

# 사례 발표 **효율적인 웹 트랜잭션 처리를 위한 웹 어플리케이션 서버**

한 재 선\*, 박 대 연\*\*

● 목 차 ●

- 1. 서 론
- 2. 웹 어플리케이션 서버의 개념과 발전
- 3. 웹 어플리케이션 서버의 구조
- 4. JEUS 웹 어플리케이션 서버의 특징
- 5. 결 론

## 1. 서 론

오늘날 인터넷과 웹의 발전은 사회생활 전반에 걸쳐 커다란 영향을 끼치고 있다. 이러한 상황에서 웹 기반의 전자상거래가 폭발적으로 확산되면서 대규모의 웹 트랜잭션을 안정적으로 처리해 줄 기본 플랫폼의 필요성이 절실히 요구되었고 이에 웹 어플리케이션 서버가 등장하게 되었다. 더군다나 기업의 정보시스템 역시 웹을 기반으로 재편되면서 웹 어플리케이션 서버는 기업용 어플리케이션을 위한 필수적인 개발 플랫폼으로도 급부상하고 있다. 웹 어플리케이션 서버는 웹과 기업의 전산시스템 사이에 위치하면서 웹 기반의 분산시스템 개발을 쉽게 도와주고 안정적인 트랜잭션 처리를 보장해 주는 일종의 미들웨어로서 대규모 트랜잭션이 필요한 웹 기반시스템에서 교통경찰 역할을 하는 것이다. 본 논문에서는 우선 웹 어플리케이션 서버의 탄생배경과 기본적인 개념에 대해 살펴 본 후 (주)티맥스소프트에서 개발한 순수 국산 웹 어플리케이션 서버인 JEUS(Java Enterprise User Solu-

tion) 웹 어플리케이션 서버의 구조와 특징에 대해 분석하였다.

## 2. 웹 어플리케이션 서버의 개념과 발전

어플리케이션 서버는 Presentation Logic, 어플리케이션(또는 비즈니스 로직) 서버, 그리고 데이터베이스 및 트랜잭션 서버로 구성되는 3계층 클라이언트/서버 모델의 일부로써 자주 사용된다. 좀더 서술적으로, 그것은 하나의 어플리케이션을 다음과 같이 나누는 것으로 비쳐질 수 있다. 첫 번째 계층으로써, 즉 프론트엔드에는 대개 PC나 워크스테이션에서 웹 브라우저 기반의 그래픽 사용자 인터페이스를 말하며, 두 번째 계층으로는 비즈니스 로직 어플리케이션, 이것은 LAN이나 인트라넷 서버에 존재할 수도 있다. 세 번째 계층으로는, 즉 백엔드에는 데이터베이스 및 트랜잭션 서버가 자리하는데, 때로는 메인프레임이나 대형 서버 상에 존재하는 것을 말한다. 오랫동안 사용해온 데이터베이스와 트랜잭션 관리용 프로그램들은 백 엔드로써, 즉 세 번째 계층의 일부를 이룬다. 어플리케이션 서버는 브라우저 기반의 프론트엔드와 데이터베이스 시스템의 백 엔드 사이에 존재하는 중개인이라고

\* KAIST 전자전산학과 전기전자전공 박사과정  
 \*\* KAIST 전자전산학과 조교수

도 볼 수 있다[1, 2].

많은 경우에서, 어플리케이션 서버는 웹 서버(HTTP 서버)와 결합되거나 함께 일하는데, 이러한 것을 웹 어플리케이션 서버라고 부른다. 웹 브라우저는 사용자를 위해 만들기 쉬운HTML 기반의 프런트엔드를 지원한다. 웹 서버는 어플리케이션 서버에 작업요구를 전달하고, 또한 수정되었거나 새로운 웹 페이지를 사용자에게 되돌려주는 데 있어 몇 가지 다른 방법들을 제공한다. 이러한 접근 방식들에는CGI, FastCGI, ASP, 그리고 JSP 등이 있다. 어떤 경우에는, 웹 어플리케이션 서버들이 CORBA의 IIOP와 같은 브로커 인터페이스 요청을 지원하기도 한다.

웹 어플리케이션 서버는 웹 기반의 정보시스템에서 기업용 어플리케이션을 위한 필수적인 개발 플랫폼으로 웹 어플리케이션이 안정되게 개발되고 운영될 수 있도록 하는 서버의 역할을 수행한다. 웹 어플리케이션 서버는 기존 정보시스템을 단순히 쉽게 웹으로 바꿔주는 프런트엔드 기능 이외에도 클라이언트/서버 환경에서 가능했던 트랜잭션 처리, 대규모 사용자 지원, 부하 분산, 보안성 등 기능의 제약 없이 기존 정보시스템을 웹으로 교체할 수 있도록 해준다. 개발자 입장에서는 복잡한 코딩 작업 없이도 웹 어플리케이션 서버가 제공하는 서비스를 호출하기만 하면 손쉽게 웹 기반의 분산시스템을 개발 할 수 있다. 또한 웹 어플리케이션 서버는 다양한 관리기능을 제공하고 클라이언트/서버 환경에서 분산됐던 업무를 단일 시스템 안에서 관리할 수 있어, 관리 및 유지보수 측면에서 상당한 효과를 가져 다 준다.

최근의 웹 어플리케이션 서버 업계의 기술 및 동향을 살펴보면 그 내용은 다음과 같다. 첫 번째로는 EJB(Enterprise Java Beans)[3], XML, CORBA, COM 등 객체지향의 컴포넌트 기술을 대폭 수용함으로써, eBusiness의 플랫폼 엔진으로써의 완전한 위상을 확보함과 동시에 eBusiness용 어플리케이션

기능을 수용, 통합화의 방향으로 발전하고 있다는 점이다.

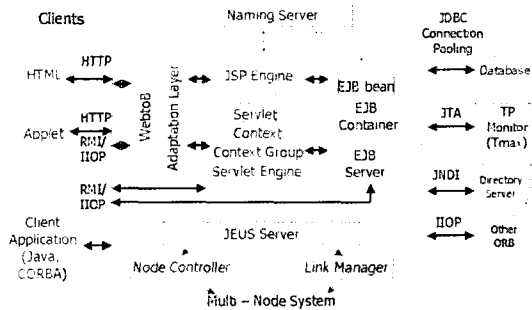
두 번째로는 J2EE[4]를 지원해 확장성을 높였으며, 사용자 필요에 따라 규모를 선택할 수 있도록 제품군의 다양화 및 적용범위를 확대함으로써, 사이버뱅킹, 카드업무, EDI, 통신망 관리, 교육, 금융, SI, 통신 등으로 그 업무영역이 확대되고 있다는 점을 들 수 있을 것이다.

세 번째로는 웹 어플리케이션 서버는 eBusiness 솔루션의 기반 미들웨어로 지속적인 성장이 예상되며, JAVA, XML, VoiceXML 등 최신 기술의 적용이 가속도가 붙어 진행이 되고 있으며, 다양하고 풍부한 웹 콘텐츠를 용이하게 관리 및 처리하기 위한 기능 보완이나 타 제품군과의 연계가 필요한 상황에 부응해 제품군들과의 진화가 이루어지고 있기도 하다.

마지막으로 전자상거래의 활성화에 따른 기업간 문서공유를 위한 XML 제품군과의 연계나 통합이 보편화되고 있으며, EAI(Enterprise Application Integration) 솔루션의 이기종 어플리케이션 통합능력과 웹 어플리케이션 서버의 웹 구현 기능을 결합한 전자적인 규모의 웹 정보시스템 구축에 대한 요구 및 수요가 높아지고 있기도 하다.

### 3. JEUS 웹 어플리케이션 서버의 구조

JEUS(Java Enterprise User Solution) 웹 어플리케이션 서버는 (주)티맥스소프트에서 자체 개발한 순수 국산 웹 어플리케이션 서버로서 인터넷으로 각광 받고 있는 자바를 기반으로 한 강력한 웹 솔루션으로 웹 환경에서 어플리케이션을 운용하는데 필요한 각종 서비스들을 제공해 주는 자바기반의 웹 어플리케이션 서버이다. JEUS의 전체적인 구조는 다음 (그림 1)과 같다[5].



(그림 1) JEUS 웹 어플리케이션 서버의 구조

### 3.1 JEUS Server

JEUS 웹 어플리케이션 서버의 핵심적인 부분으로 엔진들의 동작과 시스템의 관리를 위한 기반 구조를 제공하며 Node Controller와 Link Manager를 통하여 다중 노드환경에서 부하 분산과 장애 복구 등의 처리를 담당한다.

### 3.2 WebtoB Web Server

JEUS 웹 어플리케이션 서버의 가장 앞 단을 구성하고 있는 부분으로 사용자의 HTTP 요청을 받아 주는 웹 서버이다. WebtoB는 (주)티맥스소프트가 자체 개발한 순수 국산 웹 서버로서 다른 웹 서버와는 다른 독특한 구조를 가져 높은 성능과 다양한 기능을 제공하고 있다. JEUS 웹 어플리케이션 서버의 모든 HTTP 요청들은 우선 WebtoB에 의해 일차적으로 부하조절이 이루어 지게 된다.

### 3.3 Web Container

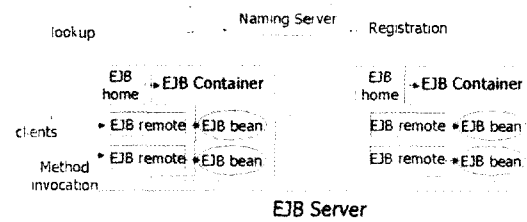
Presentation logic을 담당하는 엔진으로 Servlet과 JSP(Java Server Page) 요청을 처리해 돌려 준다. Web Container의 장점들은 다음과 같다[6].

- 부하에 따라 7여러 개의 Web Container를 구동시킬 수 있으며 이 때 자동으로 부하 조절과 장애 복구가 수행된다.
- 지능적인 쓰레드 풀링으로 컴퓨팅 자원을 효율적으로 사용한다.

- 웹 어플리케이션의 구조를 다중으로 구성함으로써 각 레벨로의 관리가 가능하다.
- 데이터베이스 연결을 자체 풀링함으로써 고속의 데이터 처리가 가능하다.
- 세션정보를 여러 Web Container 사이에 공유함으로써 장애 시에도 사용자의 세션 정보를 잃어버리지 않는다.

### 3.4 EJB(Enterprise Java Beans) Container

Enterprise Bean은 EJB Container상에서 수행된다. 컨테이너는 EJB를 제어하는 런타임 환경이며, 빈에 시스템 수준의 서비스를 제공한다. 개발자들은 이들 서비스를 개발할 필요가 없으므로, 단지 업무 로직에만 집중할 수 있다. EJB Container는 빈에게 트랜잭션 관리, 보안, 원격 클라이언트 연결성, 생명주기 관리, 데이터베이스 연결 풀링 등과 같은 서비스를 제공한다. JEUS 웹 어플리케이션 서버상에서 EJB의 동작과 구조는 (그림 2)와 같다[7].

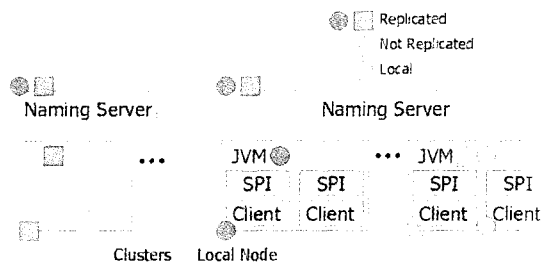


(그림 2) JEUS 웹 어플리케이션 서버에서의 EJB의 구조와 동작

### 3.5 Naming Server와 JNDI(Java Naming & Directory Interface)

JNDI는 J2EE에서 네이밍 및 디렉토리 서비스의 표준이다. 네이밍 서비스란 하나의 객체에 고유한 이름을 할당하여 그 객체에 대한 액세스를 시도하는 다른 객체가 이름만으로 해당 객체에 대한 레퍼런스를 얻을 수 있도록 해주는 것이다. 디렉토리 서비스란 네이밍 서비스를 포함하는 개념으로 단

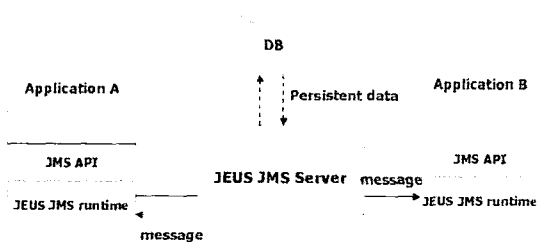
지 이름만 가지고 찾는 것 말고도 그 객체의 특성을 가지고도 찾을 수 있도록 하는 것이다. JEUS 웹 어플리케이션 서버는 독립적인 Naming Server와 JNDI API를 통해 다른 컴포넌트들(Servlet, JSP, EJB, JMS 등)에서 사용할 수 있게 한다. 네이밍 서비스의 동작은 다음 (그림 3)과 같다.



(그림 3) JEUS 웹 어플리케이션 서버의 네이밍 서비스 동작방식

### 3.6 JMS(Java Message Service) Server

기존의 메세징 오리엔티드 시스템은 기업 환경 컴퓨팅에서 나름의 중요한 위치를 차지하고 있었다. 이러한 시스템을 Java로 구현하고자 할 경우 JMS는 인터페이스 표준이 되어 준다. JMS는 기본적으로 Point-to-point 방식과 Publish / Subscribe 방식의 두 가지의 표준적 통신방법을 제시하고 그 이상의 서비스는 시스템 개발자가 직접 작성하도록 하고 있다. JEUS 웹 어플리케이션 서버의 JMS Server의 동작방식은 (그림 4)와 같다.



(그림 4) JEUS 웹 어플리케이션 서버의 JMS Server 동작방식

## 4. JEUS 웹 어플리케이션 서버의 특징

JEUS 웹 어플리케이션 서버는 다음과 같은 특징들을 가지고 있다.

### 4.1 다양한 프리젠테이션 로직 지원

- Servlet, JSP (Java Server Page), XML

### 4.2 편리한 비즈니스 로직 지원

- EJB (Enterprise Java Beans)

### 4.3 J2EE (Java 2 Enterprise Edition) 플랫폼과 호환

- JMS (Java Messaging Service) : 메시지 기반의 프로그래밍 기초를 제공
- JNDI (Java Naming and Directory Interface) : Naming Service를 제공
- JTS (Java Transaction Service) : 이기종 데이터 베이스에 걸친 글로벌 트랜잭션
- JDBC (Java Database Connectivity) Connection Pool : 데이터베이스와의 연결을 효율적으로 관리

### 4.4 다양한 엔진 종류 선택

- 어플리케이션의 성격에 따라 엔진의 타입 선택이 가능.
- 각 환경에 맞는 최적의 엔진 타입 설정

### 4.5 부하조절

- 각 서버(호스트)별, 서비스별 동적 부하 조절
- 하드웨어 성능에 따른 정적 부하조절

### 4.6 장애대책

- 서버(호스트)간의 상호감시
- 어플리케이션 프로세스에 대한 감시
- 세션 클러스터링

#### 4.7 다양한 보안기능 제공

- Login Name & Password
- 각 서비스에 대한 Access 권한 제한
- 클라이언트의 IP 어드레스 또는 도메인 제한
- CA(Certificate Agent)를 통한 사용자 인증
- SSL(Secure Socket Layer)의 사용

#### 4.8 편리한 관리자 도구의 제공

- 리소스 배치를 간편하게 할 수 있음
- 멀티 서버(호스트) 시스템을 통합적으로 제어 할 수 있음

### 5. 결론

폭발적인 성장세에 힘입은 웹 어플리케이션 서버 시장은 주도권 장악을 위하여 세계적으로 규모가 큰 대형회사들인 BEA, Oracle, Sun, IBM사 등이 업계의 선두자리를 지키기 위하여 치열한 경쟁을 벌이고 있다. 이런 상황에서 우수한 성능과 다양한 기능의 웹 어플리케이션 서버가 순수 국산 기술에 의해 개발된 것은 주목할 만한 일이다.

JEUS 웹 어플리케이션 서버는 기존의 제품들과는 달리 경량화 된 엔진 설계로 시스템의 하드웨어 요구량을 획기적으로 줄여 작은 사양의 하드웨어로도 시스템 구축이 가능하게 했으며 Java 기반의 시스템이면서도 C로 작성된 Web Server 모듈(WebtoB base)을 탑재하고 엔진들간 통신을 최적화 하여, 뛰어난 성능향상을 보인다. 이러한 성능향상을 통해 대규모의 웹 트랜잭션도 문제없이 처리할 수 있다.

### 참고문헌

- [1] R. Orfali, D. Harkey, J. Edwards, "The Essential Client/Server Survival Guide," Wiley Computer Publishing, 1996
- [2] S. M. Lewandowski, Frameworks for Component-Based Client/Server Computing, ACM Computer Surveys, Vol. 30, No. 1, March 1998
- [3] A. Thomas, Enterprise JavaBeans Technology, Patricia Seybold Group, December 1998
- [4] Sun Microsystems, Java 2 Platform Enterprise Edition Specification v1.3, 2001
- [5] (주)티맥스소프트, "JEUS Administration Guide Version 2.0", 2001
- [6] (주)티맥스소프트, "JEUS Web Container Guide Version 2.0", 2001
- [7] (주)티맥스소프트, "JEUS Enterprise Java Beans Guide Version 2.0", 2001

## 저자약력



한 재 선

1998년 부산대학교 전자공학과 (공학사)  
2000년 한국과학기술원 전기 및 전자공학과 (공학석사)  
2000년-현재 한국과학기술원 전자전산학과 전기전자  
전공 박사과정  
관심분야: 분산컴퓨팅, 인터넷 구조, 웹  
e-mail : jshan@sslslab.kaist.ac.kr



박 대 연

1990년 University of Oregon, Computer Science (공학  
사)  
1992년 University of Oregon, Computer Science (공학  
석사)  
1996년 University of Southern California, Computer  
Science (공학박사)  
1996년-1998년 한국외대 제어계측공학과 조교수  
1998년-현재 한국과학기술원 전자전산학과 조교수  
관심분야: 분산컴퓨팅, 무선인터넷 프로토콜, 인터넷  
및 웹 구조  
e-mail : daeyeon@ee.kaist.ac.kr