

IBM WebSphere Commerce



웹 서비스 구현 안내서

버전 5.5

IBM WebSphere Commerce



웹 서비스 구현 안내서

버전 5.5

주:

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 65 페이지의 『주의사항』의 정보를 읽으십시오.

초판(2003년 9월)

이 개정판은 새 개정판에 별도로 명시하지 않는 한 IBM WebSphere Commerce 버전 5.5 및 위에 나열된 제품의 모든 후속 릴리스 및 수정판에 적용됩니다. 제품 레벨에 맞는 올바른 버전을 사용하고 있는지 확인하십시오.

이 개정판은 새 개정판에 별도로 명시하지 않는 한, IBM WebSphere Commerce Studio, Business Developer Edition 버전 5.5와 IBM WebSphere Commerce Studio, Professional Developer Edition 버전 5.5 및 위에 나열된 제품의 모든 후속 릴리스와 수정판에 적용됩니다. 제품 레벨에 맞는 올바른 버전을 사용하고 있는지 확인하십시오.

책의 주문은 한국 IBM 담당자 또는 해당 지역의 IBM 지방 사무소로 문의하십시오.

IBM은 여러분의 의견을 환영합니다. 의견은 온라인 IBM WebSphere Commerce 문서 피드백 양식을 사용하여 보내주십시오. 양식은 다음 URL에서 사용 가능합니다.

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

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

목차

시작하기 전에	v
사전 요구사항	vi
이 책에 사용된 규칙	vi
기본 경로	vii
변경사항 요약	viii
제 1 장 소개	1
용어	1
개요	2
웹 서비스 사용 이유	2
웹 서비스 아키텍처	4
전제 조건	4
참조사항	5
제 2 장 WebSphere Commerce의 웹 서비스	7
WebSphere Commerce의 메시지 전달 인프라	7
WebSphere Commerce의 웹 서비스 구현 이해	7
서비스 제공자로 WebSphere Commerce 사용	8
요청 처리	8
응답	8
WebSphere Commerce를 서비스 제공자로 사 용가능하게 설정	9
서비스 요청자로서 WebSphere Commerce 사용	10
제 3 장 WebSphere Commerce를 서비스 제공자 로 사용	13
웹 서비스 인프라	13
아키텍처	13
보안 고려사항	15
웹 서비스 및 액세스 제어	15
전송 레벨 보안	16
클라이언트 코드를 생성하여 웹 서비스 액세스	17
외부 웹 클라이언트가 호출할 수 있는 견본 웹 서비 스	17
견본 코드 반입	17
OrderCreate 서비스(Business Edition)	18
OrderCreate 견본 웹 서비스 사용	19
견본 테스트	21
OrderCreate 웹 서비스의 보안	22
사용자 및 장기 구매 계약 작성	22
데이터베이스 테이블 대량 자료 반입	23

새 메시지 맵핑	26
메시지 맵퍼 구성	28
OrderStatus 서비스	28
OrderStatus 견본 웹 서비스 사용	28
OrderStatus 서비스에 대한 액세스 제어	32
OrderStatus 웹 서비스의 사용자 역할	33
OrderStatus WSDL 정의	33
새 메시지 맵핑	33
메시지 맵퍼 구성	35
OrderCreate 및 OrderStatus 웹 서비스 사용 안함	35
제 4 장 WebSphere Commerce를 서비스 요청자 로 사용	39
아키텍처	39
서비스 구현 찾기	40
서비스 구현 호출	41
WebSphere Commerce가 외부 시스템에서 호출할 수 있는 견본 웹 서비스	42
OrderFulfillment 서비스	42
인터페이스 WSDL 정의	42
태스크 명령	42
비즈니스 오브젝트	45
견본 테스트	47
OrderFulfillment 웹 서비스 사용 안함	48
부록 A. 견본 WSDL 정의 파일	51
부록 B. IBM WebSphere Studio Application Developer 사용	53
WSDL을 WebSphere Studio Application Developer로 반입	53
클라이언트 코드 및 데이터 유형 생성	54
웹 서비스 테스트	55
TCP 모니터를 사용하여 SOAP 메시지 추적	56
TCP 모니터 서버 인스턴스 작성	56
TCP 모니터 서버 인스턴스 구성	57
TCP/IP 모니터를 사용한 SOAP 메시지 추적	57
클라이언트 코드 사용자 정의	61
주의사항	65
상표	66

시작하기 전에

웹 서비스 구현 안내서는 WebSphere® Commerce 버전 5.5 Business Edition 및 Professional Edition의 비즈니스 프로세스를 웹 서비스로 제공하려는 제공자를 위해 작성되었습니다. 또한 WebSphere Commerce가 외부 시스템에서 호스팅하는 웹 서비스에 액세스하는 방법도 설명합니다. 이 안내서는 WebSphere Commerce의 웹 서비스를 사용하고 구현하는 방법을 이해해야 하는 개발자에게 도움을 줍니다. 웹 서비스를 구성하고 WebSphere Commerce에서 제공하는 기능을 사용하려는 시스템 운영자와 같은 기타 사용자도 이 안내서를 참조할 수 있습니다. 이 안내서를 보려면 WebSphere Commerce 프로그래밍 모델에 대해서 알고 있어야 합니다. 또한 웹 서비스 및 관련 프로토콜을 호출 및 구현하는 방법도 알고 있어야 합니다.

이 안내서는 WebSphere Commerce에서 웹 서비스 및 웹 서비스 클라이언트 작업을 가능하도록 방법에 관한 정보를 제공합니다. 이 안내서에서는 웹 서비스 및 그 아키텍처의 개요를 제공합니다. WebSphere Commerce의 메시지 전달 인프라, 웹 서비스를 가능하게 하는 WebSphere Commerce 런타임 지원 및 관련 참조사항에 대해 간략히 설명합니다. 이 안내서는 견본 웹 서비스와 관련 비즈니스 오브젝트 및 태스크 명령에 대해서도 설명합니다.

이 안내서는 다음 장으로 구성됩니다.

제 1 장. 소개 웹 서비스에 대한 간략한 개요와 이 안내서에서 사용되는 용어 정의, 전제 조건 및 관련 참조사항.

제 2 장. WebSphere Commerce의 웹 서비스 WebSphere Commerce의 메시지 전달 인프라에 대한 간략한 개요 및 WebSphere Commerce의 웹 서비스 아키텍처. WebSphere Commerce 비즈니스 프로세스를 웹 서비스로 제공하는 방법과 WebSphere Commerce가 외부 시스템에서 호스팅되는 웹 서비스에 액세스하는 방법을 설명합니다.

제 3 장. WebSphere Commerce를 웹 서비스 제공자로 사용 견본을 사용하여 WebSphere Commerce에 정의된 비즈니스 프로세스를 웹 서비스로 사용하는 방법을 설명합니다. WebSphere Commerce에 도입된 새 프로그램 요소와 OrderCreate 및 OrderStatus 견본 웹 서비스의 구현 정보를 설명합니다.

제 4 장. 웹 서비스 요청자로 WebSphere Commerce 사용 견본을 사용하여 WebSphere Commerce가 외부 시스템에서 정의한 서비스를 호출할 수 있는 방법을 설명합니다. WebSphere Commerce에 도입된 새 프로그램 요소와 OrderFulfillment 견본 웹 서비스의 구현 정보를 설명합니다.

부록 A. 견본 WSDL 정의 파일 견본 웹 서비스의 WSDL 정의 파일을 나열합니다.

부록 B. IBM® WebSphere Application Developer로 웹 서비스 클라이언트 코드 작성 IBM WebSphere Studio를 사용하여 웹 서비스 클라이언트 코드를 작성하는 방법을 설명합니다.

사전 요구사항

이 책의 정보를 사용하려면 다음과 같은 영역의 지식이 있어야 합니다.

- 웹 서비스 기술(예: SOAP, WSDL, UDDI)
- XML
- Java™
- JavaServer Pages 기술
- WebSphere Studio Application Developer(이 책에서는 WebSphere Studio라고도 함)
- WebSphere Commerce 프로그래밍 모델

이 책에 사용된 규칙

이 책은 다음과 같은 규칙을 사용합니다.

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

모노체 디렉토리 경로 및 파일 이름과 같이 정확하게 입력해야 하는 텍스트의 예를 나타냅니다.

가울임꼴 값을 입력해야 하는 변수 및 강조의 의미로 사용됩니다.

AIX AIX®용 WebSphere Commerce의 특정 정보를 표시합니다.

400 IBM @server iSeries™ (이전에는 AS/400®)용 WebSphere Commerce의 특정 정보를 표시합니다.

+ **Linux** xSeries Linux용 WebSphere Commerce의 특정 정보, @server zSeries® 및 S/390® Linux용 WebSphere Commerce의 특정 정보, @server iSeries Linux용 WebSphere Commerce의 특정 정보 및 @server pSeries Linux용 WebSphere Commerce의 특정 정보를 표시합니다.

Solaris Solaris Operating Environment software용 WebSphere Commerce의 특정 정보를 표시합니다.

Windows Windows®용 WebSphere Commerce의 특정 정보를 표시합니다.

DB2 DB2® Universal Database의 특정 정보를 표시합니다.

Oracle Oracle® 데이터베이스의 특정 정보를 표시합니다.

▶ Business

WebSphere Commerce Business Edition의 특정 정보를 표시합니다.

▶ Studio

WebSphere Commerce Studio, 버전 5.5(Professional 또는 Business Developer Editions)를 지칭합니다.

기본 경로

이 책은 다음과 같은 기본 설치 경로를 사용합니다.

WC_installdir WebSphere Commerce의 설치 경로를 표시합니다. 이 변수가 표시되면, WebSphere Commerce의 실제 설치 경로로 변경하십시오. 예를 들어, 다음과 같이 변경하십시오.

▶ Windows

C:\IBM\WebSphere\CommerceServer55

▶ AIX

/usr/WebSphere/CommerceServer55

▶ Linux

/opt/WebSphere/CommerceServer55

▶ Solaris

/opt/WebSphere/CommerceServer55

▶ 400

/QIBM/ProdData/CommerceServer55

WAS_installdir WebSphere Application Server의 설치 경로를 표시합니다. 이 변수가 표시되면, WebSphere Application Server의 실제 설치 경로로 변경하십시오. 예를 들어, 다음과 같이 변경하십시오.

▶ Windows

C:\IBM\WebSphere\AppServer

▶ AIX

/usr/WebSphere/AppServer

▶ Linux

/opt/WebSphere/AppServer

▶ Solaris

/opt/WebSphere/AppServer

▶ 400

/QIBM/ProdData/WebAS5/Base

WCStudio_installdir WebSphere Commerce Studio의 설치 경로를 표시합니다. 이 변수가 표시되면, WebSphere Commerce Studio의 실제 설치 경로로 변경하십시오. 예를 들어, 다음과 같이 변경하십시오.

▶ Studio C:\IBM\WebSphere\CommerceStudio55

*WCStudio_workspacedir*은 WebSphere Commerce Studio 작업 영역 디렉토리입니다. 예를 들어, 다음과 같이 변경하십시오.

▶ DB2 C:\WebSphere\workspace_db2

▶ Oracle C:\WebSphere\workspace_oracle

WAS_userdir WebSphere Application Server가 사용하는 모든 데이터의 디렉토리로 사용자가 수정하거나 구성해야 합니다. 이 값이 표시되면 다음과 같이 변경하십시오.

▶ 400 /QIBM/UserData/WebAS5/Base/*WAS_instance_name*

여기서 *WAS_instance_name*은 WebSphere Commerce 인스턴스가 연관된 WebSphere Application Server의 이름을 나타내는 변수입니다.

변경사항 요약

- + 최신 버전의 문서는 WebSphere Commerce 웹 사이트의 Technical Library 페이지에서 PDF로 보실 수 있습니다.
- + <http://www.ibm.com/software/commerce/library/>
- + 제품에 대한 최신 변경 사항은 제품 README 파일을 참조하십시오. README 파일은 WebSphere Commerce 웹 사이트에도 있습니다.
- + 이 문서의 갱신된 부분은 다음과 같은 개정 문자가 사용됩니다.
- + • "+"(플러스) 문자는 이 책에서 개정된 부분을 지시합니다.
- + 이 책에서 변경된 개정된 부분은 Linux에서만 적용됩니다.

제 1 장 소개

이 장에서는 웹 서비스의 개요에 대해 설명합니다. 이 책에서 사용하는 용어를 정의하고 전제 조건을 나열하며 정보에 대한 다른 관련 소스의 참조사항을 제공합니다.

용어

다음 용어는 이 책에서 사용된 것입니다.

서비스 제공자

서비스 제공자는 웹 서비스의 소유자입니다. 서비스 제공자는 웹 서비스를 작성하고 서비스 레지스트리 또는 직접적으로 서비스 클라이언트에 서비스를 공개하며 서비스에 대한 액세스를 제공하는 플랫폼을 호스팅합니다.

서비스 요청자

서비스 요청자는 웹 서비스와의 상호작용을 시작하는 응용프로그램입니다. 서비스 요청자의 한 예는 일반 인터넷 브라우저입니다. 다른 예로는 처리 요구사항으로 발생한 정보를 수집하기 위해 웹 서비스를 사용하는 응용프로그램이 있습니다. 서비스 요청자는 공개된 정보를 사용하여 서비스를 바인드하고 서비스를 호출합니다.

SOAP(Simple Object Access Protocol)

SOAP는 XML로 작성된 간단한 요청 및 응답 메시지를 전자 통신의 기본 프로토콜로 사용하기 위해 모델을 정의합니다. 이것은 사용하는 플랫폼 및 프로토콜은 독립적으로 운영되는 전송 메커니즘입니다. 이것을 사용하여 공개, 찾기, 바인드 및 호출 작업이 용이해집니다.

UDDI(Universal Description, Discovery, and Integration)

UDDI 저장소는 웹 서비스의 설명을 저장합니다. UDDI는 글로벌 전자 옐로우 페이지 그룹을 제공하며 사용 가능한 웹 서비스의 서비스 레지스트리를 정의하는 스펙입니다. UDDI는 회사가 자신을 서비스 제공자로 내세우면서 사용 가능한 상품 및 서비스에 대한 설명을 레지스트리에 공개할 수 있게 합니다.

웹 서비스

웹 서비스는 표준화된 XML 메시지 전달을 사용하여 액세스 가능한 네트워크 작업 전체를 보여주는 공개 인터페이스에 기반한 소프트웨어 구성요소입니다.

WSDL(Web services Description Language)

WSDL은 웹 서비스의 특정 인터페이스와 API(application programming interface)를 보여주는 방법을 제공합니다.

XML 맵핑 템플릿

XML 맵핑 템플릿(또는 맵핑 템플릿)은 인바운드 XML 메시지를 처리하

고 이를 WebSphere Commerce 명령 매개변수로 변환하는데 사용합니다. WebSphere Commerce의 맵핑 템플릿 파일은 수신되는 XML 메시지의 요소를 대상 명령 매개변수로 맵핑하는 방법을 정의합니다.

맵핑 템플릿에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오.

XML 메시지 맵퍼

WebSphere Commerce의 메시지 맵퍼는 수신되는 XML 메시지를 대상 명령에서 필요로 하는 Java 오브젝트로 변환합니다. 메시지 맵퍼는 *instance_name.xml* 구성 파일로 등록할 수 있으며 메시지 맵퍼 ID로 고유하게 식별됩니다. XML 메시지 맵퍼는 맵핑 템플릿 파일에 정의된 맵핑을 사용하여 XML 메시지를 대응 시켜 연결하는 Java 오브젝트 명령 매개변수로 변환합니다.

주: *instance_name*은 WebSphere Commerce 인스턴스의 이름입니다.

메시지 맵퍼에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 XML 메시지 맵퍼 정보를 참조하십시오.

개요

웹 서비스는 새로운 웹 응용프로그램입니다. 모든 것을 자체 포함하며(self-contained), 자체 기술적이며(self-describing) 모듈화되어 웹을 통해 공개되고 위치를 찾을 수 있고 호출될 수 있는 응용프로그램입니다. 웹 서비스는 단순 요청에서 복잡한 비즈니스 프로세스 범위까지 호출할 수 있는 기능을 수행합니다. 일단 웹 서비스가 전개 및 등록되면 다른 응용프로그램은 전개된 서비스를 찾아서 호출할 수 있습니다. 웹 서비스의 토대는 SOAP(Simple Object Access Protocol), WSDL(Web Services Description Language) 및 UDDI(Universal Description, Discovery, and Integration) 레지스트리와 같은 표준입니다.

WebSphere Commerce 버전 5.5는 WebSphere Commerce 비즈니스 기능을 웹 서비스로 사용할 수 있게 하는 코드와 문서를 포함합니다. 비즈니스 기능을 외부 시스템에서 액세스할 수 있는 웹 서비스로 사용할 수 있게 하여 WebSphere Commerce가 서비스 제공자가 되도록 할 수 있습니다. 또한 외부 시스템이 호스팅하는 웹 서비스를 호출하여 WebSphere Commerce가 서비스 요청자가 되게 할 수 있습니다.

웹 서비스 사용 이유

웹 서비스는 응용프로그램을 전보다 빠르고 쉽고 저렴한 방식으로 통합할 수 있습니다. 통합은 서비스 관점으로는 많지만 네트워크 프로토콜 관점으로는 적게 입력된 메시지에 기초하여 프로토콜 스택의 상위 레벨에서 발생하므로 비즈니스 기능이 여유있게 통합이 됩니다. 이러한 특성은 여러 기업 간이나 하나의 기업안에서 웹을 통해 비즈니스 기능을 연결하는데 이상적입니다. 기업 내부 및 외부에서 모두 응용프로그램 통합이 공통의 인프라를 사용하여 같은 방식으로 수행될 수 있도록 통합된 프로그래밍 모델을 제공함

니다. 웹 서비스의 통합 및 적용은 기존 언어 및 플랫폼을 사용하고 기존 응용프로그램을 채택하여 증가되는 방식으로 수행될 수 있습니다.

웹 서비스는 표준화된 XML 메시지 전달을 사용하여 액세스 가능한 작업 전체 보여주는 인터페이스입니다. 웹 서비스는 서비스 기술이라고 하는 공식 표준 XML 방식으로 작성됩니다. 서비스 기술에는 메시지 형식(작업 기술), 전송 프로토콜 및 위치를 포함하여 서비스와 상호작용하는데 필요한 모든 세부사항이 들어 있습니다. 인터페이스는 서비스 구현 세부사항을 숨김으로써 구현된 하드웨어 또는 소프트웨어 플랫폼 및 작성된 프로그래밍 언어와 독립적으로 사용될 수 있게 합니다.

WebSphere Commerce는 웹 서비스를 웹 서비스 제공자 또는 다른 웹 서비스의 요청자(클라이언트)로 사용되도록 설계되었습니다. 예를 들어, 외부 웹 서비스 클라이언트는 WebSphere Commerce에 연결하여 제공된 웹 서비스를 사용할 수 있습니다(그림 1 참조). 또는 WebSphere Commerce가 다른 웹 서비스에 연결하여 정보를 요청할 수 있습니다(그림 2 참조).

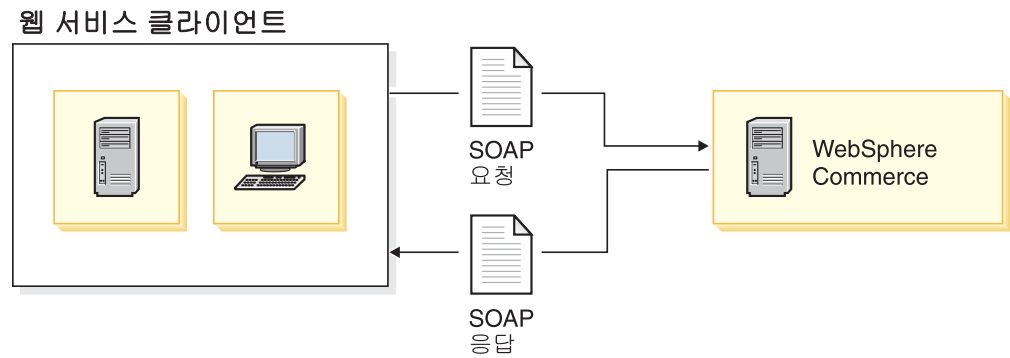


그림 1. 서비스 제공자로 WebSphere Commerce 사용

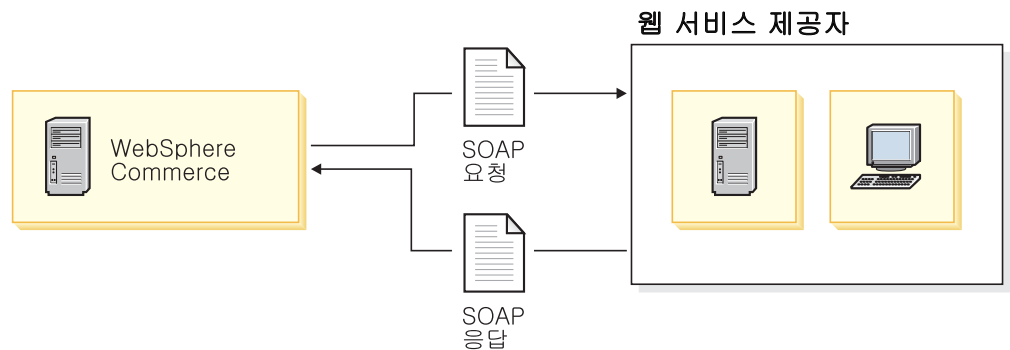


그림 2. 서비스 요청자로 WebSphere Commerce 사용

웹 서비스 아키텍처

웹 서비스 아키텍처는 세 가지 역할(서비스 제공자, 서비스 요청자, 서비스 브로커)과 세 가지 기본 작업(공개, 찾기, 바인드)으로 설명됩니다. 네트워크 구성요소는 이러한 역할 중 일부 또는 모두를 수행할 수 있습니다. 서비스 제공자는 웹 서비스를 서비스 브로커에 공개합니다. 서비스 요청자는 서비스 브로커를 사용하여 필요한 웹 서비스를 찾고 바인드합니다.

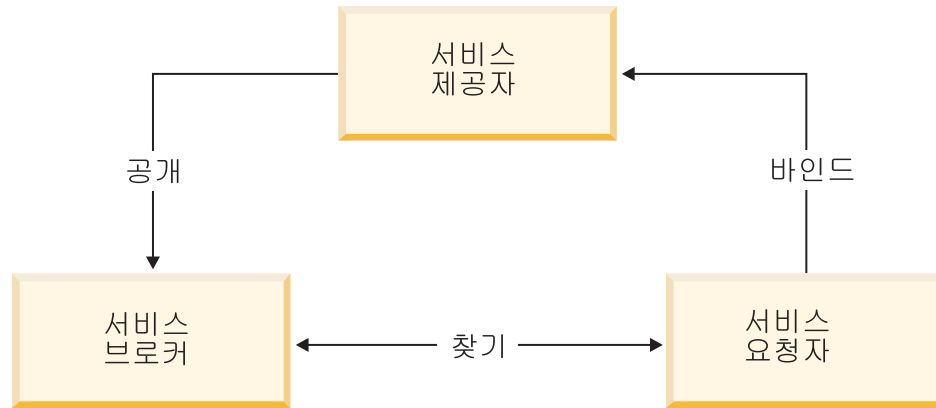


그림 3. 웹 서비스 아키텍처

전형적인 시나리오에서 서비스 제공자는 웹 서비스 구현인 네트워크 액세스 가능 서비스 모듈을 호스팅합니다. 서비스 제공자는 웹 서비스의 서비스 기술을 정의하고 이를 서비스 요청자나 서비스 레지스트리에 공개합니다. 서비스 요청자는 찾기를 기능 실행하여 서비스 브로커가 호스팅하는 서비스 레지스트리나 로컬에서 서비스 기술을 검색합니다. 서비스 요청자는 서비스 기술을 사용하여 서비스 제공자와 바인드하고 웹 서비스 구현을 호출하거나 상호작용합니다.

해당 솔루션은 런타임 지원을 제공하여 WebSphere Commerce에서 정의된 비즈니스 프로세스를 웹 서비스로 사용합니다. 기존 WebSphere Commerce 명령을 웹 서비스로 사용하는 방법의 예는 13 페이지의 제 3 장 『WebSphere Commerce를 서비스 제공자로 사용』을 참조하십시오. WebSphere Commerce가 외부 시스템에서 정의된 웹 서비스를 호출하는 방법에 대한 예는 39 페이지의 제 4 장 『WebSphere Commerce를 서비스 요청자로 사용』을 참조하십시오.

전제 조건

이 책에서 제공하는 정보를 사용하기 전에, IBM WebSphere Commerce 버전 5.5가 설치되었는지 확인하십시오. WebSphere Commerce 버전 5.5를 설치할 때, 제공되는 견본 웹 서비스를 설치하려면 사용자 정의 설치 옵션을 선택해야 합니다. 설치하려는 구성요소를 선택할 수 있습니다. **WebSphere Commerce 예제 파일**을 선택하십시오. 추가 정보는 *WebSphere Commerce 설치 안내서*를 참조하십시오.

또는, WebSphere Commerce Studio를 사용하여 견본 웹 서비스를 사용 가능하게 하는 경우 WebSphere Commerce Studio 버전 5.5의 설치를 완료했는지 확인하십시오. 추가 정보는 *WebSphere Commerce Studio 설치 안내서*를 참조하십시오.

견본 웹 서비스를 수정하려면, WebSphere Commerce Studio를 개발 환경으로 사용해야 합니다.

참조사항

웹 서비스 및 관련 기술에 대한 추가 정보는 다음 웹 사이트를 참조할 수 있습니다.

- W3C의 SOAP 제안의 구현인 Apache SOAP에 관한 정보는 <http://ws.apache.org/soap/> 참조
- SOAP 스펙에 관한 자세한 정보는 <http://www.w3.org/TR/SOAP/> 참조
- UDDI에 관한 정보는 <http://www.uddi.org/> 참조
- WSDL에 관한 정보는 <http://www.uddi.org/specification.html> 참조
- IBM Alphaworks의 웹 서비스에 관한 정보는 <http://www.alphaworks.ibm.com/webservices> 참조
- 개발자를 위한 perspective의 웹 서비스에 대한 정보는 <http://www.ibm.com/developerworks/webservices> 참조
- IBM WebSphere Commerce Business Edition 소프트웨어에 관한 전체 정보는 http://www.ibm.com/software/webservers/commerce/wc_be/ 참조
- IBM WebSphere Commerce Professional Edition 소프트웨어에 관한 전체 정보는 http://www.ibm.com/software/webservers/commerce/wc_pe/ 참조
- 최상의 WebSphere 웹 서비스 솔루션 개발을 위한 비즈니스 파트너 지원에 관한 정보는 <http://www.ibm.com/websphere/wow> 참조
- IBM WebSphere Studio에 관한 정보는 www.ibm.com/software/webservers/commerce/commercestudio/ 참조

주: 위의 웹 주소는 사전 통지없이 언제든지 변경될 수 있습니다. IBM은 비IBM 웹 사이트의 정보의 확실성 및 정확성에 대해서는 책임지지 않습니다.

제 2 장 WebSphere Commerce의 웹 서비스

이 장에서는 WebSphere Commerce의 메시지 전달 인프라를 간략히 설명하고 WebSphere Commerce의 웹 서비스 아키텍처 이해를 돕습니다. 또한 WebSphere Commerce 비즈니스 프로세스를 웹 서비스로 제공하는 방법과 WebSphere Commerce가 외부 시스템에서 호스팅되는 웹 서비스를 액세스하도록 하는 방법을 설명합니다.

WebSphere Commerce의 메시지 전달 인프라

WebSphere Commerce 런타임 아키텍처는 프로토콜 리스너, 어댑터, 컨트롤러, 명령 및 뷰를 포함합니다. WebSphere Commerce의 메시지 전달 인프라는 이러한 구성요소를 사용합니다.

프로토콜 리스너는 전송 장치로부터 인바운드 요청을 수신한 후 사용 프로토콜에 따라 요청을 적절한 어댑터로 보내는 WebSphere Commerce 런타임 구성요소입니다.

WebSphere Commerce 버전 5.5에서 RequestServlet은 SOAP 버전 1.1 메시지를 수신하는 프로토콜 리스너로 작용합니다. RequestServlet이 XML 요청을 수신하면 요청이 어댑터 관리자에게 전달됩니다. 어댑터 관리자는 어댑터 유형을 조회하여 요청을 처리할 수 있는 어댑터를 결정합니다. 특정 어댑터가 결정되면 요청이 해당 어댑터로 전달됩니다.

프로그램 어댑터는 XML 메시지가 HTTP 프로토콜을 통해 수신될 때 호출됩니다. 이를 사용하여 외부 시스템은 HTTP 프로토콜을 통해 XML 요청을 전달하여 WebSphere Commerce와 통신할 수 있습니다. 프로그램 어댑터는 조달 시스템, 공급자 시스템 등의 외부 시스템에 HTTP 프로토콜을 통해 WebSphere Commerce와 통신하는 일반적인 방법을 제공합니다. 이 방식으로 WebSphere Commerce는 외부 시스템과 완벽하게 통합됩니다.

요청 플로우에 관한 추가 정보는 *WebSphere Commerce 프로그래밍 안내서* 및 *학습서*를 참조하십시오.

WebSphere Commerce의 웹 서비스 구현 이해

WebSphere Commerce 프로그래밍 모델은 URL 명령, 컨트롤러 명령, 태스크 명령, 뷰 명령, 프로그램 어댑터, 웹 컨트롤러 등으로 구성됩니다. WebSphere Commerce 프로그래밍 모델에 관한 자세한 정보는 *WebSphere Commerce 프로그래밍 안내서* 및 *학습서*를 참조하십시오.

다음 절은 WebSphere Commerce에 서비스 제공자 및 서비스 요청자 역할을 부여하는 방법을 설명합니다.

서비스 제공자로 WebSphere Commerce 사용

WebSphere Commerce의 비즈니스 프로세스를 외부 시스템에서 액세스할 수 있는 웹 서비스로 사용하는 경우, WebSphere Commerce는 서비스 제공자가 됩니다. WebSphere Commerce 버전 5.5에는 SOAP 메시지 및 응답을 지원하는 메시지 전달 시스템의 향상된 기능이 포함됩니다. 또한, 제공된 견본은 기존 WebSphere Commerce 비즈니스 로직을 웹 서비스로 제공하는 방법을 보여줍니다.

클라이언트는 HTTP 또는 보안 HTTP 프로토콜을 사용하여 네트워크를 통해 URL 명령을 호출할 수 있습니다. 또한 WebSphere Commerce는 HTTP 및 HTTPS 프로토콜 또는 메시지 전달 미들웨어를 통해 XML 메시지를 수신할 수도 있습니다. 이 기능은 WebSphere Commerce가 HTTP를 통해 SOAP 메시지를 수신할 수 있도록 확장되었고 웹 서비스를 사용하는 데도 사용됩니다. RequestServlet은 WebSphere Commerce 시스템에 대한 모든 URL 요청의 시작점입니다. RequestServlet은 XML 메시지를 수신할 때 콘텐츠 유형을 XML로 식별하고 프로그램 어댑터를 호출합니다.

요청 처리

프로그램 어댑터는 WebSphere Commerce 명령을 호출하는 원격 프로그램 지원을 제공합니다. 이것은 요청을 수신하고 메시지 매퍼를 사용하여 요청을 명령에 대한 적절한 입력으로 변환합니다. 변환 후 프로그램 어댑터는 명령을 실행합니다.

WebSphere Commerce 버전 5.5에서는 SOAP XML 메시지의 구문 분석 및 처리를 손쉽게 해주는 새 메시지 매퍼 및 템플릿 맵핑 파일이 WebSphere Commerce 런타임 아키텍처에 추가되었습니다.

이 새로운 템플릿 맵핑 파일에는 수신되는 SOAP XML 메시지와 명령 매개변수 간 맵핑이 들어 있습니다. 새 메시지 매퍼는 이 템플릿 맵핑 파일을 찾아서 SOAP XML 메시지를 명령에 대한 적절한 입력으로 구문 분석한 후 입력을 다시 프로그램 어댑터로 전달합니다. 그런 후 프로그램 어댑터는 웹 컨트롤러를 호출하고 차례로 메시지 매퍼가 지정한 명령을 실행합니다.

응답

WebSphere Commerce에서 컨트롤러 명령이 성공적으로 실행되면 실행할 뷰 이름이 리턴됩니다. 웹 컨트롤러는 해당 뷰에 대한 뷰 명령을 호출하고 다시 전송할 메시지를 작성합니다. 응답 뷰를 작성하는 방법은 많습니다. 여기에는 다른 URL으로의 경로 재지정, JSP 템플릿으로의 이송 또는 응답에 대한 HTML 문서 작성 등이 포함됩니다.

다양한 종류의 클라이언트 장치를 위한 메시지 생성을 위해 WebSphere Commerce가 다양한 장치 형식을 지원합니다. 웹 컨트롤러는 수신되는 메시지의 장치 형식에 해당하는 뷰 이름을 선택합니다. WebSphere Commerce 버전 5.5에서는 새로운 장치 형식

인 SOAPHTTP가 SOAP XML 메시지를 위해 도입되었습니다. 동일한 뷰 이름(그러나 새 장치 형식)을 위한 새로운 JSP 템플리트를 정의할 수 있습니다. 이러한 JSP 템플리트는 SOAP 클라이언트에 리턴할 SOAP 응답을 작성합니다. 이것은 모델 뷰 컨트롤러 패턴을 따르며 비즈니스 로직에서 분리한 뷰를 사용합니다.

SOAP 응답을 처리하기 위해 새로운 뷰 명령 구현이 제공되었습니다. 이 뷰 명령 구현을 위한 인터페이스는 `com.ibm.commerce.me.soap.commands.SOAPViewCommand`이며 구현 클래스 이름은 `com.ibm.commerce.me.soap.commands.`

`HttpSOAPViewCommandImpl`입니다. 따라서 새 JSP 템플리트가 VIEWREG 테이블에 등록되어 SOAP 응답 메시지를 작성할 때마다 이 인터페이스 이름과 클래스 이름을 사용해야 합니다.

새 SOAP 뷰 명령 구현은 HTTP 경로 재지정 및 HTTP 이송 요청을 모두 처리할 수 있습니다. 일반적으로 HTTP 경로 재지정은 URL 경로 재지정에 의해 처리됩니다. SOAP 클라이언트는 URL 경로 재지정을 이해할 수 없기 때문에 이 SOAP 뷰 명령은 HTTP 경로 재지정을 HTTP 이송으로도 구현합니다.

WebSphere Commerce를 서비스 제공자로 사용가능하게 설정

다음은 WebSphere Commerce 비즈니스 프로세스를 웹 서비스로 사용하는 단계에 관한 개요입니다.

1. 웹 서비스로 제공하려는 비즈니스 로직을 지정하십시오.
2. 이 비즈니스 로직을 구현하는 URL 명령을 지정하십시오. 이것이 가능하지 않은 경우 새 URL 명령을 작성하십시오. 새 명령을 작성하여 기존 명령 집합을 호출할 수 있습니다.
3. 이 URL 명령에서 필요로 하는 필수 및 선택 매개변수를 식별하십시오.
4. 웹 서비스의 WSDL 정의를 정의하십시오.
5. 맵핑 템플리트 파일을 사용하여 수신되는 SOAP 요청을 명령 매개변수에 맵핑하십시오. `TemplateTag` 안에 WSDL 정의에 있는 모든 요소에 대한 맵핑이 들어 있는지 확인하십시오. 클라이언트의 요청을 전송할 때 SOAP 요청의 모든 요소에 대한 데이터를 반입했어야 합니다. 맵핑 템플리트 파일은 수신되는 SOAP XML 메시지의 각 요소가 XPATH 맵핑으로 명령 매개변수에 맵핑되는 방식을 정의합니다. TCP 모니터와 같은 유틸리티를 사용하여 SOAP 요청을 인터셉트한 후 인터셉트된 메시지를 사용하여 맵핑을 정의할 수 있습니다.

주: TCP 모니터는 WebSphere Studio Application Developer 5.0과 함께 사용 가능하며 추가 정보는 WebSphere Studio 도움말을 참조하십시오. TCP 모니터의 독립형 버전은 Apache AXIS 배포판에서 제공됩니다. 자세한 정보는 <http://ws.apache.org/axis/>를 참조하십시오.

6. JSP 템플리트를 작성하여 응답을 작성하십시오.

7. 서비스를 전개하십시오. 여기에는 관련 파일 및 자원에 대한 전개도 포함됩니다. 새 명령의 경우 WebSphere Commerce 명령 레지스트리에 전개하고 등록해야 합니다.

WebSphere Commerce의 비즈니스 로직은 컨트롤러 명령을 사용하여 구현되었습니다. 컨트롤러 명령은 웹 컨트롤러와 직접 상호작용하는 명령입니다. URL 명령은 URL을 사용하여 호출할 수 있는 컨트롤러 명령입니다. 결과적으로 웹 브라우저에 URL을 입력하여 URL 명령을 호출할 수 있습니다. 웹 서비스를 구현하기 전에 웹 서비스로 제공하려는 비즈니스 로직을 구현하는 URL 명령을 확인하십시오. 예를 들어, OrderStatus 명령으로는 WebSphere Commerce에서 기존 주문 상태를 갱신할 수 있습니다.

URL에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 참조 절을 참조하십시오.

서비스 요청자로서 WebSphere Commerce 사용

WebSphere Commerce를 사용하여 외부 시스템에서 호스팅하는 웹 서비스를 시작하는 경우, WebSphere Commerce가 서비스 요청자가 됩니다. WebSphere Commerce 버전 5.5에는 WebSphere Commerce 비즈니스 로직이 웹 서비스를 호출하여 다른 시스템과 통합되는 방법을 보여주는 견본이 들어 있습니다. WebSphere Commerce는 SOAP 기반 요청 및 응답을 전송하고 수신하는 웹 서비스 프록시를 통해 웹 서비스 요청을 전송합니다. 비즈니스 로직은 지정된 웹 식별 바인딩 인터페이스를 토대로 작동합니다. 제공된 견본은 이러한 인터페이스에 기초하여 웹 서비스의 호출을 지원합니다.

구현은 서비스 레지스트리에 공개되거나 로컬 파일 시스템에 저장될 수 있습니다. WebSphere Commerce에서 웹 서비스를 호출하는 데에는 다음의 작업이 포함됩니다.

1. WebSphere Commerce가 지원하는 기존 비즈니스 시나리오에 적합한 공개된 웹 서비스 인터페이스 정의를 식별합니다.
2. 서비스 구현에 대한 WSDL 정의를 로컬 WebSphere Commerce 시스템으로 다운로드합니다.
3. 웹 서비스 정의 파일을 *instance_name.xml* 구성 파일에 등록합니다(여기서 *instance_name*은 WebSphere Commerce Server 인스턴스의 이름입니다).
4. WebSphere Commerce가 사용할 수 있는 클라이언트 프록시를 작성합니다. IBM WebSphere Studio와 같은 도구를 사용하여 작성할 수 있습니다. 웹 서비스 인터페이스 정의의 클라이언트 프록시를 작성할 때 입력 메시지 및 출력 메시지에 필요한 데이터 유형도 작성됩니다. 추가 정보는 이 책의 17 페이지의 『클라이언트 코드를 생성하여 웹 서비스 액세스』 절을 참조하십시오.

또한 다른 도구(예: 웹 서비스 툴킷)를 사용하여 클라이언트 프록시를 생성할 수도 있습니다. <http://www.alphaworks.ibm.com/webservices>를 참조하십시오.

5. 로직을 구현하는 태스크 명령을 작성하여 다음을 수행하십시오.
 - a. 액세스 지점을 검색합니다.

- b. 웹 서비스의 프록시를 호출합니다.
 - c. 웹 서비스에서 리턴된 결과를 처리합니다.
6. WebSphere Commerce 명령 레지스트리를 갱신하여 태스크 명령을 등록합니다.

제 3 장 WebSphere Commerce를 서비스 제공자로 사용

이 장에서는 WebSphere Commerce에서 정의된 비즈니스 프로세스를 웹 서비스로 사용하는 방법을 설명합니다. 이 시나리오에서 WebSphere Commerce는 외부 시스템에 대한 서비스 제공자로 작동합니다. OrderCreate 및 OrderStatus 비즈니스 프로세스를 웹 서비스로 사용하는 방법을 보여주는 견본이 제공되었습니다.

웹 서비스 인프라

다음은 수신되는 웹 서비스 요청을 지원하는 WebSphere Commerce 런타임 요소입니다.

- SOAP 메시지 매퍼
- 웹 서비스 맵핑 템플릿
- SOAP 응답 뷰 명령

외부 사용자는 클라이언트 프로그램을 사용하여 WebSphere Commerce 서비스를 호출할 수 있습니다. 이러한 사용자는 SOAP 메시지를 전송 및 수신할 수 있는 모든 시스템에서 연결될 수 있습니다. WebSphere Commerce 버전 5.5는 HTTP 전송 프로토콜을 사용하는 SOAP 메시지를 지원합니다.

WebSphere Commerce 서비스는 WSDL 정의로 공개되어야 합니다. 웹 서비스 클라이언트는 WSDL 정의를 다운로드하고, 클라이언트를 생성하며 HTTP를 통한 SOAP를 사용하여 WebSphere Commerce 웹 서비스를 호출할 수 있습니다.

아키텍처

이 아키텍처의 메시지 플로우는 동기화됩니다. 이는 다음 처리를 실행하기 전에 외부 시스템이 WebSphere Commerce 시스템의 응답을 대기하는 것을 의미합니다. 인바운드 메시지 전달 시스템을 사용하여 외부 클라이언트는 웹 서비스로 공개된 WebSphere Commerce 비즈니스 프로세스를 호출할 수 있습니다.

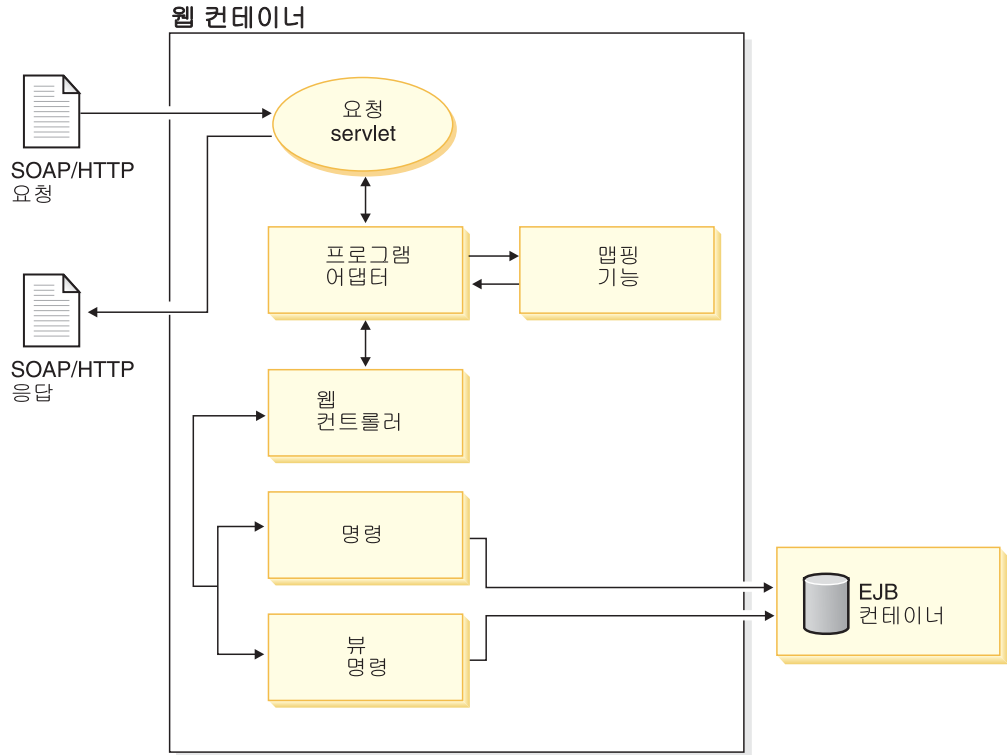


그림 4. WebSphere Commerce를 웹 서비스 제공자로 사용하기 위한 아키텍처

다음 목록은 외부 시스템이 웹 서비스를 사용하여 WebSphere Commerce 비즈니스 프로세스를 호출하는 경우의 상호작용에 대해 설명합니다.

1. 외부 시스템은 정확한 구조의 SOAP 요청을 WebSphere Commerce로 전송합니다.
2. RequestServlet은 요청을 수신하여 프로그램 어댑터에게 위임합니다.
3. 프로그램 어댑터는 등록된 SOAP 메시지 매퍼를 호출합니다.
4. SOAP 메시지 매퍼는 SOAP 메시지를 구문 분석하여 이름 값 쌍을 템플릿 파일을 사용하는 프로그램 어댑터에 리턴합니다. 이러한 템플릿 파일은 웹 서비스 요청에 있는 매개변수와 명령 매개변수 간 맵핑 정보를 포함합니다.
5. 프로그램 어댑터는 웹 컨트롤러를 호출하여 명령 이름과 이름 값 쌍을 웹 컨트롤러로 전달합니다.
6. 웹 컨트롤러는 명령을 실행합니다. 제공된 예에서는 BatchOrderRequest 및 OrderStatusCmd 명령이 호출됩니다.
7. 명령이 실행되면 뷰 이름이 웹 컨트롤러로 리턴됩니다.
8. 웹 컨트롤러는 등록된 뷰 명령을 실행하여 SOAP 응답을 작성하고 이를 프로그램 어댑터에 리턴합니다.
9. HTTP 프로그램 어댑터는 메시지를 RequestServlet에 리턴합니다.
10. RequestServlet은 대기 중인 외부 클라이언트에 응답을 전송합니다.

다음은 서비스 제공자에서 제공하는 웹 서비스를 지원하기 위해 추가된 견본입니다.

- `webservice_template.xml` 파일은 명령 특성을 사용하여 XPath를 맵핑시키기 위해 제공되었습니다.
- 다음 견본 JSP 템플릿은 SOAP 응답과 오류 메시지를 작성하기 위해 제공되었습니다.
 - `SOAP_OrderStatusResponse.jsp`
 - `SOAP_PurchaseOrderResponse.jsp`
 - `SOAP_GenericApplicationError.jsp`
 - `SOAP_GenericSystemError.jsp`

보안 고려사항

다음 절은 웹 서비스를 사용할 때 관련된 보안 고려사항에 대해 다룹니다.

웹 서비스 및 액세스 제어

액세스 제어는 WebSphere Commerce 시스템의 보호 자원에 대한 액세스를 권한이 부여된 개인 및 조직에게만 제한하는 프로세스입니다. 데이터베이스 관리를 용이하게 하고 보안을 보장하기 위해 WebSphere Commerce에는 자원에 대한 액세스를 특정 사용자에게 제한하는 액세스 제어 메커니즘이 있습니다. 액세스 제어는 다음과 같은 보안 지침으로 정의할 수 있습니다.

1. 시스템이 관리하는 자원에 대한 시스템 액세스를 사용자에게 허용하거나 거부합니다.
2. 사용자가 각 자원에서 수행할 수 있는 조치를 지정합니다.

WebSphere Commerce의 액세스 제어 모델은 액세스 제어 정책의 이행을 기본으로 합니다. 액세스 제어 및 정책에 대해 학습하려면 *WebSphere Commerce* 보안 안내서를 참조하십시오.

WebSphere Commerce가 서비스 제공자로 작동하는 경우, 외부 웹 서비스 클라이언트는 컨트롤러 명령에 정의된 비즈니스 프로세스를 호출할 수 있습니다. 결과적으로 웹 서비스는 웹 서비스 클라이언트가 호출한 명령에 적용 가능한 액세스 제어 정책을 상속합니다. 따라서 신임장 정보는 WebSphere Commerce와 함께 사용된 각 WSDL 정의에 대해 정의되어야 합니다. 사용자를 인증하는데 필요한 신임장을 전달하는 것 외에도 웹 서비스 클라이언트는 사용자에게 명령을 실행하는데 필요한 권한이 있는지 확인해야 합니다.

신임장에 대한 WSDL 정의의 XML 스키마 형식은 다음과 유사해야 합니다.

```
<complexType
name="com.ibm.commerce.webservice.datatype.Credentials">
<all>
```

```

<element name="password" nillable="true" type="string"/>
<element name="logonId" nillable="true" type="string"/>
</all>
</complexType>

```

신임장 정보는 맵핑 템플릿 파일에서 사용 가능한 템플릿 정의의 다음 필드에 맵핑되어야 합니다.

- 사용자 로그인 필드의 이름은 'logonId'이고 FieldInfo는 'CONTROL'입니다.
- 암호의 필드 이름은 'logonPassword'이고 FieldInfo는 'CONTROL'입니다.

전송 레벨 보안

전송 레벨 보안을 사용하기 위해 HTTPS를 통해 SOAP 메시지를 전송할 수 있습니다. 서비스 제공자는 HTTPS를 사용하여 SOAP 메시지를 전송할 것을 강력히 권장합니다. 그렇지 않으면 메시지를 인터셉트하는 모든 사람이 전달된 신임장을 볼 수 있습니다.

주: HTTPS를 사용하려면 SOAP 클라이언트는 프로토콜 이름으로 HTTPS를 사용하는 URL을 사용해야 합니다. 클라이언트 호출을 호출하기 전에 SSL(Secure Sockets Layer)을 사용 가능하게 하는 특성을 설정하십시오. 웹 서비스 클라이언트에 IBM Developer Kit, Java Technology Edition 버전 1.2 이상을 사용하는 경우, HTTPS를 사용 가능하게 하는 코드는 다음과 유사합니다.

```
System.setProperty("java.protocol.handler.pkgs",
com.ibm.net.ssl.internal.www.protocol");
```

```
java.security.Security.addProvider
(new com.ibm.jsse.JSSEProvider());
```

웹 서비스 클라이언트를 개발에 WebSphere Commerce Studio를 사용하는 경우 앞의 코드가 사용됩니다.

Sun Microsystems의 J2SE SDK(Java 2 Standard Edition Software Development Kit)를 사용하는 경우, HTTPS를 통해 웹 서비스를 호출하는 클라이언트 코드는 다음과 유사합니다.

```
System.setProperty("java.protocol.handler.pkgs",
com.sun.net.ssl.internal.www.protocol");
```

```
java.security.Security.addProvider
(new com.sun.net.ssl.internal.ssl.Provider());
```

WebSphere Commerce에서 정의한 인바운드 서비스를 호출하려고 할 때 클라이언트 측에서 '데이터 교환 실패', '알 수 없는 CA' 또는 '서버 예외로 연결이 거부됨'을 수신하는 경우, 클라이언트가 서버에 있는 인증을 인식하지 못한다는 것을 의미합니다. 여기에 해당하고 서버를 신뢰하는 경우, 신뢰 인증서 목록에 서버 인증을 추가할 수 있습니다.

클라이언트 코드를 생성하여 웹 서비스 액세스

이 절에서는 클라이언트 코드 생성 방법에 대해 설명합니다.

견본 웹 서비스는 WSDL 정의 파일로 공개되며 다음 디렉토리에 있습니다.

-  `WC_installdir\samples\webservices\xml\messaging`
-  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging`
-    `WC_installdir/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`


WSDL 정의를 사용하여 클라이언트 응용프로그램을 웹 서비스 요청자로 개발할 수 있습니다. 적절한 도구를 사용하여 웹 서비스의 WSDL 정의로부터 WebSphere Commerce가 구현한 웹 서비스를 액세스하는데 필요한 클라이언트 코드 및 데이터 유형을 생성할 수 있습니다.


IBM WebSphere Studio Application Developer(이 책에서는 WebSphere Studio라고도 함)는 WSDL 정의로부터 클라이언트 및 프록시 코드를 생성하는데 사용할 수 있는 도구 중 하나입니다. 자세한 정보는 53 페이지의 『부록 B. IBM WebSphere Studio Application Developer 사용』을 참조하십시오.

클라이언트 프록시를 생성하는데 사용할 수 있는 또다른 도구는 웹 서비스 툴킷입니다. <http://www.alphaworks.ibm.com/webservices>를 참조하십시오.

외부 웹 클라이언트가 호출할 수 있는 견본 웹 서비스

이 절에서는 다음 견본 웹 서비스를 사용하는 방법에 대해 설명합니다.

1.  OrderCreate 서비스
2. OrderStatus 서비스

WebSphere Commerce 시스템 또는 WebSphere Commerce Studio가 설치된 개발 시스템에서 견본 웹 서비스를 사용할 수 있습니다. WebSphere Commerce Studio 시스템에서 견본 웹 서비스를 사용하는 경우, 이절의 지시사항을 따르고  라고 표시된 디렉토리 경로를 참조하십시오.

견본 코드 반입

견본 웹 서비스를 편집하거나 수정하려면 견본 소스 코드를 WebSphere Commerce Studio로 반입해야 합니다. 이 코드를 반입하려면 다음을 수행하십시오.

1. 시작 > 프로그램 > **IBM WebSphere Commerce Studio** > **WebSphere Commerce** 개발 환경에서 WebSphere Commerce 개발 환경을 여십시오.
2. J2EE perspective로 전환하고 J2EE 탐색기 보기를 선택하십시오.
3. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 펼치십시오. **src** 디렉토리를 마우스 오른쪽 버튼으로 누르고 반입을 누르십시오. 반입 마법사가 열립니다.
4. 파일 시스템을 선택하고 다음을 누르십시오.
5. 찾아보기를 누르고 다음 디렉토리를 찾으십시오.
`WCStudio_installdir\Commerce\samples\webservices\com`
6. 모두 선택을 누른 후, 완료를 누르십시오.
7. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 마우스 오른쪽 버튼으로 누르고 특성을 선택하십시오.
8. **Java** 빌드 경로를 누른 후 라이브러리 탭으로 이동하십시오.
9. 변수 추가를 누르십시오. SOAPJAR 및 XERCES 변수를 선택하고 확인을 누르십시오.
10. 확인을 누르십시오.
11. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 마우스 오른쪽 버튼으로 누르고 프로젝트 다시 빌드를 선택하십시오.

이제 WebSphere Commerce Studio로 견본 소스 코드 반입을 완료했습니다. WebSphere Test Environment에서 WebSphereCommerceServer 서버를 시작하는 방법에 관한 추가 정보는 *WebSphere Commerce Studio* 설치 안내서를 참조하십시오.

OrderCreate 서비스(Business Edition)

OrderCreate 웹 서비스는 WebSphere Commerce Business Edition 및 WebSphere Commerce Business Developer Edition에 적용됩니다.



이 절에서는 WebSphere Commerce 의 비즈니스 로직을 조달 응용프로그램과 같은 외부 응용프로그램의 주문 수신을 지원하는 웹 서비스로 사용하는 방법에 대해 보여줍니다. 조달 시스템은 WebSphere Commerce BatchOrderRequest 명령을 사용하여 WebSphere Commerce에서 주문을 작성할 수 있습니다. 이 견본 OrderCreate 웹 서비스 구현은 구매자 조직의 주문을 수신하는 웹 서비스를 작성하는 방법에 대해 설명합니다. OrderCreate 견본 웹 서비스를 사용하려면, WebSphere Commerce 시스템의 공개된 상점이 필요합니다.

일반적으로 판매자는 WebSphere Commerce에서 주문을 작성하기 전에 구매자 조직을 알고 있어야 합니다. 또한 구매자가 OrderCreate 웹 서비스를 호출할 수 있으려면 구매자 조직이 WebSphere Commerce에 등록되어 있어야 합니다.


OrderCreate 견본 웹 서비스 사용

견본 OrderCreate 웹 서비스를 사용하려면 다음을 수행하십시오.

1. 웹 서비스로 제공하려는 비즈니스 로직을 지정하십시오. 이 경우 구매자 조직이 조달 시스템과 유사한 응용프로그램을 사용하여 구매 주문을 작성할 수 있게 하십시오.
2. 이 비즈니스 로직을 구현하는 명령을 지정하십시오. 이 경우엔 BatchOrderRequest 명령입니다.
3. BatchOrderRequest 명령에서 필요로 하는 필수 및 선택 매개변수를 식별하십시오. BatchOrderRequest 명령의 설명은 WebSphere Commerce Production 및 Development 온라인 도움말에 있습니다. 전체 매개변수 목록에 대한 정보는 다음 디렉토리에 있는 webservice_template.xml 파일을 참조하십시오.

-  `WC_installdir\samples\webservices\xml\messaging`
 -  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging`
-    `WC_installdir/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`


4. OrderCreate 웹 서비스에 대해 WSDL 정의를 작성하십시오. 견본 OrderCreate.wsd1 정의 파일은 다음 디렉토리에 있습니다.

-  `WC_installdir\samples\webservices\xml\messaging`
 -  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging`
-    `WC_installdir/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/ samples/webservices/xml/messaging`

이 WSDL 정의를 수정하여 제공하는 서비스의 정의 요구사항을 충족시킬 수 있습니다.

OrderCreate 웹 서비스에 공개된 WSDL 정의는 이 웹 서비스를 호출하는데 필요한 템플릿 및 데이터 유형 스키마를 바인딩하는 액세스 지점을 포함하고 있습니다.

5. 맵핑 템플릿 파일을 사용하여 수신되는 SOAP 요청을 명령 매개변수에 맵핑시키십시오. 견본 맵핑 템플릿은 다음 디렉토리에 있습니다.

-  `WC_installdir\samples\webservices\xml\messaging\webservice_template.xml`

-  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging\webservice_template.xml`
-    `WC_installdir/samples/webservices/xml/messaging/webservice_template.xml`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/webservice_template.xml`

이 파일을 사용자 정의할 수 있습니다. 기본 WSDL 정의를 수정한 경우, WSDL 정의의 변경사항과 일치시키기 위해 이 기본 맵핑 템플릿을 수정해야 할 수 있습니다.







6. OrderCreate 기본 웹 서비스를 포함하는 템플릿을 사용하려면 다음을 수행하십시오.

a. webservice_template.xml 파일을 다음 디렉토리로 복사하십시오.

-  `WC_installdir\xml\messaging`
-  `WCStudio_installdir\Commerce\xml\messaging`
-    `WC_installdir/xml/messaging`
-  `/QIBM/UserData/CommerceServer55/instances/instance_name/xml/messaging`

b. webservice_template.xml 파일이 이미 존재하고 아무 정보도 가지고 있지 않으면 겹쳐쓰십시오. 파일에 정보가 있으면 OrderCreate 서비스에 상응하는 TemplateDocument 정보를 기존 파일로 복사하십시오.

7. "SOAPINTEGRATION"에 대한 메시지 맵퍼를 사용하십시오. 다음 디렉토리에 있는 instance_name.xml 구성 파일을 여십시오.

-  `WC_installdir\instances\instance_name\xml`
-  `WCStudio_installdir\Commerce\instances\instance_name\xml`
-    `WC_installdir/instances/instance_name/xml`
-  `/QIBM/UserData/CommerceServer55/instances/instance_name/xml`







"WCS.SOAPINTEGRATION"이라는 메시지 맵퍼를 지정하십시오. "enable" 속성 값이 "false"이면 "true"로 설정하십시오.

8. JSP 템플릿을 작성하여 응답을 작성하십시오. 이 시나리오에서는 다음 기본 JSP 템플릿을 복사하십시오.







- SOAP_PurchaseOrderResponse.jsp - OrderCreate 요청에 대한 SOAP 응답 메시지를 작성합니다.

- SOAP_GenericApplicationError.jsp - 응용프로그램 오류시 SOAP 오류 응답을 작성합니다.
- SOAP_GenericSystemError.jsp - 시스템 오류시 SOAP 오류 응답을 작성합니다.

다음 디렉토리에서 복사하여

-  `WC_installdir\samples\webservices\wcstores`
 -  `WCStudio_installdir\Commerce\samples\webservices\wcstores`
-    `WC_installdir/samples/webservices/wcstores`
-  `WC_installdir/samples/webservices/wcstores`

다음 디렉토리로 복사하십시오.

-  `WAS_installdir\installedApps\cell_name`
 \ `WC_instance_name.ear\Stores.war`
 여기서 `cell_name`은 WebSphere Application Server가 설치된 시스템의 이름입니다.
-  `WCStudio_workspacedir\Stores\Web Content`
-    `WAS_installdir/installedApps/cell_name`
 `WC_instance_name.ear/Stores.war`
-  `WAS_userdir/installedApps/cell_name/ WC_instance_name.ear/Stores.war`

9. 구매자 조직을 작성하고 구매자와 공급자 조직 간에 장기 구매 계약을 작성하십시오. 추가 정보는 이 책의 22 페이지의 『사용자 및 장기 구매 계약 작성』을 참조하십시오.
10. 이 책의 23 페이지의 『데이터베이스 테이블 대량 자료 반입』에서 설명한 대로 데이터베이스 테이블로 대량 자료 반입을 수행하십시오.
11. WebSphere Commerce를 사용하는 경우 변경사항을 적용하려면 Commerce Server 인스턴스를 다시 시작하십시오. WebSphere Commerce Studio 시스템을 사용하는 경우, WebSphere Test Environment에서 WebSphereCommerceServer 서버를 다시 시작하십시오.

견본 테스트

WSDL 정의로부터 웹 서비스 클라이언트를 생성할 수 있습니다. 클라이언트는 PurchaseOrderRequest 메시지를 보내어 이 웹 서비스를 호출할 수 있습니다. 주문이 성공적으로 제출되면 응답 메시지에는 상태 코드(200), "완료" 상태 메시지, 상태 텍스트 및 WebSphere Commerce에서 작성된 OrderID가 포함됩니다. 오류시에는 해당 오류 메시지를 받게 됩니다.

OrderCreate 웹 서비스의 보안

이 견본에서 웹 서비스로 제공되는 비즈니스 로직에서는 요청하는 구매자 조직이 인증되어야 합니다. WebSphere Commerce 액세스 제어 모델에서 명령 및 관련 웹 서비스에 대한 인증은 요청자의 WebSphere Commerce 사용자 ID 및 암호를 제공함으로써 수행될 수 있습니다. 이 때 사용자는 WebSphere Commerce 아용자로 등록되어 있어야 합니다.

견본 OrderCreate 웹 서비스를 사용하려는 사용자를 인증하려면 사용자가 속한 구매자 조직이 WebSphere Commerce에 등록되어 있어야 합니다. OrderCreate 서비스를 처리하기 전에 전달된 신임장을 검증해야 합니다.

OrderCreate 웹 서비스의 신임장에는 WebSphere Commerce 시스템의 공급자 및 구매자에 대한 조직 코드 및 조직 코드 유형이 들어 있습니다. 클라이언트는 ORGCODE 테이블에 등록된 것과 동일한 값의 조직 코드 및 조직 코드 유형을 전송해야 합니다.

다음은 WSDL 정의에 있는 신임장에 대한 스키마입니다. 여기서 identity는 조직 코드이고 domain은 조직 코드 유형이며 shared secret는 구매자 및 공급자 조직에서 공유하는 암호입니다.

```
<complexType
name="com.ibm.commerce.webservice.datatype.ordercreate.
Credential">
<all>
<element name="identity" nillable="true" type="string"/>
<element name="domain" nillable="true" type="string"/>
<element name="sharedSecret" nillable="true" type="string"/>
</all>
```

주: shared secret은 전송자 신임장에만 적용할 수 있습니다.

액세스 제어

WebSphere Commerce는 OrderCreate 메시지를 수신할 때 BatchOrderRequest 명령을 호출하여 수신되는 요청을 처리합니다. OrderCreate 웹 서비스를 사용하려면 WebSphere Commerce를 사용하는 공급자와 웹 서비스 클라이언트로 작동하는 구매자 간에 장기 구매 계약이 존재해야 합니다.

사용자 및 장기 구매 계약 작성

구매자 조직이 존재하지 않으면 OrderCreate 웹 서비스를 호출하기 전에 다음을 완료하십시오. 다음 단계를 수행하여 새 구매자 조직을 작성하고 조달 구매자 역할을 지정하고 판매자 및 구매자 조직 간에 장기 구매 계약을 작성하십시오.

1. 관리 콘솔을 사용하여 구매자 비즈니스 조직을 작성하십시오. 자세한 정보는 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오.
2. 이 구매자 조직에 다음 역할을 지정하십시오.

- a. 조달 구매자
 - b. 조달 구매자 관리자
 - c. 조달 관리자
3. 관리 콘솔에서 이 구매자 조직의 사용자를 작성하십시오. 이 사용자의 사용자 ID는 구매자 조직의 그룹 사용자 ID이어야 하고 암호는 구매자 조직의 공용 암호여야 합니다. 그룹 사용자 ID 및 공용 암호는 15 페이지의 『보안 고려사항』에서 언급한 구매자 조직 신임장의 일부입니다.
 4. 3단계에서 작성한 사용자에게 다음 역할을 지정하십시오.
 - a. 조달 구매자
 - b. 조달 구매자 관리자
 5. WebSphere Commerce 액셀러레이터를 사용하여 이 구매자 조직의 비즈니스 계정을 작성하십시오. 비즈니스 계정 작성 방법에 관한 지시사항은 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오. 비즈니스 계정을 작성할 때 다음 사항에 유의하십시오.
 - 고객 페이지에서 고객에게 상점의 기본 장기 구매 계약 하에 구매 허용 선택란을 선택하지 마십시오.
 - 구매 주문 페이지에서 주문시 구매 주문 번호를 지정할 수 있음 선택란을 선택하지 마십시오.
 - 신용 한도 지불 방법을 사용하고 신용 한도 페이지에서 신용 한도 계정 번호를 지정하십시오.
 6. 작성된 비즈니스 계정의 장기 구매 계약을 작성하십시오. 비즈니스 계정의 장기 구매 계약 작성 방법에 대한 지시사항은 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오. 장기 구매 계약을 작성할 때 다음 사항에 유의하십시오.
 - 일반 탭을 누르십시오. 장기 구매 계약 이름 및 설명을 입력하고 장기 구매 계약 활성 기간을 선택하십시오.
 - 고객 탭을 누르십시오. 이전에 작성한 구매자 조직을 선택하십시오.
 - 지불 탭을 누르십시오. 신용 한도 사용이 선택되었는지 확인하십시오.
 7. 제출을 눌러 장기 구매 계약을 활성화하십시오.







앞의 단계에서는 WebSphere Commerce에서 구매자 조직의 계정과 공급자(WebSphere Commerce) 조직 및 구매자 조직간의 장기 구매 계약을 작성합니다. 필요한 액세스 권한이 구매자 조직에 지정됩니다.

데이터베이스 테이블 대량 자료 반입












BatchOrderRequest 명령을 실행하려면 데이터베이스 스크립트를 실행하여 몇 가지 테이블에 대량 자료 반입을 수행해야 합니다.

데이터베이스를 갱신하려면 다음을 수행하십시오.

1. 다음 디렉토리로 이동하십시오.







-  `WC_installdir\samples\webservices\bin`
-  `WCStudio_installdir\Commerce\samples\webservices\bin`
-     `WC_installdir/samples/webservices/bin`


2. 다음 테이블에 설명된 대로 구성에 대해 올바른 스크립트 파일 세트를 찾으십시오.

운영체제	데이터베이스 유형	스크립트 파일 이름
 		Webservices_DBUpdate.db2.bat Webservices_UndoDBUpdate.db2.bat
		Webservices_DBUpdate.oracle.bat Webservices_UndoDBUpdate.oracle.bat
 		Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh
		Webservices_DBUpdate.oracle.sh Webservices_UndoDBUpdate.oracle.sh
 		Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh

+
+
+
+
+







3. 해당 스크립트 파일을 다음 디렉토리로 복사하십시오.


-  `WC_installdir\samples\webservices\bin`
-  `WCStudio_installdir\Commerce\samples\webservices\bin`
-    `WC_installdir/samples/webservices/bin`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/bin`





4.  `wcs.webservices.enable.sql` 및 `wcs.webservices.disable.sql` 스크립트를 다음 디렉토리에서

-  `WC_installdir\samples\webservices\schema\db2`
-  `WCStudio_installdir\Commerce\samples\webservices\schema\db2`
-    `WC_installdir/samples/webservices/schema/db2`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/schema/db2`





다음 디렉토리로 복사하십시오.

-  `WC_installdir\schema\db2`
 -  `WCStudio_installdir\Commerce\schema\db2`
-    `WC_installdir/schema/db2`
-  `/QIBM/UserData/CommerceServer55/
WebSphere_Commerce_instance_name/schema`

 `wcs.webservices.enable.sql` 및 `wcs.webservices.disable.sql` 스크립트를 다음 디렉토리에서

-  `WC_installdir\samples\webservices\schema\oracle`
 -  `WCStudio_installdir\Commerce\samples\webservices\
schema\oracle`
-   `WC_installdir/samples/webservices/schema.oracle`

다음 디렉토리로 복사하십시오.






-  `WC_installdir\schema\oracle`
 -  `WCStudio_installdir\Commerce\schema\oracle`
-   `WC_installdir/schema/oracle`


5. 4단계에서 복사한 `wcs.webservices.enable.sql` 및 `wcs.webservices.disable.sql` SQL 스크립트에 사용된 다음 기본값을 OrderCreate 웹 서비스를 호출하는 구매자 조직의 정보에 맞게 수정하십시오.

- `WebServicesUserId` - 구매자 조직의 조직 코드입니다. 이것은 조직 코드 유형과 함께 구매자 조직을 고유하게 식별합니다.
- `Webservices@ibm.com` - 구매자 조직의 조직 코드 유형입니다.
- `aurora` - 공급자 조직의 조직 코드입니다. 일반적으로 이것은 조직의 D-U-N-S 번호이거나 조직의 다른 표준 고유 식별자입니다(D-U-N-S 번호는 비즈니스 식별 및 추적에 대한 업계의 글로벌 표준으로 사용되는 9자리 숫자입니다).
- `DUNS` - 공급자 조직의 조직 코드 유형입니다. 조직 코드의 DUNS 번호를 사용하는 경우 "DUNS"를 사용하십시오.





6. SQL 스크립트를 갱신한 후에 해당 명령 스크립트 파일을 실행하십시오.

 `DB2` DB2 데이터베이스를 사용하는 경우:

-   `DB2` 명령창에서 `Webservices_DBUpdate.db2.bat`를 실행하십시오.
-    명령창에서 `Webservices_DBUpdate.db2.sh`를 실행하십시오.

-  QSH 세션을 시작하십시오. /QIBM/ProdData/CommerceServer55/samples/webservices/bin/ Webservices_DBUpdate.db2.sh 명령을 실행하십시오.

 Oracle 데이터베이스를 사용하는 경우:

-   명령창에서 Webservices_DBUpdate.oracle.bat를 실행하십시오.
-   명령창에서 Webservices_DBUpdate.oracle.sh를 실행하십시오.

WC_installdir\logs\Webservices_DBUpdate.log 로그 파일을 확인하고 실행이 완료되었는지 확인하십시오.

주: OrderCreate 웹 서비스만 사용하려는 경우, OrderFulfillment 웹 서비스의 ExtOrderProcessServiceCmdImpl 및 WSDLFinderCmdImpl 태스크 명령을 등록하는 SQL 스크립트의 행을 주석 처리해야 합니다.

새 메시지 매핑







OrderCreate 웹 서비스를 사용하도록 webservice_template.xml 파일과 함께 새 기본 매핑 세트가 제공되었습니다. 다음은 메시지 매핑의 템플릿 문서입니다.

```
<!-- mapping for inbound order create message -->
<TemplateDocument>
  <DocumentType version='1.0'>ns1:orderCreate</DocumentType>
  <StartElement>ns1:orderCreate</StartElement>
  <TemplateTagName>BatchOrderCreate10Map</TemplateTagName>
  <CommandMapping>
    <Command CommandName='BatchOrderRequest'>
      <Constant Field='protocolName'>SOAP</Constant>
      <Constant Field='protocolVersion'>1.1</Constant>
    </Command>
  </CommandMapping>
</TemplateDocument>
```

이 서비스에 대응하는 메시지 매핑은 BatchOrderCreate10Map이며 다음 디렉토리의 webservice_template.xml 템플릿 파일에서 제공됩니다.

-  WC_installdir\samples\webservices\xml\messaging
 — 
 WCStudio_installdir\Commerce\samples\webservices\xml\messaging
-    WC_installdir/samples/webservices/xml/messaging
-  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging

제공된 메시지 매핑은 다음 디렉토리에 지정된 WSDL 정의에 대응시키십시오.

-  `WC_installdir\samples\webservices\xml\messaging\OrderCreate.wsdl`
-  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging\OrderCreate.wsdl`
-    `WC_installdir/samples/webservices/xml/messaging/OrderCreate.wsdl`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/OrderCreate.wsdl`

제공된 WSDL 정의에서 생성된 웹 서비스 클라이언트가 전송한 SOAP 요청 메시지로 작업합니다.

메시지 맵핑 정의에는 SOAP 요청의 XPATH 정보를 알고 있어야 합니다. WSDL 정의에서 XPATH 정보를 얻을 수 있습니다. 또는 SOAP 메시지에서 찾는 것이 좀더 편리할 수 있습니다. 다음 절에서는 SOAP 메시지 액세스 방법 및 맵핑 템플릿 정의 방법에 대해 설명합니다.

해당 SOAP 요청에서 OrderCreate 메시지(BatchOrderCreate10Map)로의 대응과 유사한 메시지 맵핑을 정의하려면 다음을 수행하십시오.

1. WSDL 정의를 사용하여 클라이언트 프록시를 생성하십시오. 53 페이지의 『부록 B. IBM WebSphere Studio Application Developer 사용』에 설명된 단계를 참조하십시오.
2. 전송 레벨 인터셉션 유틸리티를 사용하여 SOAP 요청 및 SOAP 응답 문서를 보십시오. 예를 들어, WebSphere Studio Application Developer 5.0에서 사용가능한 TCP 모니터 또는 다른 HTTP 터널링 유틸리티가 메시지를 로깅할 수 있습니다. 액세스 지점을 설정하여 클라이언트 프록시가 인터셉터를 지정하게 하십시오. SOAP 요청을 보면 요청의 맵핑 작성 및 응답을 작성하는 JSP 파일 작성에 도움이 됩니다.
3. 템플릿 정의 파일의 TemplateDocument 요소를 정의하십시오. 여기에는 SOAP XML 메시지 수신시 실행되는 명령 정보가 들어 있습니다. 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 "템플릿 정의 파일의 TemplateDocument 요소" 절을 참조하십시오.
4. 2단계에서 확보한 SOAP 요청을 사용하여 태그 정의 목록을 포함하는 TemplateTag를 정의하십시오. TemplateTag가 WSDL 정의에 있는 모든 요소의 맵핑을 포함하는지 확인하십시오. 클라이언트의 요청을 전송할 때 SOAP 요청의 모든 요소에 대한 데이터를 반입했어야 합니다. 각 태그 정의는 수신되는 SOAP XML 메시지의 요소 또는 속성을 명령 매개변수에 맵핑합니다. WebSphere Commerce Production 및 Development 온라인 도움말의 "템플릿 정의 파일의 TemplateDocument 요소" 절을 참조하십시오.

메시지 매퍼 구성

SOAP 메시지 매퍼는 `webservice_template.xml` 웹 서비스 템플릿 정의 파일을 사용하여 수신되는 SOAP 요청을 명령 특성에 맵핑합니다. 결과적으로 템플릿을 변경하여 SOAP 메시지 매퍼를 구성할 수 있습니다.

다른 WSDL 정의가 있거나 서비스 정의를 변경하려는 경우, 템플릿의 맵핑을 수정해야 합니다. 맵핑은 명령 매개변수에 맵핑된 XML 메시지의 XPATH 정의로 구성됩니다. 이 경우 새 서비스 유형에 따라 데이터 유형의 스키마에 맞게 XPATH 구조를 변경해야 합니다.

OrderStatus 서비스






이 절에서는 WebSphere Commerce에 제출된 주문 상태를 갱신하는 WebSphere Commerce 비즈니스 로직을 웹 서비스로 사용하는 방법을 보여줍니다. OrderStatus 웹 서비스를 사용하여 웹 서비스 호출을 통해 WebSphere Commerce OrderStatus 명령을 호출할 수 있습니다. OrderStatus 컨트롤러 명령으로 외부 시스템은 WebSphere Commerce의 기존 주문 상태와 관련된 정보를 갱신할 수 있습니다. OrderStatus 견본 웹 서비스를 사용하려면 WebSphere Commerce 시스템의 공개된 상점에서 실행된 주문이 있어야 합니다.

제공된 견본은 웹 서비스 요청의 WebSphere Commerce OrderStatus 명령 사용 방법을 보여줍니다.



OrderStatus 견본 웹 서비스 사용

다음은 견본 OrderStatus 웹 서비스를 구현하는 단계에 대한 개요입니다.







1. 웹 서비스로 제공하려는 비즈니스 로직을 지정하십시오. 이 경우, 주문을 이행하는 응용프로그램이 WebSphere Commerce의 주문 상태를 갱신하도록 하십시오.
2. 이 비즈니스 로직을 구현하는 명령을 지정하십시오. 여기에서는 견본 OrderStatusService 명령입니다. 이것은 기존 OrderStatus 명령보다 향상된 명령입니다.
3. 이 명령에서 필요로 하는 필수 및 선택 매개변수를 식별하십시오. 이러한 정보는 WebSphere Commerce Production 및 Development 온라인 도움말에 있습니다.
4. OrderStatus 웹 서비스에 대해 WSDL 정의를 작성하십시오. 견본 OrderStatus.wsd1 정의 파일은 다음 디렉토리에 있습니다.

-  `WC_installdir\samples\webservicess\xml\messaging`
-  `WCStudio_installdir\Commerce\samples\webservicess\xml\messaging`
-    `WC_installdir/samples/webservicess/xml/messaging`

-  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging
5. OrderCreate 서비스를 사용하는 경우 이 단계는 실행하지 마십시오. 그렇지 않은 경우에는 맵핑 템플릿 파일을 사용하여 수신되는 SOAP 요청을 명령 매개변수에 맵핑시키십시오. 견본 맵핑 템플릿은 다음 디렉토리에 있습니다.
-  *WC_installdir*\samples\webservices\xml\messaging\webservice_template.xml
 -  *WCStudio_installdir*\Commerce\samples\webservices\xml\messaging\webservice_template.xml
 -    *WC_installdir*/samples/webservices/xml/messaging/webservice_template.xml
 -  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/webservice_template.xml
- a. webservice_template.xml 파일을 다음 디렉토리로 복사하십시오.
-  *WC_installdir*\xml\messaging
 -  *WCStudio_installdir*\Commerce\xml\messaging
 -    *WC_installdir*/xml/messaging
 -  /QIBM/UserData/CommerceServer55/instances/*instance_name*/xml/messaging
- b. webservice_template.xml 파일이 이미 존재하고 아무 정보도 가지고 있지 않으면 겹쳐쓰십시오. 파일에 정보가 있으면 OrderStatus 서비스에 상응하는 TemplateDocument를 기존 파일로 복사하십시오.
6. OrderCreate 서비스를 사용하는 경우 이 단계는 실행하지 마십시오. 그렇지 않으면 "SOAPINTEGRATION"에 대한 메시지 맵퍼를 사용하십시오. 다음 디렉토리에 있는 *instance_name.xml* 구성 파일을 여십시오.
-  *WC_installdir*\instances*instance_name*\xml
 -  *WCStudio_installdir*\Commerce\instances*instance_name*\xml
 -    *WC_installdir*/instances/*instance_name*/xml
 -  /QIBM/UserData/CommerceServer55/instances/*instance_name*/xml
- "WCS.SOAPINTEGRATION"이라는 메시지 맵퍼를 지정하십시오. "enable" 속성 값이 "false"이면 "true"로 설정하십시오.
7. JSP 템플릿을 작성하여 응답을 작성하십시오. 응답을 작성할 수 있는 견본 JSP 파일 SOAP_OrderStatusResponse.jsp가 제공됩니다.
- a. SOAP_OrderStatusResponse.jsp를 다음 디렉토리에서
-  *WC_installdir*\samples\webservices\wcstores




-  `WCStudio_installdir\Commerce\samples\webservices\wcstores`
-    `WC_installdir/samples/webservices/wcstores`
-  `WC_installdir/samples/webservices/wcstores`

다음 디렉토리로 복사하십시오.







-  `WAS_installdir\installedApps\cell_name\WC_instance_name.ear\Stores.war`
-  `WCStudio_installdir\Commerce\IBM WebSphere Test Environment\hosts\default_host\default_app\web`
-    `WAS_installdir/installedApps/cell_name/WC_instance_name.ear/Stores.war`
-  `WAS_userdir/installedApps/cell_name/WC_instance_name.ear/Stores.war`

이 JSP 템플릿은 WebSphere Commerce에서 주문 상태가 성공적으로 갱신되면 SOAP 응답 메시지를 작성합니다.

b. OrderCreate 서비스를 사용하는 경우 이 단계를 실행하지 않아도 됩니다. 그렇지 않은 경우에는 SOAP_GenericApplicationError.jsp 및 SOAP_GenericSystemError.jsp 파일을 다음 디렉토리에서

-  `WC_installdir\samples\webservices\wcstores`
-  `WCStudio_installdir\Commerce\samples\webservices\wcstores`
-    `WC_installdir/samples/webservices/wcstores`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/wcstores`

다음 디렉토리로 복사하십시오.

-  `WAS_installdir\installedApps\cell_name\WC_instance_name.ear\Stores.war`
-  `WCStudio_workspacedir\Stores\Web Content`
-    `WAS_installdir/installedApps/cell_name/WC_instance_name.ear/Stores.war`
-  `WAS_userdir/installedApps/cell_name/WC_instance_name.ear/Stores.war`

이러한 JSP 템플릿은 응용프로그램 또는 시스템 오류시 SOAP 오류 메시지를 작성합니다.

8. OrderCreate 서비스를 사용하는 경우 이 단계를 실행하지 않아도 됩니다. 그렇지 않은 경우에는 다음을 수행하십시오.

a. 다음 테이블에 설명된 대로 구성에 대해 올바른 스크립트 파일 세트를 찾으십시오.

운영체제	데이터베이스 유형	스크립트 파일 이름
Windows Studio	DB2	Webservices_DBUpdate.db2.bat Webservices_UndoDBUpdate.db2.bat
	Oracle	Webservices_DBUpdate.oracle.bat Webservices_UndoDBUpdate.oracle.bat
AIX Solaris	DB2	Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh
	Oracle	Webservices_DBUpdate.oracle.sh Webservices_UndoDBUpdate.oracle.sh
400 Linux	DB2	Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh

+
+
+
+
+

b. 해당 스크립트 파일을 다음 디렉토리로 복사하십시오.

- Windows WC_installdir\samples\webservices\bin
- Studio WCStudio_installdir\Commerce\samples\webservices\bin
- AIX Linux Solaris WC_installdir/samples/webservices/bin
- 400 /QIBM/ProdData/CommerceServer55/samples/webservices/bin

c. DB2 wcs.webservices.enable.sql 및 wcs.webservices.disable.sql 스크립트를 다음 디렉토리에서

- Windows WC_installdir\samples\webservices\schema\db2
- Studio WCStudio_installdir\Commerce\samples\webservices\schema\db2
- AIX Linux Solaris WC_installdir/samples/webservices/schema/db2
- 400 /QIBM/ProdData/CommerceServer55/samples/webservices/schema/db2

다음 디렉토리로 복사하십시오.

- **Windows** `WC_installdir\schema\db2`
- **Studio** `WCStudio_installdir\Commerce\schema\db2`
- **AIX** **Linux** **Solaris** `WC_installdir/schema/db2`
- **400** `/QIBM/ProdData/CommerceServer55/schema/db2`

Oracle `wcs.webservices.enable.sql` 및 `wcs.webservices.disable.sql` 스크립트를 다음 디렉토리에서

- **Windows** `WC_installdir\samples\webservices\schema\oracle`
- **Studio** `WCStudio_installdir\Commerce\samples\webservices\schema\oracle`
- **AIX** **Solaris** `WC_installdir/samples/webservices/schema/oracle`

다음 디렉토리로 복사하십시오.

- **Windows** `WC_installdir\schema\oracle`
- **Studio** `WCStudio_installdir\Commerce\schema\oracle`
- **AIX** **Solaris** `WC_installdir/schema/oracle`

d. OrderStatus 웹 서비스와 관련되지 않은 스크립트 명령문을 주석 처리하여 7 단계에서 복사한 SQL 스크립트 파일을 갱신하십시오.

e. 7b단계에서 복사한 스크립트 파일을 실행하십시오.

9. WebSphere Commerce 시스템을 사용하는 경우, 변경사항을 적용하려면 Commerce Server 인스턴스를 다시 시작하십시오. WebSphere Commerce Studio 시스템을 사용하는 경우, WebSphere Test Environment에서 WebSphereCommerceServer 서버를 다시 시작하십시오.

OrderStatus 서비스에 대한 액세스 제어

OrderStatus 웹 서비스의 경우 WebSphere Commerce 액세스 제어 모델이 사용됩니다. WebSphere Commerce가 요청자를 인증하고 요청자가 필요한 조치를 수행할 권한이 있는지 확인할 수 있도록 웹 서비스 요청 시 사용자 ID 및 암호를 전달해야 합니다. 사용자 ID 및 암호가 다음과 같이 WSDL 정의에 정의되어 있습니다.

신임장의 XML 스키마 형식은 다음과 유사해야 합니다.

```
<complexType
name="com.ibm.commerce.webservice.datatype.Credentials">
<all>
<element name="password" nillable="true" type="string"/>
<element name="logonId" nillable="true" type="string"/>
</all>
</complexType>
```

신임장 정보는 템플릿 정의의 다음 필드에 맵핑되어야 합니다.

- 사용자 로그인 필드 이름은 'logonId'이고 FieldInfo는 'CONTROL'입니다.
- 암호 필드 이름은 'logonPassword'이고 FieldInfo는 'CONTROL'입니다.

예를 들어, OrderStatus 서비스의 경우 로그온 ID 및 암호는 맵핑 템플릿의 다음 절을 사용하여 OrderStatus 명령에 맵핑되어 있습니다.

```
<TemplateTag name='OrderStatus20Map'>
  <!--Execution Environment -->
  <Tag XPath='updateOrderStatusMessage/controlArea/credentials/logonId'
    Field='logonId' FieldInfo='CONTROL' />
  <Tag XPath='updateOrderStatusMessage/controlArea/credentials/password'
    Field='logonPassword' FieldInfo='CONTROL' />
```

이 견본은 WebSphere Commerce에 정의된 액세스 제어 요구사항을 충족시키는 일반적인 방법을 제공합니다.

OrderStatus 웹 서비스의 사용자 역할


OrderStatus 서비스는 OrderStatusServiceCmd 및 OrderStatusCmd 컨트롤러 명령을 사용합니다. 기본적으로 이러한 컨트롤러 명령들은 자원으로서

SellersCmdResourceGroup 자원 그룹에, 조치로서 BackendOrderStatusCreate 조치 그룹에 추가되어 있습니다. 결과적으로 '판매자' 역할에 지정된 사용자만

OrderStatusCmd 및 OrderStatusServiceCmd 명령을 실행할 수 있습니다. 즉, 신입장이 OrderStatus 메시지의 일부로 전달된 사용자에게 '판매자' 역할이 있어야 한다는 것입니다(기본적으로, WebSphere Commerce 운영자 권한이 있는 사용자는 이러한 명령을 모두 실행할 수 있습니다. 결과적으로 운영자 권한이 없는 사용자에 대해 OrderStatus 서비스를 사용하려는 경우에 한해서만 이 사항이 적용됩니다).

OrderStatus WSDL 정의

OrderStatus 웹 서비스는 다음 디렉토리에 있는 OrderStatus WSDL 정의와 함께 정의되었습니다.

-  `WC_installdir\samples\webservices\xml\messaging`
-  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging`
-    `WC_installdir/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`

새 메시지 맵핑

OrderStatus 웹 서비스를 사용하도록 맵핑 템플릿 `webservice_template.xml` 파일에 새 맵핑 세트가 제공되었습니다. 다음은 메시지 맵핑의 템플릿 문서입니다.


```
<!-- mapping for inbound order status update message -->
<TemplateDocument>
```

```






<DocumentType version='1.0'>ns1:update_WCS_OrderStatus
</DocumentType>
<StartElement>ns1:update_WCS_OrderStatus</StartElement>
<TemplateTagName>OrderStatus20Map</TemplateTagName>
<CommandMapping>
  <Command CommandName='OrderStatusService'>
    Condition='OrderStatusCommand'>
  <Constant Field='redirecturl'>
    SOAP_OrderStatusResponse
  </Constant>
  <Constant Field='SOAP_OrderStatusResponse'>
    SOAP_OrderStatusResponse.jsp
  </Constant>
  </Command>
</CommandMapping>
</TemplateDocument>

```

이 서비스에 대응하는 메시지 맵핑은 OrderStatus20Map이며 다음 디렉토리의 webservice_template.xml 템플릿 파일에서 제공됩니다.

-  *WC_installdir*\samples\webservices\xml\messaging
-  *WCStudio_installdir*\Commerce\samples\webservices\xml\messaging
-    *WC_installdir*/samples/webservices/xml/messaging
-  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging

제공된 메시지 맵핑은 다음 디렉토리에 지정된 WSDL 정의에 대응시키십시오.

-  *WC_installdir*\samples\webservices\xml\messaging\Orderstatus.wsd1
-  *WCStudio_installdir*\Commerce\samples\webservices\xml\messaging\Orderstatus.wsd1
-    *WC_installdir*/samples/webservices/xml/messaging/Orderstatus.wsd1
-  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/OrderStatus.wsd1

메시지 맵핑은 제공된 WSDL 정의에 생성된 웹 서비스 클라이언트가 전송한 메시지로 작업합니다.

메시지 맵핑 정의에는 SOAP 요청의 XPATH 정보를 알고 있어야 합니다. WSDL 정의에서 XPATH 정보를 얻을 수 있습니다. 또는 SOAP 메시지中寻找는 것이 좀더 편리할 수 있습니다. 다음 절에서는 SOAP 메시지 액세스 방법 및 맵핑 템플릿 정의 방법에 대해 설명합니다.

해당 SOAP 요청에서 OrderStatus 메시지(OrderStatus20Map)로의 대응과 유사한 메시지 매핑을 정의하려면 다음을 수행하십시오.

1. WSDL 정의 파일을 사용하여 클라이언트 프록시를 생성하십시오. 추가 정보는 53 페이지의 『부록 B. IBM WebSphere Studio Application Developer 사용』에 설명된 단계를 참조하십시오.
2. 전송 레벨 인터셉션 유틸리티를 사용하여 SOAP 요청 및 SOAP 응답 문서를 보십시오. 예를 들어, Apache AXIS 배포판에서 사용 가능한 TCP 모니터 또는 다른 HTTP 터널링 유틸리티가 메시지를 로그할 수 있습니다. 액세스 지점을 설정하여 클라이언트 프록시가 인터셉터를 지정하게 하십시오. 이것은 요청의 매핑 작성 및 응답을 작성하는 JSP 파일 작성에 도움이 됩니다.
3. 템플릿 정의 파일의 TemplateDocument 요소를 정의하십시오. 여기에는 XML 메시지 수신시 실행되는 명령 정보가 들어 있습니다. 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말의 "템플릿 정의 파일의 TemplateDocument 요소" 절을 참조하십시오.
4. 태그 정의 목록을 포함하는 TemplateTag를 정의하십시오. TemplateTag가 WSDL 정의에 있는 모든 요소의 매핑을 포함하는지 확인하십시오. 클라이언트의 요청을 전송할 때 SOAP 요청의 모든 요소에 대한 데이터를 반입했어야 합니다. 각 태그 정의는 수신되는 XML 메시지의 요소 또는 속성을 명령 매개변수에 매핑합니다. WebSphere Commerce Production 및 Development 온라인 도움말의 "템플릿 정의 파일의 TemplateDocument 요소" 절을 참조하십시오.

메시지 매퍼 구성

SOAP 메시지 매퍼는 webservice_template.xml 웹 서비스 템플릿 정의 파일을 사용하여 수신되는 SOAP 요청을 명령 특성에 매핑합니다. 결과적으로 템플릿을 변경하여 SOAP 메시지 매퍼를 구성할 수 있습니다. 인바운드 메시지 템플릿 정의 파일에 관한 추가 정보는 WebSphere Commerce Production 및 Development 온라인 도움말을 참조하십시오.

다른 WSDL 정의가 있거나 서비스 정의를 변경하려는 경우, 템플릿의 매핑을 수정해야 합니다. 매핑은 명령 매개변수에 매핑된 XML 메시지의 XPATH 정의로 구성됩니다. 이 경우 새 서비스 유형에 따라 데이터 유형의 스키마에 맞게 XPATH 구조를 변경해야 합니다.







OrderCreate 및 OrderStatus 웹 서비스 사용 안함

이 절에서는 WebSphere Commerce에서 제공하는 OrderCreate 및 OrderStatus 웹 서비스를 사용하지 않는 방법을 설명합니다.

1. 다음 JSP 파일을
 - SOAP_PurchaseOrderResponse.jsp

- SOAP_GenericApplicationError.jsp
- SOAP_GenericSystemError.jsp
- SOAP_OrderStatusResponse.jsp

다음 디렉토리에서 삭제하십시오.

-  *WAS_installdir*\installedApps\cell_name\
WC_instance_name.ear\Stores.war
-  *WCStudio_workspacedir*\Stores\Web Content
-    *WAS_installdir*/installedApps/cell_name/
WC_instance_name.ear/Stores.war
-  *WAS_userdir*/installedApps/cell_name/ WC_instance_name.ear/
Stores.war

2. 다음 디렉토리의 webservice_template.xml 파일을







-  *WC_installdir*\xml\messaging
-  *WCStudio_installdir*\Commerce\xml\messaging
-    *WC_installdir*/xml/messaging
-  /QIBM/UserData/CommerceServer55/instances/
instance_name/xml/messaging

다음 예제에서 표시되는 빈 ECTemplate 노드를 포함하는 빈 템플릿으로 대체하십시오.





```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE ECTemplate SYSTEM 'ec_template.dtd'>
<ECTemplate>
</ECTemplate>
```

3. 다음 디렉토리에서 wcs.webservices.disable.sql 스크립트를 찾으십시오.

 DB2 데이터베이스를 사용하는 경우:

-  *WC_installdir*\schema\db2
-  *WCStudio_installdir*\Commerce\schema\db2
-    *WC_installdir*/schema/db2
-  QIBM/UserData/CommerceServer55/instances/*instance_name*/
schema

 Oracle 데이터베이스를 사용하는 경우:







-  *WC_installdir*\schema\oracle
-  *WCStudio_installdir*\Commerce\schema\oracle
-   *WC_installdir*/schema/oracle

4. 3단계에서 찾은 `wcs.webservices.disable.sql` 스크립트를 텍스트 편집기에서 열고 다음 값이 `wcs.webservices.enable.sql` 스크립트에 있는 값과 동일한지 확인하십시오.





- `storeent_id` - 상점의 고유 식별자입니다.
- `OrgCode1` - 공급자 조직에 대응하는 `ORGCODE` 테이블의 항목에 대한 1차 키입니다.
- `OrgCode2` - 구매자 조직에 대응하는 `ORGCODE` 테이블의 항목에 대한 1차 키입니다.
- `MemberId` - 구매자 조직이 속한 구성원 그룹의 1차 키입니다.

5. 4단계에서 설명한 변경을 완료한 후에 해당 명령 스크립트 파일을 실행하십시오.

 DB2 데이터베이스를 사용하는 경우:

-   DB2 명령창에서 `Webservices_UndoDBUpdate.db2.bat`를 실행하십시오.
-    명령창에서 `Webservices_UndoDBUpdate.db2.sh`를 실행하십시오.
-  QSH 세션을 시작하십시오. `/QIBM/ProdData/CommerceServer55/samples/webservices/bin/ Webservices_UndoDBUpdate.db2.sh` 명령을 실행하십시오.

 Oracle 데이터베이스를 사용하는 경우:


-   DB2 명령창에서 `Webservices_UndoDBUpdate.oracle.bat`를 실행하십시오.
-   명령창에서 `Webservices_UndoDBUpdate.oracle.sh`를 실행하십시오.

로그 파일을 확인하고 실행이 완료되었는지 확인하십시오. 스크립트 파일의 실행이 완료된 후에 로그 파일의 이름 및 위치가 표시됩니다.

주: `OrderCreate` 및 `OrderStatus` 웹 서비스만 사용하지 않으려고 하거나 `OrderFulfillment` 웹 서비스를 사용하지 않는 경우 4단계에서 나열한 스크립트 파일에서 다음 SQL 문을 주석 처리하십시오.

- ```
delete from cmdreg where interfacename=
'com.ibm.commerce.webservice.utils.WSDLFinderCmd'
and classname='com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl';
```
- ```
delete from cmdreg WHERE storeent_id=
storeent_id AND interfacename='
com.ibm.commerce.order.commands.ExtOrderProcessCmd';
```

여기서 `storeent_id`는 상점의 고유 식별자입니다.

6.  17 페이지의 『견본 코드 반입』에서 반입한 견본 코드를 삭제하고 WebSphereCommerceServerExtensionsLogic 프로젝트를 다시 빌드하십시오.
7. WebSphere Commerce 시스템을 사용하는 경우, 변경사항을 적용하려면 Commerce Server 인스턴스를 다시 시작하십시오. WebSphere Commerce Studio 시스템을 사용하는 경우, WebSphere Test Environment에서 WebSphereCommerceServer 서버를 다시 시작하십시오.

제 4 장 WebSphere Commerce를 서비스 요청자로 사용

이 절에서는 WebSphere Commerce가 외부 시스템에서 정의한 웹 서비스를 호출할 수 있는 방법을 설명합니다. WebSphere Commerce가 웹 서비스를 찾고 호출할 수 있게 하는 견본을 설명합니다. 또한 OrderFulfillment 견본 웹 서비스의 구현 정보도 설명합니다.

아키텍처

이 아키텍처의 메시지 플로우는 동기화됩니다. 이것은 WebSphere Commerce가 다음 처리로 진행하기 전에 외부 시스템으로부터의 응답을 대기한다는 것을 의미합니다. 웹 서비스를 호출하여 외부 시스템과 상호작용하는 태스크 명령이 구현되었습니다.

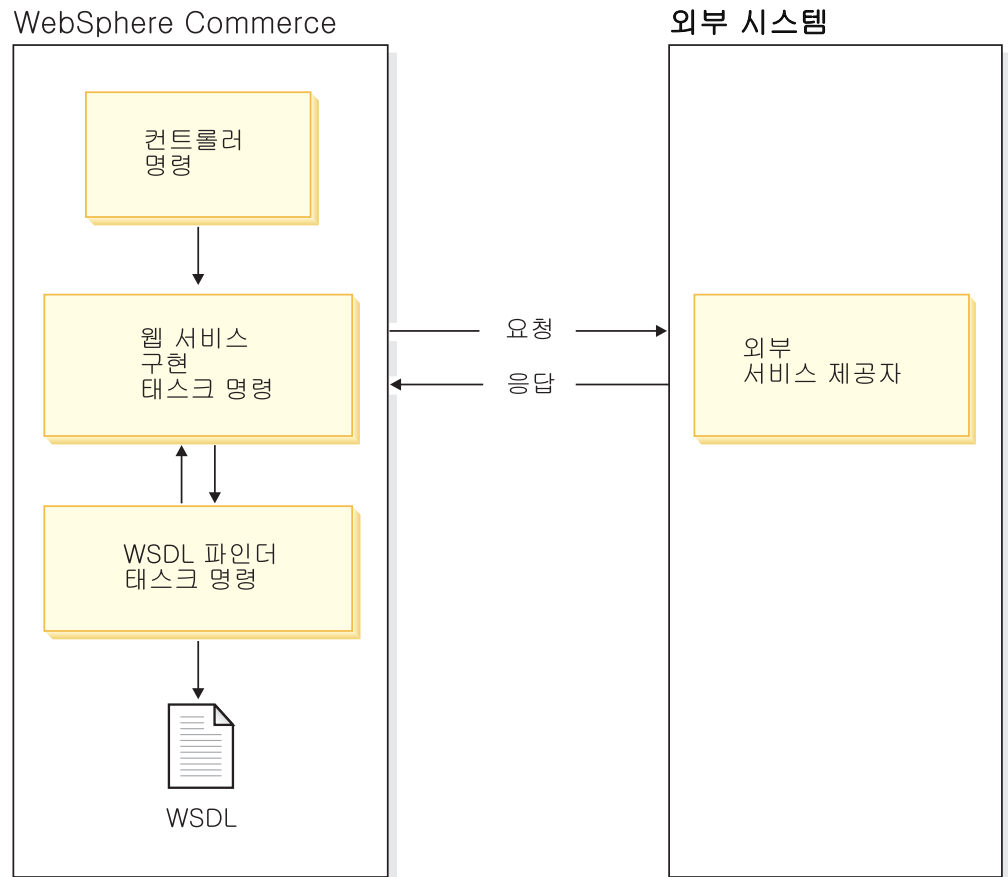


그림 5. WebSphere Commerce를 서비스 요청자로 사용하기 위한 아키텍처

WebSphere Commerce 프로그래밍 모델을 사용하여 명령을 대체함으로써 기존 비즈니스 프로세스의 작동을 사용자 정의할 수 있습니다. 컨트롤러 명령은 다수의 태스크 명

령을 호출하여 비즈니스 프로세스 실행을 완료합니다. 컨트롤러 명령 작동을 수정하는 한 방법은 컨트롤러 명령이 호출하는 태스크 명령을 새롭게 구현하는 것입니다.

컨트롤러 명령의 일부로 호출되는 태스크 명령을 찾으려면 WebSphere Commerce Production 및 Development 온라인 도움말의 "명령, 태스크 및 테이블에 대한 상호 참조" 절을 참조하십시오. 다음 목록은 WebSphere Commerce 비즈니스 프로세스가 외부 시스템에서 호스팅하는 웹 서비스를 호출할 때의 상호작용을 설명합니다.

1. 컨트롤러 명령은 등록된 태스크 명령을 호출합니다. 이 경우 웹 서비스를 호출하여 외부 시스템과 상호작용하는 태스크가 구현되었습니다. 이 태스크 명령에는 외부 시스템과 통합하는 코드가 들어 있습니다.
2. 태스크 명령은 지정된 웹 서비스의 액세스 지점을 검색합니다.
3. 태스크 명령은 서비스 액세스점 및 그 입력 매개변수를 지정하여 웹 서비스 프록시를 호출합니다.
4. 프록시는 웹 서비스를 호출하고 응답을 수신하며 이를 다시 태스크 명령에 리턴합니다.






서비스 구현 찾기

WebSphere Commerce가 외부 시스템에서 정의한 웹 서비스를 호출하게 하려면 먼저 필요한 서비스 유형에 대한 구현을 호스팅하는 서비스 제공자를 찾아야 합니다. 웹 서비스를 호출하기 위해 태스크 명령을 식별된 서비스 제공자에 지정할 수 있습니다. 보다 많은 유연성을 위해 WSDLFinderCmd 태스크 명령에 정의되었고 서비스 제공자를 찾는 태스크 명령이 포함되어 있습니다. 이 태스크 명령의 기본 구현이 다음 디렉토리에서 제공됩니다.

-  `WC_installdir\samples\webservices\com\ibm\commerce\webservice\utils`
-  `WCStudio_installdir\Commerce\samples\webservices\com\ibm\commerce\webservice\utils`
-    `WC_installdir/samples/webservices/com/ibm/commerce/webservice/utils`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/com/ibm/commerce/webservice/utils`

서비스 구현을 찾는 한 가지 방법은 서비스 레지스트리를 찾는 것입니다. 그러나 WebSphere Commerce 버전 5.5에서 제공하는 견본은 로컬 파일 시스템을 사용하여 웹 서비스의 WSDL 정의를 검색합니다. WSDLFinderCmdImpl은 `instance_name.xml` 구성 파일에서 로컬로 제공되는 등록된 WSDL 정의 파일을 찾습니다. WSDL 서비스 정의를 등록하려면 다음 디렉토리에 있는 `instance_name.xml` 구성 파일을 편집하십시오.

-  `WC_installdir\instances\instance_name.xml`

-  `WCStudio_installdir\Commerce\instances\instance_name\xml`
-    `WC_installdir/instances/instance_name/xml`
-  `/QIBM/UserData/CommerceServer55/instances/instance_name/xml`

그리고 다음을 수행하십시오.

1. SOAPWSDLFinderInfo 요소를 찾으십시오. 이 요소에는 WSDL 정의 경로 및 WSDL 파일을 등록하는 속성이 있습니다. `instance_name.xml` 파일에서 발췌한 다음 내용이 이러한 매개변수를 보여줍니다.

```
<SOAPWSDLFinderInfo
  LocalWSDLFiles="OrderFulfillment.wsdl"
  LocalWSDLPath="WC_installdir/xml/messaging"
  UDDIQueryURL=""
  UDDIPublishURL=""
  WSDLFinderImplClassName=" "
  display="false" />
```

2. LocalWSDLFiles 값을 로컬 파일 시스템에 있는 쉽표로 구분된 WSDL 정의 목록을 포함하도록 설정하고 LocalWSDLPath를 이러한 파일이 존재하는 디렉토리로 설정하십시오.
3. 이 견본은 로컬 파일 시스템을 사용하여 서비스 정의를 찾으므로 UDDIQueryURL 및 UDDIPublishURL 값을 빈 문자열로 설정하십시오.
4. WSDLFinderImplClassName 값을 빈 문자열로 설정하십시오.

동적으로 파트너를 찾고 파트너가 서비스 레지스트리에 구현되어 공개한 서비스를 호출하려면, WebSphere Commerce 버전 5.5에서 제공하는 WSDLFinderCmdImpl 견본 태스크 명령의 구현을 새로 작성하고 등록해야 합니다. 사용자 정의 구현에 해당 코드를 포함시켜 WSDL 정의를 찾으십시오. 여기에서 UDDIQueryURL 및 UDDIPublishURL 필드를 사용하여 UDDI 레지스트리를 등록할 수 있습니다. 태스크 명령 대체 방법에 관한 추가 정보는 *WebSphere Commerce 프로그래밍 안내서* 및 *학습서*를 참조하십시오.

서비스 구현 호출

컨트롤러 명령의 일부로 호출되는 태스크 명령을 찾으려면 WebSphere Commerce Production 및 Development 온라인 도움말의 "명령, 태스크 및 테이블에 대한 상호 참조" 절을 참조하십시오.

다음 테이블은 OrderProcessCmd의 일부로 실행되는 태스크를 보여줍니다.


태스크 명령 이름	설명
CheckOrderCmd	CheckOrderCmd 태스크 명령은 "제출됨" 상태의 주문이 처리되었는지 확인합니다.
DoPaymentCmd	OrderProcessCmd 컨트롤러 명령은 DoPaymentCmd 태스크 명령을 호출하여 주문에 대한 지불을 처리합니다.
ExtOrderProcessCmd	OrderProcess 명령을 완료하기 직전에 필요한 추가 처리를 수행합니다.

UpdateInventoryCmd	UpdateInventory 태스크 명령으로 항목의 재고를 갱신할 수 있습니다.
--------------------	--

이 테이블에서 보듯이 OrderProcess 명령을 사용자 정의하여 외부 서비스 센터에 이행 요청을 전송하려는 경우 ExtOrderProcessCmd 태스크 명령으로 대체할 수 있습니다. ExtOrderProcessCmd의 새 구현은 OrderFulfillment 서비스를 구현한 파트너를 찾고 해당 서비스를 호출할 수 있습니다.

WebSphere Commerce가 외부 시스템에서 호출할 수 있는 견본 웹 서비스

이 절에서는 OrderFulfillment 견본 웹 서비스를 사용하는 방법에 대해 설명합니다.

WebSphere Commerce 시스템 또는 WebSphere Commerce Studio가 설치된 개발 시스템에서 OrderFulfillment 견본을 사용할 수 있습니다. WebSphere Commerce Studio 시스템에서 이 견본을 사용하는 경우 이 절의 지시사항을 따르고 "  " 로 표시된 디렉토리 경로를 참조하십시오.

OrderFulfillment 서비스

이 절에서는 WebSphere Commerce가 외부 시스템에서 구현된 웹 서비스를 호출하는 방법을 설명합니다. OrderFulfillment 서비스는 WebSphere Commerce가 서비스 요청자로 작동되는 방법을 보여줍니다. OrderFulfillment 견본 웹 서비스를 사용하려면 WebSphere Commerce 시스템의 공개된 상점이 필요합니다.

WebSphere Commerce 환경에서 이 웹 서비스를 사용하는 경우 WebSphere Commerce Studio 환경도 또한 필요합니다.

인터페이스 WSDL 정의

OrderFulfillment 서비스의 비즈니스 로직은 OrderFulfillment 서비스인 사전 정의된 서비스 인터페이스와 함께 작동하도록 설계되었습니다. OrderFulfillment 서비스의 인터페이스 정의 WSDL 파일은 알려진 서비스에 대한 기본 정적 바인딩 및 참조용으로 로컬 파일 시스템에 있습니다.

태스크 명령

OrderProcessCmd의 일부로 호출되는 ExtOrderProcessServiceCmdImpl 명령은 외부 서비스와 상호작용하는 ExtOrderProcessServiceCmd 인터페이스를 구현합니다. 프록시를 통해 웹 서비스에 대한 요청을 시작합니다. 이 태스크 명령은 OrderFulfillment 요청을 각 서비스 센터에 전송하여 제출된 주문을 처리합니다.

다음은 OrderFulfillment 서비스를 사용하는 단계입니다.

1. 제 3 장 23 페이지의 『데이터베이스 테이블 대량 자료 반입』의 설명대로 이미 WebSphere Commerce를 서비스 제공자로 사용하는 경우, 이 단계를 실행하지 마십시오. 그러나 이전에 OrderFulfillment 명령에 대한 절을 주석 처리한 경우 이 단계를 실행해야 합니다.

WebSphere Commerce Server 명령 레지스트리에서

ExtOrderProcessServiceCmdImpl 및 WSDLFinderCmdImpl을 등록하려면 이러한 명령과 관련된 SQL 문을 실행하십시오. SQL 스크립트 파일은 다음 디렉토리에 있습니다.

▶ DB2 DB2 데이터베이스를 사용하는 경우:

- ▶ Windows WC_installdir\samples\webservices\schema\db2\ wcs.webservices.enable.sql
- ▶ Studio WCStudio_installdir\Commerce\samples \webservices\schema\db2\wcs.webservices.enable.sql
- ▶ AIX ▶ Linux ▶ Solaris WC_installdir/samples/webservices/ schema/db2/ wcs.webservices.enable.sql
- ▶ 400 /QIBM/ProdData/CommerceServer55/samples/ webservices/schema/db2/wcs.webservices.enable.sql

▶ Oracle Oracle 데이터베이스를 사용하는 경우:

- ▶ Windows WC_installdir\samples\webservices\schema\ oracle\wcs.webservices.enable.sql
- ▶ Studio WCStudio_installdir\Commerce\samples\webservices\schema\ oracle\ wcs.webservices.enable.sql
- ▶ AIX ▶ Solaris WC_installdir/samples/webservices/schema/ oracle/ wcs.webservices.enable.sql

다음은 실행해야 하는 앞의 스크립트 파일에 있는 SQL 문입니다.

```
insert into cmdreg (storeent_id, interfacename, description,
  classname, properties, lastupdate, target) values(0, 'com.ibm.commerce.
  webservice.utils.WSDLFinderCmd', 'Wsd1 finder class for webservices',
  com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl',
  null, null, 'Local');
```

```
insert into cmdreg (storeent_id, interfacename, description,
  classname, properties, lastupdate, target)values(storeent_id, '
  'com.ibm.commerce.order.commands.ExtOrderProcessCmd', '
  'webservices implementation of ExtOrderProcessCmd',
  'com.ibm.commerce.webservice.order.commands.ExtOrderProcessServiceCmdImpl',
  null,null,'Local');
```






여기서 *storeent_id*는 상점의 고유 식별자입니다.

2. OrderFulfillment 서비스를 찾으려면 다음을 수행하십시오.
 - a. SOAPWSDLFinderInfo 요소를 *instance_name.xml* 구성 파일에서 찾으십시오.

- b. LocalWSDLFiles 및 LocalWSDLPath 값이 올바르고 WSDL 정의 파일이 존재하는 디렉토리를 반영하는지 확인하십시오. SOAPWSDLFinderInfo 요소는 다음 내용과 유사해야 합니다.

```
<SOAPWSDLFinderInfo
  LocalWSDLFiles="OrderFulfillment.wsdl"
  LocalWSDLPath="WC_installdir/xml/messaging"
  WSDLFinderImplClassName=""
  display="false" />
```

- c. 다음 디렉토리에 있는 견본 OrderFulfillment.wsdl WSDL 파일을

-  *WC_installdir\samples\webservices\xml\messaging*
-  *WCStudio_installdir\Commerce\samples\webservices\xml\messaging*
-    *WC_installdir/samples/webservices/xml/messaging*
-  */QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging*

다음 디렉토리로 복사하십시오.

-  *WC_installdir\xml\messaging*
-  *WCStudio_Installdir\Commerce\xml\messaging*
-    *WC_installdir/xml/messaging*
-  */QIBM/UserData/CommerceServer55/ instances/instance_name/xml/messaging*

3. OrderFulfillment.wsdl 파일에 정의된 서비스의 서비스 제공자를 지정하십시오. OrderFulfillment.wsdl을 서비스 제공자의 정보로 갱신하십시오.

- a. 2c단계에서 복사한 OrderFulfillment.wsdl 파일을 여십시오.
- b. 서비스 정의 노드("service name"으로 시작하는 절)를 찾으십시오. 다음과 유사합니다.

```
<service name=" POCreateService ">
<port binding="tns: POCreateBinding " name=" POCreatePort">
<soap:address location="http://localhost:8080/WebservicesForWCBEWeb/
  servlet/rpcrouter"/>
</port>
</service>
```

- c. 기존 강조표시된 URL을 대상 웹 서비스의 URL로 변경하십시오.
 - d. 파일을 저장하고 닫으십시오.
4. OrderCreate 또는 OrderStatus 견본 웹 서비스를 사용하는 경우 4와 5단계를 무시하십시오. 그렇지 않은 경우 이 단계를 수행하십시오. OrderFulfillment 웹 서비스에 제공된 견본 코드를 전개하려면 WebSphere Commerce Studio가 설치 및

구성된 개발 시스템이 있어야 합니다. WebSphere Studio Application Developer가 필요한 다음 목록의 단계는 이 개발 시스템에서 수행해야 합니다.

5. 견본 웹 서비스를 편집하거나 수정하려면 견본 소스 코드를 WebSphere Commerce Studio로 반입해야 합니다. 이 코드를 반입하려면 다음을 수행하십시오.
 - a. 시작 > 프로그램 > **IBM WebSphere Commerce Studio > WebSphere Commerce** 개발 환경에서 WebSphere Commerce 개발 환경을 여십시오.
 - b. J2EE perspective로 전환하고 J2EE 탐색기 보기를 선택하십시오.
 - c. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 펼치십시오. **src** 디렉토리를 마우스 오른쪽 버튼으로 누르고 반입을 누르십시오. 반입 마법사가 열립니다.
 - d. 파일 시스템을 선택하고 다음을 누르십시오.
 - e. 찾아보기를 누르고 다음 디렉토리를 찾으십시오.
`WCStudio_installdir\Commerce\samples\webservices\com`
 - f. 모두 선택을 누른 후, 완료를 누르십시오.
 - g. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 마우스 오른쪽 버튼으로 누르고 특성을 선택하십시오.
 - h. **Java** 빌드 경로를 누른 후 라이브러리 탭으로 이동하십시오.
 - i. 변수 추가를 누르십시오. SOAPJAR 및 XERCES 변수를 선택하고 확인을 누르십시오.
 - j. 확인을 누르십시오.
 - k. **WebSphereCommerceServerExtensionsLogic** 프로젝트를 마우스 오른쪽 버튼으로 누르고 프로젝트 다시 빌드를 선택하십시오.
6. WebSphere Commerce 서버에서 견본 웹 서비스를 사용하는 경우, 이 코드를 대상 WebSphere Commerce 서버로 전개해야 합니다. 명령을 새로 전개하는 방법에 관한 추가 정보는 *WebSphere Commerce 프로그래밍 안내서* 및 *학습서를* 참조하십시오.
7. WebSphere Commerce 시스템을 사용하는 경우, 변경사항을 적용하려면 Commerce Server 인스턴스를 다시 시작하십시오. WebSphere Commerce Studio 시스템을 사용하는 경우, WebSphere Test Environment에서 WebSphereCommerceServer 서버를 다시 시작하십시오.

비즈니스 오브젝트

ExtOrderProcessServiceCmdImpl 태스크 명령은 SOAP XML 요청 메시지를 작성하기 위해 웹 서비스 프록시가 사용하는 비즈니스 오브젝트 세트를 대량 반입합니다. 이 요청 메시지는 OrderFulfillment WSDL 정의의 요소를 반영합니다.

OrderFulfillment 서비스는 OrderFulfillment WSDL 정의에서 생성된 견본 비즈니스 오브젝트 세트와 함께 제공됩니다. WebSphere Commerce가 외부 웹 서비스를

호출할 때, 53 페이지의 『부록 B. IBM WebSphere Studio Application Developer 사용』의 설명대로 IBM WebSphere Studio를 사용하거나 기타 도구를 사용하여 해당 클라이언트 프록시 및 오브젝트를 생성할 수 있습니다.

다음 테이블은 OrderFulfillment 서비스에 사용된 패키지 및 비즈니스 오브젝트를 나열합니다.

패키지 이름	오브젝트 이름	설명
com.ibm.commerce.webservice.datatype	Address	이 패키지는 여러 가지 웹 서비스 메시지에 걸쳐 사용된 모든 오브젝트로 구성됩니다.
	ContactInfo	
	ContactPersonName	
	ControlArea	
	Credentials	
	Email	
	InvoiceInfo	
	ProductDimension	
	ProductMeasurement	
	ProductWeight	
	RequisitionerID	
	Telephone	
	UserData	
UserDataField		

패키지 이름	오브젝트 이름	설명
com.ibm.commerce.webservice.datatype.orderfulfillment	BillToInfo	이 패키지는 OrderFulfillment XML 메시지에 특정된 데이터가 있는 모든 오브젝트를 포함합니다.
	BuyOrgAccountingDetail	
	BuyOrgInfo	Report_NC_PurchaseOrder는 응답으로 수신된 OrderFulfillmentStatus 오브젝트 및 프록시에 전송된 상위 랩퍼 오브젝트입니다.
	DataArea	
	DateTimeReference	
	ItemShippingSchedule	
	ItemUnitPrice	
	MerchantInfo	
	MonetaryAmount	
	OrderFulfillmentStatus	
	PCardInfo	
	Report_NC_PurchaseOrder	
	ReportPO	
	ReportPOHeader	
	ReportPOItem	
	RequisitionerInfo	
	ServiceAllowanceCharge	
	ShipDateReference	
	ShippingCarrierInfo	
	ShipToInfo	
TaxInfo		
TotalPriceInfo		







견본 테스트

WebSphere Commerce의 공개된 상점에서 주문을하면 OrderFulfillment 웹 서비스가 호출됩니다. 웹 서비스는 외부 서비스 제공자에 연결되어 서비스 센터를 통해 주문을 제출합니다. 서비스 제공자가 제출된 주문의 ID를 리턴하면 WebSphere Commerce 데이터베이스의 ORDERS 테이블의 ORMORDER 열의 OrderID가 갱신됩니다.

OrderFulfillment 웹 서비스 사용 안함

이 절에서는 WebSphere Commerce에서 제공하는 기본 OrderFulfillment 웹 서비스를 사용하지 않는 방법을 설명합니다.

1. OrderFulfillment 웹 서비스를 사용하기 위해 전개한 webservices.jar JAR 파일을 삭제하십시오.
2. 다음 디렉토리에서 OrderFulfillment.wsdl 파일을 삭제하십시오.

-  `WC_installdir\xml\messaging`
-  `WCStudio_installdir\Commerce\xml\messaging`
-    `WC_installdir/xml/messaging`
-  `/QIBM/UserData/CommerceServer55/instances/instance_name/xml/messaging`

3. OrderFulfillment.wsdl 파일에 대한 참조를 `instance_name.xml` 구성 파일에서 제거하십시오.
4. OrderCreate 및 OrderStatus 웹 서비스를 사용하지 않으려고 다음 SQL 문을 주석 처리 했거나 OrderCreate 및 OrderStatus 웹 서비스를 사용하지 않는 SQL 스크립트를 실행하지 않은 경우, 다음 SQL 문을 실행하십시오.







```
delete from cmdreg where interfacename='com.ibm.commerce.webservice.utils.WSDLFinderCmd' and classname='com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl';
```

```
delete from cmdreg where storeent_id= storeent_id and interfacename='com.ibm.commerce.order.commands.ExtOrderProcessCmd';
```





여기서 `storeent_id`는 상점의 고유 식별자입니다.

앞의 SQL 문은 다음 스크립트 파일에 있습니다.


 DB2 데이터베이스를 사용하는 경우:

-   DB2 명령창에서 `Webservices_UndoDBUpdate.db2.bat`를 실행하십시오.
-    명령창에서 `Webservices_UndoDBUpdate.db2.sh`를 실행하십시오.
-  QSH 세션을 시작하십시오. `Webservices_UndoDBUpdate.db2.sh` 명령을 실행하십시오.

 Oracle 데이터베이스를 사용하는 경우:

-   DB2 명령창에서 `Webservices_UndoDBUpdate.oracle.bat`를 실행하십시오.
-   명령창에서 `Webservices_UndoDBUpdate.oracle.sh`를 실행하십시오.

로그 파일을 확인하고 실행이 완료되었는지 확인하십시오. 스크립트 파일의 실행이 완료된 후에 로그 파일의 이름 및 위치가 표시됩니다.

5.  17 페이지의 『견본 코드 반입』에서 반입한 견본 코드를 삭제하고 WebSphereCommerceServerExtensionsLogic 프로젝트를 다시 빌드하십시오.
6. WebSphere Commerce 시스템을 사용하는 경우, 변경사항을 적용하려면 Commerce Server 인스턴스를 다시 시작하십시오. WebSphere Commerce Studio 시스템을 사용하는 경우, WebSphere Test Environment에서 WebSphereCommerceServer 서버를 다시 시작하십시오.

부록 A. 견본 WSDL 정의 파일

WebSphere Commerce 버전 5.5에서 제공하는 견본 웹 서비스의 WSDL 정의는 다음 디렉토리에 있습니다.

-  `WC_installdir\samples\webservices\xml\messaging`
 -  `WCStudio_installdir\Commerce\samples\webservices\xml\messaging`
-    `WC_installdir/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/ samples/webservices/xml/messaging`

웹 서비스 이름	파일 이름
OrderFulfillment	OrderFulfillment.wsdl
OrderCreate	OrderCreate.wsdl
OrderStatus	OrderStatus.wsdl

부록 B. IBM WebSphere Studio Application Developer 사용

다양한 도구를 사용하여 웹 서비스 코드를 생성할 수 있습니다. 이 웹 서비스 구현과 함께 제공되는 견본 코드는 IBM WebSphere Studio Application Developer를 사용하여 생성되었습니다. 이 절의 정보는 IBM WebSphere Studio Application Developer 버전 5.5에서 테스트되었습니다. WebSphere Commerce가 없거나 WebSphere Commerce Studio를 사용하지는 않지만 WebSphere Commerce에서 제공하는 웹 서비스를 액세스하기 위한 웹 서비스 클라이언트를 개발하려면 웹 서비스 클라이언트 코드를 생성할 수 있는 임의의 도구를 사용할 수 있습니다.

이 부록에서는 WebSphere Commerce에서 제공하는 서비스의 테스트 클라이언트를 생성하기 위해 WebSphere Studio를 사용하는 방법에 대해 설명합니다. 여기에는 WSDL 정의 파일을 WebSphere Studio로 반입하고 테스트 클라이언트를 실행하며 TCP 모니터 서버 인스턴스를 작성 및 구성하고 TCP 모니터를 사용하여 SOAP 메시지 추적하는 것이 포함됩니다.

또한 WSTK(Web services Toolkit)를 사용하여 클라이언트 코드 및 데이터 유형을 생성할 수도 있습니다. 웹 서비스 툴킷에 관한 추가 정보는 <http://www.alphaworks.ibm.com/tech/webservices toolkit>을 참조하십시오. WSTK는 웹 서비스에서 대두되는 기술을 소개하고 최신 웹 서비스 스펙에 대한 이해를 돕습니다. 그러나, 웹 서비스의 제품 레벨 개발 환경의 경우 IBM WebSphere Studio Application Developer를 권장합니다.

또는 Apache SOAP 구현을 클라이언트로 사용하여 수동으로 클라이언트 코드 및 데이터 유형을 작성할 수 있습니다. 자세한 정보는 Apache SOAP 문서를 참조하십시오.

이 절에서는 WebSphere Studio Application Developer에 대한 지식이 있다고 가정합니다. WebSphere Studio를 사용한 웹 서비스 및 엔터프라이즈 응용프로그램 개발에 익숙하지 않은 경우, 먼저 IBM developerworks 사이트(<http://www.ibm.com/developerworks/webservices>)에서 제공하는 WebSphere Studio 문서 또는 학습서를 숙지해야 합니다.







WSDL을 WebSphere Studio Application Developer로 반입

WSDL을 WebSphere Studio로 반입하기 전에 웹 응용프로그램이 작성되었고 이 웹 응용프로그램이 속한 엔터프라이즈 응용프로그램도 작성되었는지 확인하십시오. 웹 응용프로그램이 존재하지 않으면 지시사항을 따르기 전에 웹 응용프로그램을 작성하십시오. WebSphere Commerce 개발 환경을 설정했으면, 새로운 엔터프라이즈 응용프로그램

램 및 웹 응용프로그램 램을 작성하지 마십시오. 그 대신

WebSphereCommerceServerExtensionsLogic 프로젝트를 사용하십시오.

다음은 IBM WebSphere Studio를 사용하여 클라이언트 코드 및 데이터 유형을 생성하는 프로시저의 개요입니다.

1. WSDL 정의를 반입할 새 폴더(예: wsdl)를 웹 프로젝트에서 작성하십시오. 웹 프로젝트가 없는 경우 작성하십시오.
2. wsdl 폴더를 오른쪽 마우스 버튼으로 누르십시오.
 - a. 반입 > 파일 시스템을 선택하고 다음을 누르십시오.
 - b. 찾아보기를 누르고 WSDL 정의 파일이 저장된 폴더를 찾으십시오.
 - c. 확인을 누르십시오. 다음 디렉토리에 WSDL 정의가 있습니다.
 -  WC_installdir\samples\webservices\xml\messaging
 -  WCStudio_installdir\Commerce\samples\webservices\xml\messaging
 -    WC_installdir/samples/webservices/xml/messaging
 -  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging
3. 오른쪽 패널에서 WebSphere Studio로 반입할 WSDL을 선택하고 완료를 누르십시오.

클라이언트 코드 및 데이터 유형 생성

WSDL 정의를 WebSphere Studio로 반입한 후, 웹 서비스를 테스트하기 위해 클라이언트 코드 및 견본 테스트 클라이언트를 생성할 수 있습니다. 클라이언트 프록시는 Java 클라이언트에서 웹 서비스에 액세스하는데 사용할 수 있는 Java 클래스입니다. 견본 테스트 클라이언트는 WebSphere Studio에서 웹 서비스를 테스트할 수 있는 견본 JSP를 제공합니다. 다음을 수행하여 클라이언트 프록시 및 견본 테스트 클라이언트를 생성하십시오.

주: 다음의 지시사항에 따라 WebSphere Commerce에서 외부 클라이언트가 호스팅하는 웹 서비스에 액세스할 수 있는 클라이언트 코드를 생성할 수 있습니다. 유일한 차이점은 서비스 제공자에서 외부 서비스 구현 정의를 얻어서 시작하고 WSDL 정의를 WebSphere Studio로 반입해야 한다는 것입니다.

1. 웹 perspective로 이동하여 생성된 클라이언트 코드를 저장할 웹 프로젝트를 선택하십시오. 웹 프로젝트가 없는 경우 작성하십시오.
2. WSDL 정의 파일을 선택한 후 오른쪽 마우스 버튼으로 눌러 클라이언트 코드를 생성하십시오. 메뉴에서 새로 만들기 > 기타를 선택하십시오. 새로 만들기 마법사가 표시됩니다.

3. 왼쪽 패널에서 웹 서비스를 오른쪽 패널에서 웹 서비스 클라이언트를 선택하십시오. 다음을 누르십시오.
4. 웹 서비스 클라이언트 대화 상자에서 **Java** 프록시를 선택하고 다음을 누르십시오.
5. 웹 서비스 WSDL 파일 선택 대화 상자에서 선택된 WSDL 파일이 선택되었는지 확인하고 다음을 누르십시오.
6. 웹 서비스 바인딩 프록시 생성 대화 상자에서 클래스 이름을 기록하고 다음을 누르십시오.
7. 생성된 프록시 테스트를 선택하십시오. 테스트 기능 값이 웹 서비스 견본 JSP 파일로 설정되었는지 확인하십시오. 완료를 누르십시오.
8. 웹 perspective로 전환하여 견본 폴더를 펼치십시오. 테스트 JSP 클라이언트가 표시됩니다.

웹 서비스 테스트

이 절에서는 앞 절에서 생성한 견본 테스트 클라이언트를 사용하여 WebSphere Commerce가 제공하는 웹 서비스를 테스트하는 방법에 대해 설명합니다. 다음을 수행하여 웹 서비스를 테스트하십시오.

1. 최소한 한 개의 WebSphere Test Environment 서버가 존재하는지 확인하십시오. 서버가 없으면 다음을 수행하여 서버를 작성하십시오.
 - a. 서버 perspective로 이동하십시오.
 - b. 파일 > 새로 만들기 > 서버 및 서버 구성을 선택하십시오.
 - c. 서버 이름을 입력하고 유형을 **WebSphere 버전 5.0 > Test Environment**로 선택하십시오.
2. TestClient.jsp를 오른쪽 마우스 버튼으로 누르고 서버에서 실행을 선택하십시오. 테스트 클라이언트가 브라우저 창에 나타납니다.
3. 왼쪽 프레임에서 **getEndPoint()** 메소드를 선택하고 호출을 누르십시오. 결과 프레임에서 리턴된 끝점이 연결할 서버의 URL인지 확인하십시오.
4. 리턴된 끝점이 WebSphere Commerce Server가 실행되는 서버와 일치하지 않으면 WebSphere Commerce Server를 지시하도록 URL을 설정하십시오.
 - a. 왼쪽 프레임에서 **setEndPoint()** 메소드를 선택하십시오.
 - b. 오른쪽 프레임에서 WebSphere Commerce Server URL을 갱신하고 호출을 누르십시오.
5. 클라이언트 프록시의 비즈니스 메소드를 선택하십시오. OrderCreate 프록시의 메소드는 orderCreate입니다.
 - a. 양식으로 전달할 매개변수 정보를 입력하십시오. 오른쪽 프레임에 표시됩니다.
 - b. 모든 필수 필드에 대한 값을 입력하십시오.

- c. 호출을 누르십시오. SOAP 메시지가 작성되어 종료지점으로 지정된 서버로 전송됩니다.

결과 프레임에서 웹 서비스 호출 결과를 볼 수 있습니다.

TCP 모니터를 사용하여 SOAP 메시지 추적

이 절에서는 TCP/IP 모니터 서버 인스턴스를 작성 및 구성하는 방법과 TCP/IP 모니터 서버를 사용하여 SOAP 메시지를 추적하는 방법에 대해 설명합니다. 이는 웹 서비스를 디버깅하거나 개발 중인 웹 서비스의 맵핑 템플릿을 정의할 때 유용합니다.

TCP/IP 모니터는 웹 브라우저와 응용프로그램 서버 간 모든 요청 및 응답을 모니터링하는 단순 서버입니다. 클라이언트와 응용프로그램 서버 간 모든 트래픽을 캡처하고 표시할 수 있습니다.

여기에서는 새 TCP/IP 모니터 서버 인스턴스를 작성하고 다음을 수행해야 합니다.

- TCP/IP 모니터 서버를 구성하여 요청을 WebSphere Commerce Server로 전송하십시오.
- SOAP 요청과 응답을 인터셉트할 수 있도록 테스트 클라이언트를 구성하여 TCP/IP 모니터를 지정하십시오.

TCP 모니터 서버 인스턴스 작성

TCP/IP 모니터 서버 인스턴스인 TCPMon을 작성하려면 다음을 수행하십시오.

1. 기본 메뉴에서 파일 > 새로 만들기 > 기타를 선택하십시오.
2. 왼쪽 패널에서 서버를 오른쪽 패널에서 서버 및 서버 구성을 선택하고 다음을 누르십시오.
3. 새 서버 작성 및 서버 구성 창에서 다음을 수행하십시오.
 - a. 서버 이름(예: TCPMon)을 입력하십시오.
 - b. 서버 유형(TCP/IP 모니터링 서버)을 선택하십시오.
 - c. 다음을 누르십시오.
4. 새 서버 작성 및 서버 구성 창에서 다음을 수행하십시오.
 - a. 원격 호스트 필드에 WebSphere Commerce Server 호스트 이름을 입력하십시오.
 - b. 원격 포트 필드에 WebSphere Commerce Server가 청취하는 포트 번호를 입력하십시오. WebSphere Commerce가 클라이언트와 동일한 시스템에서 실행되는 경우 localhost를 호스트 이름으로 사용할 수 있습니다.
 - c. 완료를 누르십시오.

TCP 모니터 서버 인스턴스 구성

WebSphere Commerce Server의 리스너 포트가 TCP 모니터의 원격 포트가 되도록 TCP 모니터를 구성하십시오. 다음을 수행하십시오.

1. 서버 perspective로 이동하십시오.
2. 서버 구성 보기에서 TCP 모니터 인스턴스를 두 번 누르십시오. 다음 그림에서와 같이 TCP 모니터의 서버 구성이 편집기 보기에서 열립니다.

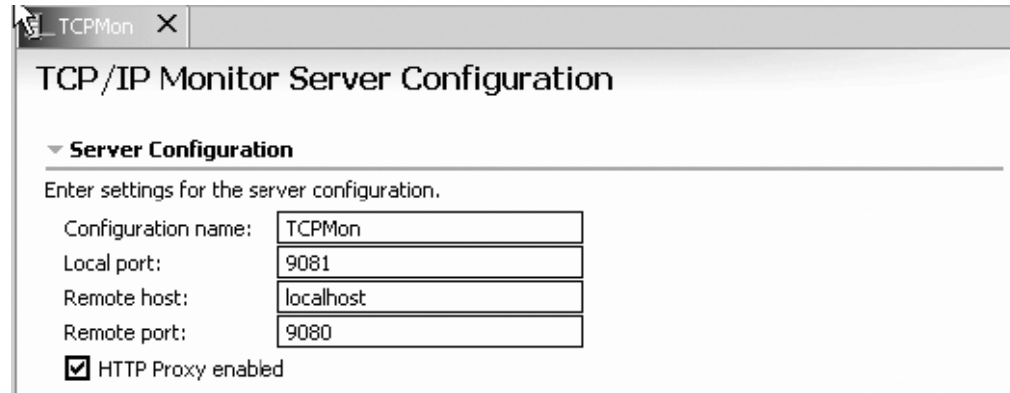


그림 6. TCP/IP 모니터 서버 구성 구성

여기서, 로컬 포트는 TCP 모니터가 청취 중인 포트이며 원격 포트는 WebSphere Commerce Server가 청취 중인 포트입니다.

TCP/IP 모니터를 사용한 SOAP 메시지 추적

TCP/IP 모니터를 구성한 후에 TCP/IP 모니터를 사용하여 SOAP 메시지를 추적하십시오. 이를 수행하려면 서버에서 생성된 견본 테스트 클라이언트를 시작하고 이를 구성하여 TCP/IP 모니터로 요청을 전송하십시오. 다음을 수행하십시오.

1. TCP/IP 모니터 및 WebSphere Commerce Server가 실행 중인지 확인하십시오.
 - a. WebSphere Commerce Server가 WebSphere Studio 외부에서 실행되는 경우, WebSphere Application Server 관리 콘솔을 사용하여 서버를 시작하십시오.
 - b. WebSphere Commerce Server가 WebSphere Commerce Studio에서 실행되는 경우, 서버 perspective로 이동하십시오. 서버 보기를 여십시오. WebSphereCommerceServer Server의 상태가 중지됨인 경우 서버를 오른쪽 마우스 버튼으로 누르고 시작을 선택하십시오. 서버 보기에서 상태가 시작됨으로 변경될 때까지 기다리십시오.
2. 웹 perspective로 이동하십시오. sample\TestClient.jsp를 찾아보십시오. sample\TestClient.jsp를 오른쪽 마우스 버튼으로 누르고 서버에서 실행을 선택하십시오.
3. 서버 선택 창에서 JSP 파일이 실행되어야 하는 서버가 선택되었는지 확인하십시오. 완료를 누르십시오.

4. 서버 클라이언트 선택이 프롬프트된 경우 웹 브라우저를 선택하고 완료를 누르십시오. 다음 그림에서처럼 새 브라우저 창이 세개의 프레임으로 함께 열립니다.

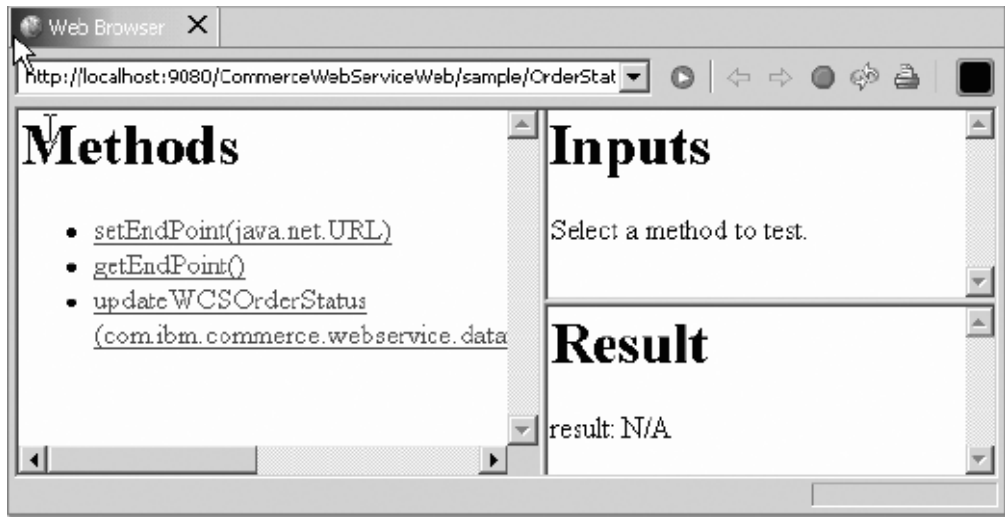


그림 7. 웹 브라우저 선택

- a. 다음 그림에서처럼 Methods 프레임에서 **getEndPoint()** 메소드를 선택하십시오.

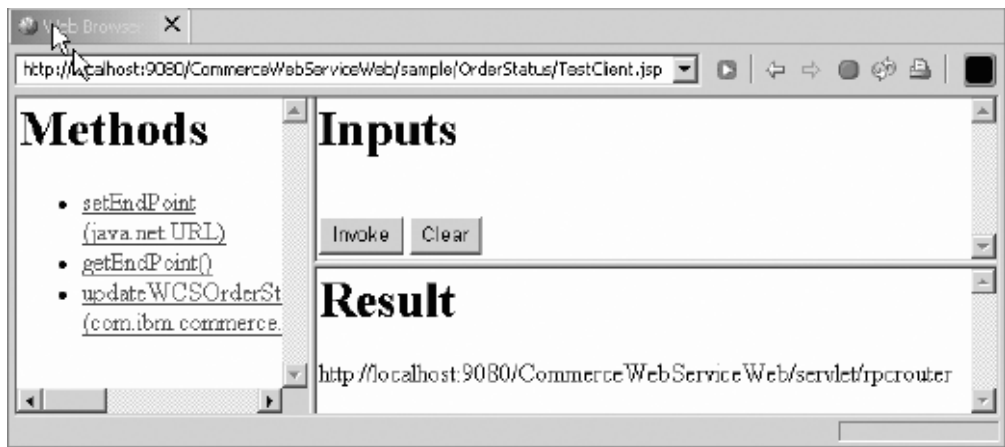


그림 8. getEndPoint() 메소드 선택

- b. Inputs 프레임에서 **Invoke**를 누르십시오. 결과가 Result 프레임에 표시됩니다.
5. TCP 모니터로 요청을 전송하기 위한 테스트 클라이언트를 구성하려면 다음을 수행해야 합니다.
 - a. Methods 프레임에서 **setEndPoint()** 메소드를 선택하십시오.
 - b. Inputs 프레임에 TCP 모니터의 **url**을 입력하십시오. 예를 들어, `http://localhost:9080/webapp/wcs/stores/servlet/`이고 여기서 9080은 TCP 모니

터가 실행되는 포트이고 webapp/wcs/stores/servlet/은 WebSphere Commerce 요청 Servlet이 있는 웹 경로입니다.

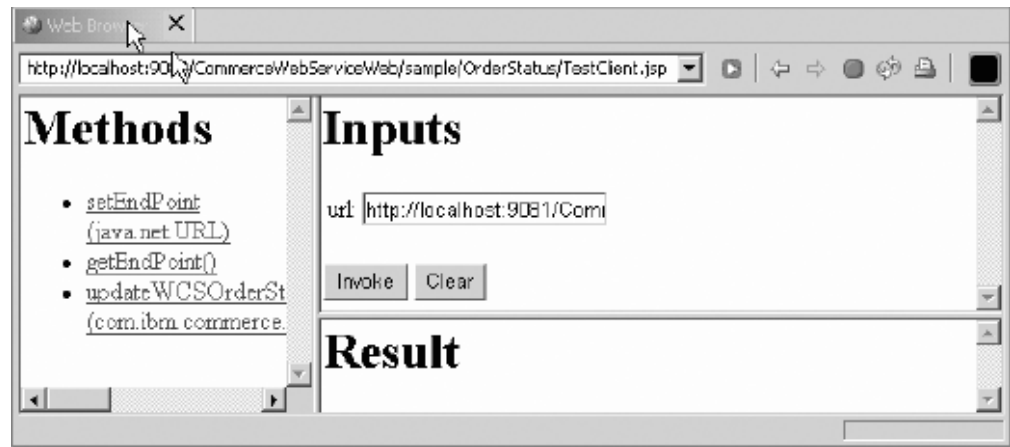


그림 9. *setEndPoint()* 메소드 선택

- c. **Invoke**를 누르십시오.
6. **getEndPoint()** 메소드를 다시 호출하여 종료 지점을 확인하십시오.
7. 비즈니스 로직을 호출하려면 다음을 수행하십시오.
 - a. 메소드 프레임에서 비즈니스 로직 메소드(예: **updateWCSOrderStatus()**)를 누르십시오.
 - b. 다음 그림에서처럼 Inputs 프레임에 모든 필수 매개변수의 값을 입력하십시오.

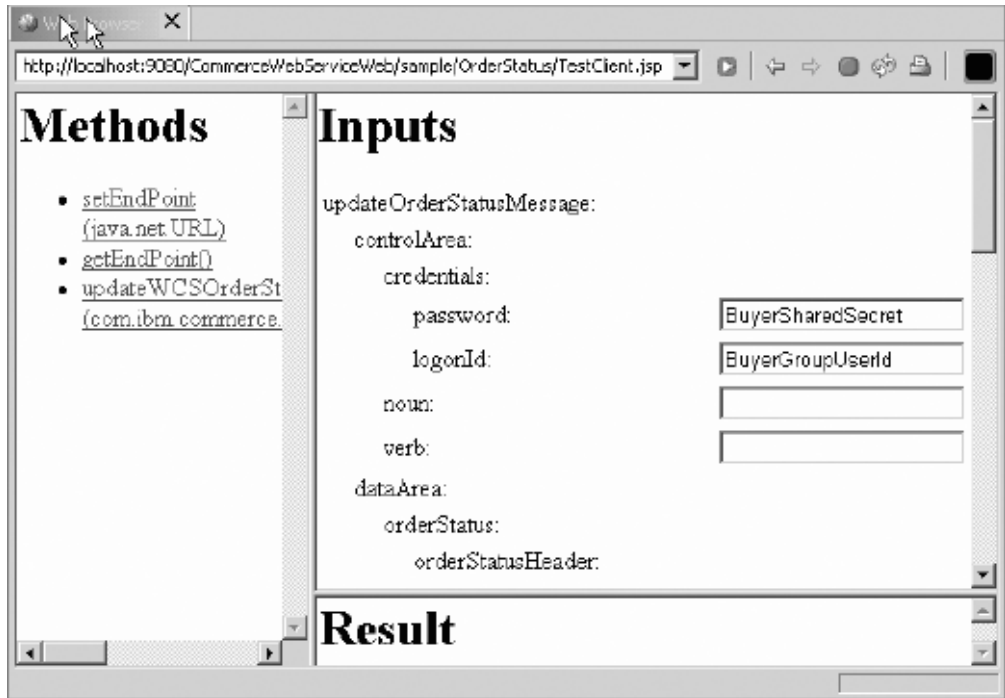


그림 10. 비즈니스 로직 선택

- c. **Invoke**를 누르십시오.
- d. 결과가 **Result** 프레임에 표시됩니다.
- e. TCP/IP 모니터 보기로 이동하여 최신 메시지를 선택하십시오. 다음 그림에서처럼 SOAP 요청 및 응답이 표시됩니다.

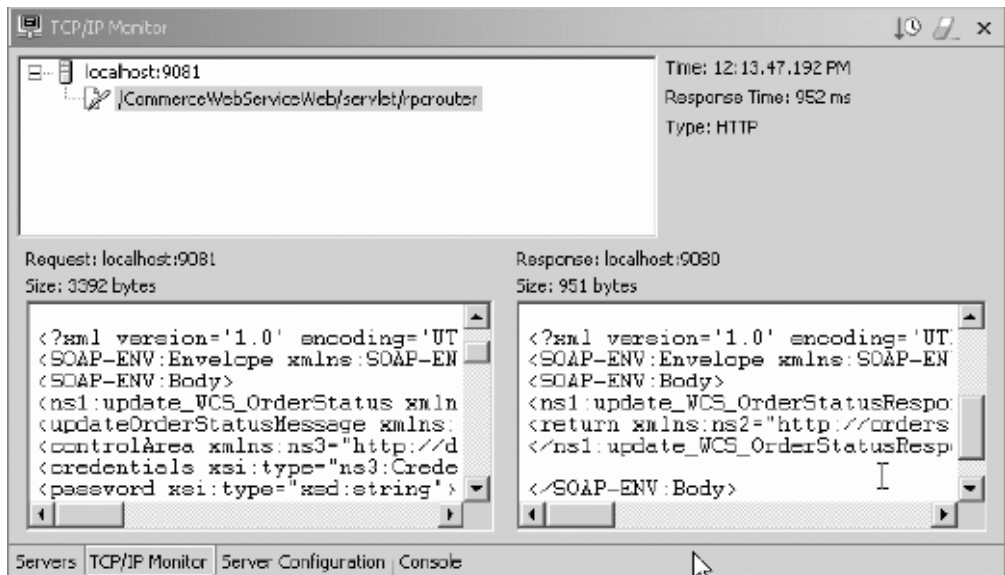


그림 11. SOAP 요청 및 응답 표시

클라이언트 코드 사용자 정의

클라이언트 코드를 생성할 때, 생성된 기본 클라이언트 코드가 항상 WebSphere Commerce에서 작동하는 것은 아니라는 점을 고려해야 합니다. 예를 들어, 기본 BeanSerializer 클래스를 사용하여 데이터 유형을 직렬화하는 경우, 일부 데이터 유형은 WebSphere Commerce가 서버 측에서 이해할 수 없는 XML 문자열로 변환됩니다. 예를 들어, OrderCreate 견본 웹 서비스는 이러한 데이터 구조를 가지고 있습니다. 이 데이터 구조는 WebSphere Commerce 명령에 대한 인터페이스로 제공되면 UserData 데이터 구조로 불립니다. UserData 데이터 구조를 사용하여 메시지 중심 인터페이스를 통해 명령으로 전달할 수 있습니다.

다음 절에서는 WebSphere Commerce가 사용할 수 있도록 UserData 데이터 구조를 직렬화하는 방법을 설명합니다.

주: 제공된 UserData 데이터 구조는 견본이며 OrderCreate 웹 서비스에 대해 선택적입니다.

WebSphere Commerce를 사용하여 외부 시스템은 사용자 정의 이름 값 쌍을 전송된 XML 메시지의 일부로 포함시킬 수 있습니다. 이를 지원하기 위해 USERDATA라는 XPATH 유형을 사용하여 UserData 요소를 포함하는 다음과 같은 견본 XML 메시지에 설명된 대로 처리합니다.

```
<UserData>
  <UserDataField name="abc">xyz</UserDataField>
</UserData>
```

이 UserData 요소의 XPATH 정의는 다음과 같습니다.

```
<Tag XPath='.../UserData/UserDataField'
  XPathType='USERDATA' />
```

XPATH 프로세서는 UserData 요소를 이름 값 쌍 "abc=xyz"로 변환합니다.

다음과 유사한 데이터 구조가 있는 경우 Apache SOAP 구현에서 제공되는 기본 bean 직렬 변환기에 문제가 있을 수 있습니다. Apache SOAP 구현은 데이터 구조의 모든 데이터 유형을 XML 문서의 개별 요소로 변환합니다. Apache SOAP 클라이언트를 사용하여 Java 오브젝트를 SOAP XML 메시지로 변환하는 경우, WebSphere Commerce는 이 XML의 구문을 분석할 수 없고 따라서 UserData 오브젝트를 작성할 수 없습니다.

```
class UserData
(
  private String userDataField = null;
  private String name = null;
  // getter and setter methods here ....
)
```

이 클래스는 XML 구조로 직렬화됩니다.

```

<UserData>
  <UserDataField>xyz</UserDataField>
  <name>abc</name>
</UserData>

```

결과적으로 WebSphere Commerce 메시지 매퍼는 이것을 USERDATA 구조로 인식하지 못하게 됩니다.

필요한 형식으로의 UserData 데이터 유형 직렬화를 지원하려면 사용자 정의 직렬 변환기를 작성하고 이를 SOAP 맵핑 레지스트리로 등록해야 합니다. 사용자 정의 직렬 변환기는 다음 marshall 메소드를 가지고 있어야 합니다.

```

/*
 * serialize the com.ibm.commerce.webservices.datatype.UserData
 * to the corresponding XML schema as expected by WC.
 *
 * @param String inScopeEncStyle
 * @param Class javaType
 * @param Object src
 * @param Object context
 * @param Writer sink
 * @param NSStack nsStack
 * @param XMLJavaMappingRegistry xjmr
 * @param SOAPContext ctx
 * @exception IllegalArgumentException
 * @exception IOException
 */
public void marshall(String inScopeEncStyle,
                    Class javaType,
                    Object src,
                    Object context,
                    Writer sink,
                    NSStack nsStack,
                    XMLJavaMappingRegistry xjmr,
                    SOAPContext ctx)
    throws IllegalArgumentException, IOException
{
    if (!javaType.equals(UserData.class))
    {
        throw new IllegalArgumentException
("Can only serialize UserData instances");
    }
    nsStack.pushScope();
    if (src != null)
    {
        SoapEncUtils.generateStructureHeader(inScopeEncStyle,
                                           javaType,
                                           context,
                                           sink,
                                           nsStack,
                                           xjmr);

        UserData data = (UserData)src;
        sink.write("<UserDataField name=\"" + data.getName() + "\">");
        sink.write(data.getValue() +
"</UserDataField>");
    }
}

```



```
        sink.write("</" + context + '>');
    }
else
    {
        SoapEncUtils.generateNullStructure(inScopeEncStyle,
                                           javaType,
                                           context,
                                           sink,
                                           nsStack,xjmr);
    }
    nsStack.popScope();
}
```

주: WebSphere Commerce 버전 5.5에서 제공하는 견본 서비스를 앞의 코드와 작동하도록 설계되어 있습니다.

주의사항

이 정보는 미국에서 제공되는 제품 및 서비스를 대상으로 구성했습니다. IBM은 다른 국가에서 이 문서에 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 재산권을 침해하지 않는 한, 기능상 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 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것은 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-chome, Minato-ku

Tokyo 106-0032, 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.

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

(1) 독립적으로 작성된 프로그램 및 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및 (2) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 라이선스 사용자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 조항 및 조건(예를 들어, 사용료 지불 등)에 따라 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.

이 책은 다른 회사의 인터넷 사이트에 대한 참조를 포함하여 이러한 회사의 제품에 관한 정보를 포함할 수 있습니다. IBM은 이러한 정보의 정확성, 완결성 또는 사용에 대해 책임지지 않습니다.

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 기능성 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 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다.

상표

IBM, and the IBM 로고는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록상표입니다.

다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표 또는 등록상표입니다.

AS/400
AIX
DB2
DB2 Universal Database™
@server
IBM

iSeries
MQSeries®
S/390
zSeries
WebSphere

Microsoft®, Windows 및 Windows NT®는 Microsoft Corporation의 상표 또는 등록상표입니다.

Java 및 모든 Java 기반 상표는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

기타 회사, 제품 또는 서비스 이름은 다른 회사의 상표이거나 서비스표입니다.

IBM