

전자상거래시스템을 위한 EJB 컴포넌트 기반의 워크플로우 설계

정화영

예원대학교 정보경영학부

A Design of Workflow based on EJB Component for E-Business System

Hwa-Young Jeong

Yewon University

E-mail : jmichael@hanmir.com

요 약

전자상거래 분야는 다양한 소비자의 욕구를 지원하기 위한 콘텐츠를 기반으로 활성화되고 있다. 즉, 효율적인 비즈니스 트랜잭션 처리를 위한 기법들이 연구되고 있으며 EJB와 같은 웹 기반 컴포넌트 기술이 응용되고 있다.

따라서, 본 연구는 웹 기반 EJB 컴포넌트 기술을 응용한 효율적인 전자상거래 콘텐츠 관리 및 처리를 위한 워크플로우를 설계하였다. 이에 따라, 다양한 콘텐츠의 핸들링을 위하여 웹 서버 컴포넌트 모델인 EJB를 사용하였으며, 기능단위의 컴포넌트들이 다양한 콘텐츠를 처리함으로써 재사용, 유지보수등의 시스템 개발 효율성을 높일 수 있다.

I. 서 론

디지털기술을 기반으로 한 산업계의 파급은 디지털경제를 형성하게되었으며, 산업정보화의 진전, 산업간·지역 간의 통합화진전, 그리고 소프트웨어, 서비스화의 가속 등으로 전개되었다. 이에 따라, 상품의 디지털화가 가장 보편적인 현상으로서, 인터넷을 기반으로 전자상거래(EC : Electronic Commerce)의 구성내용물인 콘텐츠에 대한 디지털서비스가 급속도로 확산되었다. 또한, 전자상거래형태가 디지털기술을 기반으로 하여 국가와 기업간의 거래(A2B : Authority to Business), 기업 간 거래(B2B : Business to Business), 기업과 개인간의 거래(B2C : Business to Customer)의 다양한 형태로 진전되었다[1]. 즉, 이들 여러 형태의 기술들은 비즈니스측면에서의 전자상거래와 더불어 인터넷에서 제공된 정보와 서비스를 확대하여 고객에게 보다 많은 기회를 부여하도록 한 것이다.

이에 따라, 전자화된 시장을 지원하는 시스템 컴포넌트들을 인터넷기반으로 통합하기 위한 기술적인 표준을 제시되었다. 그리하여, 변화하는 다양한 이질적인 시스템간의 상호 운영성 및 증가되는 트래픽과 트랜잭션 요구를 만족시킬 수 있는 확장성과 유연성을 갖는 시스템 아키텍처가 필요하였다. 따라서, 잘 정의된 인터페이스에 기반한 블랙박스 부품들의 조립에 의한 컴포넌트 기반 소프트웨어 개발(CBD : Component Based Development)방법은 응용시스템의 개발에 응용

성, 생산성, 유지보수성, 유연성, 재사용성등을 제공함으로써 전자상거래 시스템 구축의 새로운 대안으로 제시되고 있다[2]. 이는, 컴포넌트를 부품처럼 조립하고 기능이 개선된 부품을 재조립하는 재사용방법을 사용함으로써[3], 보다 유연하고 효율적인 전자상거래 시스템 구축이 가능하게 된 것이다.

현재의 컴포넌트 기반은 COM+, CORBA, EJB 등으로 대표된다. 특히, EJB는 인터넷 환경 속에서 애플리케이션을 안정적으로 구축할 수 있도록 도와주는 서버측 컴포넌트에 대한 표준모델로써, 기존의 개발자들이 부담해야했던 트랜잭션, 동시성제어, 지속성, 보안등의 처리를 EJB 컨테이너가 담당하여준다. 그러므로, 개발자들은 비즈니스 로직에만 전념할 수 있게 해 주는 장점이 있다[5, 6].

이러한 배경으로, 본 논문에서는 다중서버 환경에서 서버 컴포넌트 모델인 EJB를 통하여 컴포넌트 기반의 효율적인 전자상거래 구현 기법을 살펴보고자 한다.

II. 관련연구

2.1 전자상거래 시스템 아키텍처

기존에 구축된 전자상거래 시스템들은 구현 아키텍처에 따라 다음 3가지로 분류할 수 있다[2].

첫째, Client/Server 아키텍처는 데이터베이스 서버와 데이터를 처리하고 표현하는 응용한정적인 큰 규모의 클라이언트 프로그램이 요구된다. 따라서, 데이터베이스 부하증가에 대한 확장과 유지보수가 어렵다.

둘째, N-Tier 아키텍처는 클라이언트와 데이터베이스 서버 사이에 응용서버가 존재한다. 응용서버는 비즈니스로직을 구현하고 클라이언트는 이를 표현하도록 되어있다. 그러나, 일반적으로 응용서버 하나만 갖음으로써 신뢰성에 대한 문제를 야기시킨다.

셋째, 웹 기반 아키텍처는 웹 브라우저만 클라이언트에 배치하고 비즈니스 로직구현은 시스템 독립적인 모듈의 조립으로 구성한다.

이는, <그림 1>과 같이 웹 서버만이 클라이언트의 요구에 응함으로써 웹 서버의 부하를 줄여주고 단일 인터페이스 제공 및 개별로직의 구현이 가능하다.

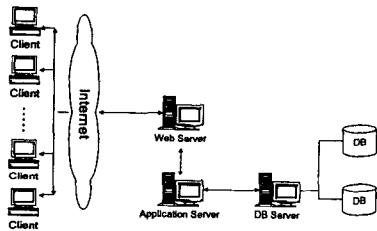


그림 1. 웹 기반 아키텍처

2.3 컴포넌트 기반 전자상거래 시스템

기존의 전자상거래 시스템들은 전형적인 클라이언트/서버 아키텍처를 사용함으로써 서버의 부하처리가 힘들뿐 아니라 분산 관리되는 응용한정적인 클라이언트 프로그램의 유지보수가 매우 어렵다. 따라서, 웹을 이용한 애플리케이션의 동적인 조정이 필요한 전자상거래 시스템에서는 잘 정의된 인터페이스에 기반한 즉각적인 사용자 및 기능적인 요구변화에 적용할 수 있는 컴포넌트 기반의 접근방법이 가장 적절한 대안이 된다. 즉, 컴포넌트 개발방법(CBD)은 객체지향 패러다임에 근거하여 비즈니스 로직을 포함하는 컴포넌트의 인터페이스를 통한 각 컴포넌트간 조합을 이용함으로써 새로운 소프트웨어를 개발하는 방법이다. 이를 위하여, 컴포넌트 명세화와 구현, 패키지, 생산된 컴포넌트의 재사용 관리 및 뿔, 조립에 의한 응용생성에 이르는 체계적인 프로세스가 컴포넌트 저장소를 중심으로 병행되어야 한다[6]. 이에 따라, 다음 <그림 2>와 같이 e-Business 컴포넌트는 비즈니스 프로세스 모델링을 통해 얻어진 비즈니스 로직과 기존의 컴포넌트와 설계, 데이터베이스, ERP(Enterprise Resource Planning) 인터페이스를 통해서 컴포넌트로 전환함으로써 e-Business 시스템의 변화와 개혁을 가져다준다.

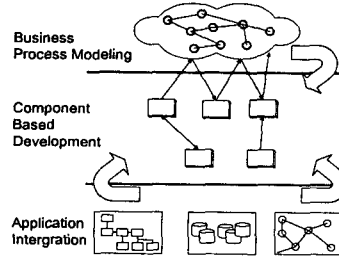


그림 2. e-Business 컴포넌트 구조

III. 전자상거래 시스템 구축을 위한 EJB컴포넌트 운용

기존의 전자상거래 시스템 구축은 3계층(3-Tier) 모델을 이용하였다. 이는 각 역할에 따라 Presentation Logic, Business Logic, Data Access Logic으로 나뉜다[7].

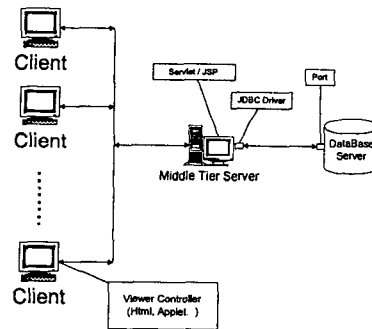


그림 3. 웹상에서의 3-Tier 모델

즉, <그림 3>과 같이 데이터베이스서버와 미들티어 서버를 별도로 두며, 비즈니스로직은 미들티어에 두어 클라이언트가 서비스요청을 처리하게 된다. 이는, 데이터베이스가 바뀔 경우 실제데이터베이스를 접속하는 미들티어 서버만 드라이버를 변경하면 되고, 업무가 변경될 경우 해당 미들티어 서버의 로직만 변경하면 되었다. 그러나, 이러한 구조는 서버의 부하처리가 힘들고 분산관리되는 응용한정적인 클라이언트 프로그램에 대한 유지보수가 어렵다[8]. 따라서, 다음 <그림 4>와 같이 데이터베이스를 접속하는 웹서버를 별도로 두어 클라이언트의 요구사항을 처리하는 웹서버와의 인터페이스를 통하여 해당 정보를 주고받는 다중서버환경에서의 구현방식이 가장 적절하다. 이는, 로직을 분산시킴으로써 서버의 부하를 줄일 수 있고, 분산되는 자원들을 효율적으로 관리할 수 있다.

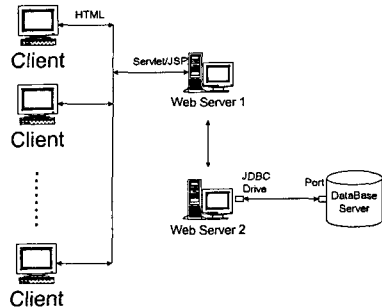


그림 4. 웹 상에서의 다중서버 환경

IV. EJB 컴포넌트 기반 전자상거래 워크플로우

EJB 컴포넌트 기반 전자상거래 시스템 구성은 <그림 5>와 같이 나타낼 수 있다. 클라이언트의 요구를 받은 Main 쇼핑몰 서버에서는 이를 분석하여 해당 구매EJB 컴포넌트 또는 재고 EJB컴포넌트 등으로 분류되어 서비스를 분할하며, 해당 EJB 컴포넌트 서버에서는 자체의 데이터베이스를 기준으로 서비스를 재개한다.

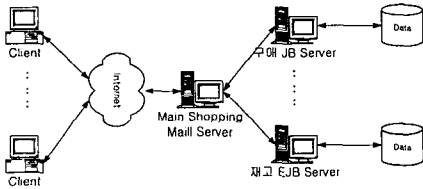


그림 5. 쇼핑몰에서 EJB 컴포넌트 운용

Main 쇼핑몰에서 구축되는 웹 서버의 구성은 EJB 컴포넌트의 적용일 경우 JSP와 서블릿을 둘 수 있다. 따라서, 서블릿을 위한 Resin Server의 가동이 필요하며, JSP, EJB를 위하여 JDK와 J2SDKEE가 서버에 적용되어 있어야 한다. <그림 6>과 같이 JSP와 서블릿에서 클라이언트의 요구를 분석하여 해당 EJB컴포넌트를 호출하게 되면, EJB 컴포넌트에서 데이터베이스와 연동되어 클라이언트의 요구를 처리하도록 한다.

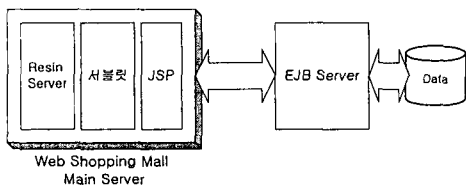


그림 6. EJB 컴포넌트 적용 워크플로우

<그림 7>은 EJB 적용 및 조립을 위한 Deployt-

ool을 나타낸다.

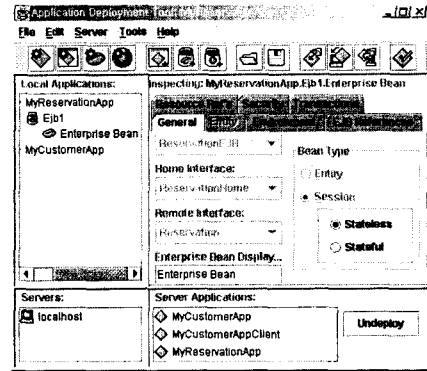


그림 7. EJB 컴포넌트 적용을 위한 Deploytool

또한, <그림 8>과 같이 데이터베이스 엔진으로 J2SDKEE에서 제공되는 Informix사의 Cloudscape를 사용할 수 있다.

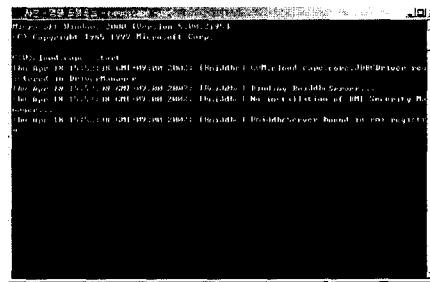


그림 8. Cloudscape 데이터베이스 엔진 가동

EJB컴포넌트가 클라이언트의 요구를 처리하기 위해서는 서버측에 <그림 9>와 같이 EJB서버가 가동되어야만 한다.

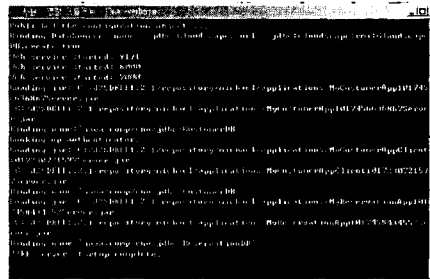


그림 9. EJB 서버 가동

V. 결 론

본 시스템은 전자상거래 시스템에서 EJB 컴포넌트를 기반으로 구현될 수 있는 워크플로우에 대하여 살펴보았다. 즉, 다중 EJB서버환경에서 Main Web Shopping Mail Server를 중심으로 각 EJB서버를 통하여 컴포넌트를 호출함으로써 이에 관한 서비스를 처리하는 방식으로 분산 웹 서버를 구성할 수 있다. 이러한 효과는 Main Web Server에서 중요한 로직과 해당 데이터베이스를 각 EJB 서버에 분산시킴으로써 Main Server의 부하를 줄일 수 있으며, 해당 로직에 관한 유지보수를 용이하게 할 수 있다. 또한, EJB 컴포넌트를 활용함으로써 Java가 갖는 이기종간의 이식성과 효율성을 활용할 수 있다.

향후 연구과제로는 합성되는 각 컴포넌트간의 명세에 따라 그 구조를 명확히 하고, 이를 효율적으로 사용하기 위한 컴포넌트 합성기술이 필요하다. 즉, 많은 수의 컴포넌트들이 합성되어 하나의 시스템을 이룰 경우 각 컴포넌트간의 관계와 로직의 구성, 효율적인 합성 및 운용 기법등이 필요하다.

참고문헌

- [1] 이동길외, "E-비즈니스와 확장형 ERP", MIT 경영과 정보기술, 2000.
- [2] 차정은, 김행곤, "전자상거래 시스템 구축을 위한 컴포넌트 아키텍처 및 명세 방법 연구", 한국정보처리학회 논문지 제7권 제5호, 2000. 5.
- [3] Cuno Pfister, Clemens Szyperski, "Why Objects are Not Enough", First International Component Users Conference, 1996.
- [4] Duane Morine, Brain Ballard, "Create Common Framework for your E-Business Applications", e-BUSINESS MAGAZINE, 1999, October.
- [5] Ed Roman, "Mastering Enterprise JavaBeans & the Java 2 Platform, Enterprise Ed", Wiley & Sons, 1999.
- [6] Paul Allen, Realizing e-Business with Components, Adison-Wesley, 2001.
- [7] "An Enterprise Architecture for Distributed Computing", <http://www.dcc.buffalo.edu/archives/meetings/011097gm1/index.html>, University at Buffalo Web, 05 Apr 2000.
- [8] Gunjan Sinha, "Build a Component Architecture for E-Commerce", e-BUSINESS MAGAZINE, 1999, March.