다.

대변을 발행하려면 다음을 수행하십시오.

1. 대변을 발행하려는 주문을 찾으려면 탐색 프레임에서 주문 검색을 누르십시오.
2. 주문 검색 페이지에서 다음을 입력할 수 있습니다(이 학습의 경우에는 이들 필드를 완료할 필요가 없습니다).

표 17. 주문 검색 필드

필드 이름	설명
판매자	주문을 검색하려는 판매자의 이름입니다. WebSphere Commerce Payments 데이터베이스에 500개 미만의 판매자가 있는 경우, 드롭 다운 목록에서 판매자 이름을 선택하십시오. WebSphere Commerce Payments 데이터베이스에 500개 이상의 판매자가 있는 경우 판매자의 이름을 입력하십시오.
주문 번호	사용자가 지정하며 주문을 고유하게 식별하는 번호입니다.
상태	주문의 상태: <ul style="list-style-type: none"> <li>• 요청</li> <li>• 주문</li> <li>• 환불 가능</li> <li>• 취소</li> <li>• 종료</li> </ul>
지불 유형	주문하는 데 사용되는 지불 유형 또는 프로토콜을 식별합니다(예: OfflineCard).
주문 날짜	지정된 시간 범위 중에 시작된 주문을 검색하려면 이후 및 이전 필드를 사용하십시오. <ul style="list-style-type: none"> <li>• 이후 해당 날짜 및 그 이후에 열린 모든 주문을 검색할 날짜를 지정하십시오.</li> <li>• 이전 해당 날짜 및 그 이전에 열린 모든 주문을 검색할 날짜를 지정하십시오.</li> </ul>
주문 총계	<ul style="list-style-type: none"> <li>• 통화 이 주문에 사용되는 통화입니다. 드롭 다운 목록에서 통화 유형을 선택하십시오.</li> <li>• 초과 사용자가 지정하는 값보다 크거나 같은 주문 총계를 갖는 모든 주문을 검색할 값을 지정하십시오.</li> <li>• 미만 사용자가 지정하는 값보다 작거나 같은 주문 총계를 갖는 모든 주문을 검색할 값을 지정하십시오.</li> </ul>
계정	이 주문이 처리될 계정입니다. 500개 이상의 계정이 정의된 경우 입력 필드에 계정 번호를 입력하십시오.



3. 검색을 누르십시오.
4. 주문 검색 결과 페이지에서 환불 가능 상태에 있는 주문에 대한 주문 번호를 눌러서 해당 주문의 정보를 보십시오.
5. 주문 페이지에서 대변을 눌러서 이 주문에 대한 대변을 작성하십시오.
6. 대변 작성 페이지에 다음 정보가 표시됩니다.

표 18. 대변 작성 필드

필드 이름	설명
통화	이 주문에 사용되는 통화의 유형입니다. 이것은 읽기 전용 필드입니다.
주문 총계	주문에 사용되는 통화로 표현되는 주문의 총액입니다. 이것은 읽기 전용 필드입니다.
승인 총계	주문에 사용되는 통화로 표현되는 승인된 주문의 총액입니다. 이것은 읽기 전용 필드입니다.
예치금 총계	주문에 사용되는 통화로 표현되는 예치된 주문의 총액입니다. 이것은 읽기 전용 필드입니다.
대변 금액	이 필드는 판매자 관리자가 구매자에게 대변할 총액으로 완료해야 합니다.
승인 이유	수동 오프라인 승인 요청 프로세스에서 리턴되는 승인 코드입니다. 대변 상태가 환불로 변경됩니다.
거부 이유	수동 오프라인 승인 요청 프로세스에서 리턴되는 거부 이유입니다. 대변 상태가 거부로 변경됩니다.

대변 금액을 입력하고 대변을 누르십시오.

대변 처리가 완료되면 주문 페이지가 최신 정보로 고쳐지고 대변 상태가 표시됩니다. 새로 작성된 대변이 대변에 표시됩니다.

## 12단계: 일괄처리 총계 보기

이 학습의 마지막 단계는 매일의 결산 일괄처리 총계를 보는 것입니다. WebSphere Commerce Payments 보고서 기능을 사용하면 종료 상태에 있는 일괄처리에 대한 일간 총계를 볼 수 있습니다.

매일의 결산 일괄처리 총계 보고서를 작성하려면 다음을 수행하십시오.

1. 탐색 프레임에서 보고서를 누르십시오.
2. 보고서 페이지에서 매일의 결산 일괄처리 총계를 누르십시오.
3. 일괄처리 총계 보고서 페이지에서 일괄처리 총계 보고서를 보려는 해당 날짜를 입력하십시오. 현재 날짜에 대한 보고서를 작성하려면 이 필드를 공백으로 두십시오.
4. 판매자 이름을 입력하거나 선택하십시오. 판매자 이름을 입력하지 않으면 지정된 날짜에 대한 모든 일괄처리 목록이 표시됩니다. 500개 이상의 일괄처리가 있는 경우, 처음 500개의 일괄처리만이 표시됩니다.
5. 검색을 누르십시오.



매일의 결산 일괄처리 총계 보고서가 검색 페이지에 지정된 날짜에 종료된 모든 일괄처리의 총계를 계산합니다. 검색 날짜를 지정하지 않았으므로 작성된 보고서에는 현재 날짜의 일괄처리 총계가 들어 있습니다. 이 총계는 통화별로 계산되므로 통화당 하나의 행이 있습니다. 이 총계는 OfflineCard Cassette를 통해 작성된 것뿐 아니라 모든 지불 유형에 대해 작성된 지불 및 대변이 포함됩니다.

이제 지불 관리자와 판매자 관리자로서의 하루를 완료했습니다. 개별 비즈니스 모델은 많이 다를 수 있으나, 이 학습에서는 WebSphere Commerce Payments 작업을 설정하는 기본 경로를 요약하여 설명하고 WebSphere Commerce Payments에서 구현되는 기본 지불 처리를 설명합니다. WebSphere Commerce Payments 사용자 인터페이스의 특정 필드에 대한 추가 정보는 온라인 도움말을 참조하십시오.



## 부록 D. WebSphere Commerce Payments에 대한 자국어 지원 (NLS) 정보

이 장에서는 WebSphere Commerce Payments 자국어 지원(NLS)에 대한 보충 정보를 제공합니다. WebSphere Commerce Payments의 자국어 버전(NLV)을 설치하려는 경우 이 정보를 검토하는 것이 중요합니다.

### NLS 힌트와 추가정보

#### Adobe Acrobat Reader 제한 사항

AIX 플랫폼에서 일부 언어의 경우 PDF 파일이 올바르게 표시되지 않을 수 있습니다. WebSphere Commerce Payments PDF 파일(즉, 문서 파일)을 보거나 인쇄할 수 없는 경우, Windows<sup>®</sup> 또는 Windows 2000 운영체제의 브라우저에서 파일을 보거나 인쇄할 수 있습니다. 이 점은 AIX Adobe Acrobat Reader의 제한 사항입니다.

#### 코드 페이지

문자가 스페인어로 올바르게 표시되지 않는 경우, 코드 페이지 ISO-8859-1을 사용 중인지 확인하십시오.

#### WebSphere Commerce Payments 사용자 이름

WebSphere Commerce Payments는 다음 테이블의 문자로 구성된 사용자 이름만을 지원합니다. 이 세트 외의 문자를 포함하는 사용자 이름은 액세스할 수 없습니다.

이 테이블은 WebSphere Commerce Payments 사용자 이름 지정을 위해 지원되는 문자 세트의 목록을 표시합니다.

표 19. WebSphere Commerce Payments 사용자 이름 및 기타를 위한 휴대용 문자 세트(PCS)

ASCII 16진 값	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
문자	공백	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
ASCII 16진 값	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
문자	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
ASCII 16진 값	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
문자	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
ASCII 16진 값	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
문자	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	--

표 19. WebSphere Commerce Payments 사용자 이름 및 기타를 위한 휴대용 문자 세트(PCS) (계속)

ASCII 16진 값	60	61	62	63	64	65	66	67	68	69	6A	6B	6 C	6D	6E	6F
문자	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
ASCII 16진 값	70	71	72	73	74	75	76	77	78	79	7A	7B	7 C	7D	7E	
문자	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

## 지원되는 언어 및 로케일

다음 테이블은 WebSphere Commerce Payments가 지원하는 각 언어와 대응하는 로케일의 목록을 표시합니다.

표 20. WebSphere Commerce Payments에서 지원되는 언어 및 로케일

언어	로케일
포르투갈어	pt_BR
영어	en_US
프랑스어	fr_FR
독일어	de_DE
이탈리아어	it_IT
일본어	Ja_JP
한국어	ko_KR
중국어	zh_CN
스페인어	es_ES
대만어	Zh_TW

---

## 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 한국 IBM 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 수 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 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것을 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chrome, Minato-ku

Tokyo 1061, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다.

IBM은 타인의 권리 비침해, 상품성 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, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및(또는) 프로그램을 사전 통지없이 언제든지 개선 및(또는) 변경할 수 있습니다.

이 정보에서 언급되는 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.

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

- (1) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및
- (2) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램의 라이선스 사용자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩  
한국 아이.비.엠 주식회사  
고객만족센터

이러한 정보는 해당 조항 및 조건에 따라(예를 들면, 사용료 지불 포함) 사용할 수 있습니다.

이 정보에 기술된 라이선스가 있는 프로그램 및 이 프로그램에 대해 사용 가능한 모든 라이선스가 있는 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스에서부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로, 비IBM 제품과 관련된 성능의 확장성, 호환성 또는 기타 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위해 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

이 책은 다른 회사의 인터넷 사이트에 대한 참조를 포함하여 이러한 회사의 제품에 관한 정보를 포함할 수 있습니다. IBM은 이러한 정보의 정확성, 완전성 또는 사용에 대해 책임지지 않습니다.

이 제품은 SET 프로토콜을 기반으로 합니다.

저작권:

이 정보에는 여러 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 응용프로그램이 들어 있습니다. 귀하는 이러한 샘플 응용프로그램의 작성 기준이 된 운영 플랫폼의 응용프로그램 프로그래밍 인터페이스(API)에 부합하는 응용프로그램을 개발,

사용, 판매 또는 배포할 목적으로 추가 비용없이 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한 샘플 프로그램은 모든 조건하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이들 샘플 프로그램의 신뢰성, 서비스 가능성 또는 기능을 보증하거나 암시하지 않습니다. 귀하는 IBM의 응용프로그램 프로그래밍 인터페이스(API)에 부합하는 응용프로그램을 개발, 사용, 판매 또는 배포할 목적으로 추가 비용없이 이러한 샘플 응용프로그램을 어떤 형태로든 복사, 수정 또는 배포할 수 있습니다.

이러한 샘플 프로그램 또는 파생 제품의 각 사본이나 그 일부에는 반드시 다음과 같은 저작권 표시가 포함되어야 합니다.

(C) (귀하의 회사명) (연도). 이 코드의 일부는 IBM Corp.의 샘플 프로그램에서 파생됩니다. (C) Copyright IBM Corp. 1996, 2003. All rights reserved.

#### 상표

IBM 로고 및 다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록상표입니다.

- AIX
- CrossWorlds
- DB2 Extenders™
- DB2 Universal Database™
- IBM iSeries
- Intelligent Miner™
- Lotus®
- OS/400
- pSeries™
- S/390
- VisualAge®
- WebSphere
- xSeries®
- zSeries
- Redbooks
- SecureWay
- DFS
- VisualAge
- DB2
- AS/400

Solaris, Java 및 모든 Java 기반 상표 및 로고는 미국 또는 기타 국가에서 사용되는 Sun Microsystems의 상표 또는 등록상표입니다.

Microsoft®, Windows 및 Windows 로고는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 등록상표입니다.

SET 및 SET 로고는 SET Secure Electronic Transaction™ LLC 소유의 상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

기타 회사, 제품 또는 서비스 이름은 다른 회사의 상표이거나 서비스표입니다.









**IBM**