

오디세우스 객체지향 멀티미디어 데이터베이스 관리 시스템을 위한 웹-데이터베이스 게이트웨이의 설계 및 구현

이재길^{*}, 한옥신, 이민재, 황규영
{ ariel, wshan, mjlee, kywhang }@mozart.kaist.ac.kr
한국과학기술원 전산학과/첨단정보기술연구센터

Design and Implementation of a Web-Database Gateway for the ODYSSEUS Object-Oriented Multimedia Database Management System

Jae-Gil Lee, Wook-Shin Han, Min-Jae Lee and Kyu-Young Whang
Department of Computer Science and
Advanced Information Technology Research Center
Korea Advanced Institute of Science and Technology

요 약

최근 웹의 놀랄만한 성장으로 인터넷에 존재하는 데이터의 양은 기하 급수적으로 증가하고 있다. 이와 같은 방대한 양의 데이터를 체계적으로 관리, 검색하기 위해 데이터베이스 관리 시스템의 사용이 절실해지고, 웹 상에서 데이터베이스를 사용하기 위한 웹과 데이터베이스의 연동 방법에 관한 연구가 활발히 진행되고 있다. 그러나 현재까지 웹과 데이터베이스의 연동을 위해 개발된 웹-데이터베이스 게이트웨이는 응용 프로그램 작성의 편의성과 범용 프로그래밍 기능을 동시에 제공하지 못하고 있다. 본 논문에서는 객체지향 멀티미디어 데이터베이스 관리 시스템인 오디세우스를 위한 웹-데이터베이스 게이트웨이를 설계하고 구현한다. 본 논문에서 구현한 웹-데이터베이스 게이트웨이는 웹과 데이터베이스의 연동 방식으로 서버 전용 API 방식을 사용하기 때문에 사용자의 요청을 보다 빠르게 처리할 수 있다. 또한 본 논문에서 구현한 웹-데이터베이스 게이트웨이는 HTML 페이지 작성만으로 응용 프로그램 개발이 가능하므로 사용자가 보다 쉽게 웹 기반 데이터베이스 응용 프로그램을 개발할 수 있으며, 범용 프로그래밍 언어로 작성된 함수를 HTML 페이지 내에서 수행할 수 있는 기능을 제공하므로 다양한 웹 기반 데이터베이스 응용 프로그램을 개발할 수 있다.

1. 서 론

웹은 최근 놀랄만한 성장을 하고 있는 비통계적인 이질 분산 정보 시스템이다. 웹은 전세계에 걸쳐 네트워크 상에 존재하는 문서, 이미지, 비디오, 사운드 데이터 등과 같은 이질적인 정보의 액세스를 허용한다. 이러한 기능을 가진 웹과 정보를 저장하고 관리하는 DBMS의 통합은 필연적이다[Ger97]. DBMS의 입장에서 보면 웹을 통하여 기존에 구축된 정보를 직접 원거리에서 액세스하는 것이 가능하고 웹의 입장에서 보면 원하는 정보를 검색하기 위해 DBMS의 검색 기능을 사용할 수 있을 뿐만 아니라 DBMS의 데이터 관리 기능을 이용하여 데이터의 일관성 문제를 처리할 수 있다.

웹과 DBMS의 통합 기술은 데이터베이스를 액세스하는 응용의 위치에 따라 웹 서버측에서의 데이터베이스 액세스 방법과 클라이언트측에서의 데이터베이스 액세스 방법으로 분류될 수 있다[Kim96]. 웹 서버측에서의 데이터베이스 액세스 방법은 웹-데이터베이스 게이트웨이를 이용하여 데이터베이스를 액세스하는 방식으로 매 사용자 요청마다 새로운 웹 페이지가 요구되는 응용 프로그램에 주로 사용된다. 웹 클라이언트측에서의 데이터베이스 액세스 방법은 웹 클라이언트의 Java 애플릿을 통해서 데이터베이스를 액세스하는 방식으로 주로 대화형 응용 프로그램에 사용된다.

주요 상용 DBMS 업체들은 웹 기반 DBMS 응용 프로그램을 쉽게 작성할 수 있도록 자사의 DBMS를 위한 웹-데이터베이스 게이트웨이를 제공하고 있다. 현재 상용화된 제품으로는 UniSQL사의 UniWeb[Uni97], Informix사의 Web Datablade[In98], Oracle사의 Oracle Application Server[Or98] 등이 있다. 그러나 이러한 제품들은 모두 프로그래밍의 편의성과 범용 프로그래밍 기능을 동시에 지원하지 못하는 문제가 있다. 또한 이들 제품이 사용하는 DBMS들은 모두 정보 검색 엔진과 DBMS 엔진이 밀접하게 연결되어 있지 않기 때문에 대규모 정보 검색시 검색 성능이 저하될 수 밖에 없다.

본 논문에서는 한국과학기술원 전산학과 데이터베이스 및 멀티미디어 연구실에서 1995년 7월에 발표한 객체지향 멀티미디어 DBMS인 오디세우스[Par94]를 위한 웹-데이터베이스 게이트웨이를 설계하고 구현한다. 본 논문의 웹-데이터베이스 게이트웨이는 프로그래밍의 편의성과 범용 프로그래밍 기능을 동시에 얻을 수 있도록 설계 및 구현된다. 오디세우스는 정보 검색 엔진과 DBMS 엔진이 밀접하게 연결되어 있기 때문에 대규모 정보 검색에서 빠른 검색 속도를 얻을 수 있다. 따라서 본 논문에서 구현한 웹-데이터베이스 게이트웨이는 전자 도서관, 논문 검색 시스템, 웹 검색 엔진 등의 다양한 응용 프로그램들을 쉽게 개발할 수 있게 해준다.

본 논문의 구성의 다음과 같다. 제 2 장에서는 관련 연구로서 UniWeb, Informix Web Datablade, Oracle Web Application Server에 대해서 설명한다. 제 3 장에서는 오디세우스를 위한 웹-데이터베이스 게이트웨이의 설계 및 구현 방법에 대해서 설명한다. 제 4 장에서는 결론을 내리고 논문의 공헌을 요약한다.

2. 관련 연구

본 장에서는 웹-데이터베이스 게이트웨이의 분류 방법 및 종류에 대해서 설명하고 주요 상용 DBMS 업체에서 제공하는 웹-데이터베이스 게이트웨이를 살펴본다. 웹-데이터베이스 게이트웨이는 웹 서버와 데이터베이스를 연동하는 방법과 제공하는 프로그래밍 인터페이스에 따라 분류된다.

웹 서버와 데이터베이스를 연동하는 방법은 크게 CGI 방식, 서버 전용 API 방식, 전용 서버 방식으로 나눌 수 있다[Kim96]. CGI 방식은 매 사용자의 요청마다 CGI 프로세스가 동적으로 생성되어 데이터베이스를 액세스하는 방법이다. 서버 전용 API 방식은 웹 서버에서 제공하는 API를 사용해서 매 사용자의 요청마다 웹 서버에 등록된 함수를 호출하여 데이터베이스를 액세스하는 방법이다. 전용 서버 방식

* 본 연구는 첨단정보기술연구센터를 통하여 과학재단의 지원과 웹 기반 정보검색 OODBMS를 위한 Web-DBMS 통합기술의 개발과제를 통해 과학기술부의 지원을 받았음.

은 DBMS 기능이 이미 내장된 전용 웹 서버를 사용하여 데이터베이스를 액세스하는 방법이다.

웹-데이터베이스 게이트웨이에서 제공하는 프로그래밍 인터페이스는 태그, 스크립트 언어, 범용 프로그래밍 언어 방식으로 나눌 수 있다. 태그 방식은 HTML 페이지 내에서 데이터베이스를 액세스 할 수 있도록 확장된 태그를 사용하는 방식이다. 스크립트 언어 방식은 HTML 페이지 내에서 데이터베이스를 액세스 할 수 있는 스크립트 언어 및 스크립트 라이브러리를 사용하는 방식이다. 범용 프로그래밍 언어 방식은 범용 프로그래밍 언어에 데이터베이스 액세스 기능과 HTML 태그 출력 기능을 추가한 언어를 제공하는 방식이다.

본 장의 구성은 다음과 같다. 제 2.1 절에서는 UniSQL 사의 UniWeb 에 대해서 살펴본다. 제 2.2 절에서는 Informix 사의 Web Datablade 에 대해서 살펴본다. 제 2.3 절에서는 Oracle 사의 Oracle Application Server 에 대해서 살펴본다.

2.1 UniWeb

UniWeb 은 UniSQL 의 UniSQL/X 를 위한 웹-데이터베이스 게이트웨이이다. UniWeb 은 개발자를 위한 풀과 런타임 환경으로 구성된다. 개발자는 UniWeb 에서 제공하는 프로그래밍 인터페이스를 통해 UniSQL/X 응용 프로그램을 작성할 수 있다. 이러한 DBMS 응용 프로그램은 런타임시에 UniWeb 에서 제공하는 런타임 환경에 의해 서버에서 자동으로 수행된다.

UniWeb 은 웹과 데이터베이스의 연동 방법으로 기존의 CGI 방식의 성능 문제를 어느 정도 해결한 Fast-CGI 라는 방식을 사용한다 [Uni97]. Fast-CGI 방식은 데이터베이스에 접속하는 프로세스를 데몬 방식으로 바꾸어 사용자의 요청이 들어올 때마다 디스패처 프로세스가 생성되고 디스패처 프로세스가 데몬 프로세스와 통신하여 데이터베이스를 액세스한다. Fast-CGI 방식은 사용자의 요청마다 생성되는 디스패처 프로세스가 일반적으로 데이터베이스에 접속하는 프로세스보다 크기가 작기 때문에 기존의 CGI 방식에 비해 서버에 주는 부하가 감소한다 [Kim96]. 그러나 CGI 방식이 가지는 성능 저하 문제를 근본적으로 해결한 것은 아니기 때문에 동시에 사용자의 요청이 증가할 경우 성능 저하 문제가 발생할 수 있다.

UniWeb 은 프로그래밍 인터페이스로 UniTCL 이라는 스크립트 언어를 제공한다 [Uni97]. UniTCL 은 데이터베이스를 액세스하고 질의문을 수행할 수 있도록 기존의 TCL 스크립트 언어를 확장한 것이다. 응용 프로그램 작성 방법은 HTML 페이지 내에 UniTCL 코드를 삽입하고 그 시작과 끝을 각각 [] 문자로 표시한다. 이와 같이 삽입된 UniTCL 스크립트는 UniWeb 에 의해 런 타임시에 처리 결과로 대치되어 웹 브라우저로 출력된다. UniTCL 은 TCL 에 기반한 스크립트 언어이므로 범용 프로그래밍 기능을 지원하나 일반 사용자들에게는 생소한 스크립트 언어이기 때문에 UniWeb 의 사용을 위해서는 UniTCL 스크립트 언어를 새로 배워야 하는 문제가 있다.

2.2 Informix Web Datablade

Web Datablade 는 Informix 의 Universal Database Server 를 위한 웹-데이터베이스 게이트웨이이다. Informix 는 Datablade 라는 서버 확장 기법을 사용해 DBMS 에 특정 기능을 추가할 수 있는 구조를 제공하고 있다. Web Datablade 도 그의 일종으로 DBMS 에 웹 기반 데이터베이스 응용 프로그램을 수행할 수 있는 기능을 추가한다.

Web Datablade 는 웹과 데이터베이스의 연동 방식으로 CGI 방식과 서버 전용 API 방식을 사용한다 [Inf98]. CGI 방식의 Web Datablade 에서는 사용자의 요청이 들어올 때마다 Webdriver 라는 CGI 프로그램을 수행하여 데이터베이스에 접속한다. Webdriver 는 Universal Server 에서 응용 프로그램 페이지를 꺼내오기 위해 WebExplode 함수를 호출하는 SQL 질의식을 구성한다. 최근 버전의 Web Datablade 에서는 CGI 방식의 Webdriver 외에 Netscape Server API (NSAPI) [Net97] 를 사용해 구현된 Webdriver 또한 제공하고 있다. CGI 방식을 사용하는 Web Datablade 는 사용자의 요청마다 매번 Webdriver 프로세스가 생성되므로 동시에 사용자의 요청이 증가할 경우 성능 저하가 발생할 수 있다.

Web Datablade 는 프로그래밍 인터페이스로 HTML 태그와 유사한 형태의 Web Datablade 태그를 지원한다 [Inf98]. Web Datablade 태그는 웹 기반 데이터베이스 응용 프로그램을 개발하는데 필요한 기능을 제공한다. Web Datablade 태그는 총 4 개로 MISQL, MIVAR, MIBLOCK,

MIERROR 가 있다. 응용 프로그램 작성 방법은 HTML 페이지에 Web Datablade 태그를 일반 HTML 태그와 같이 삽입하는 것이다. 이와 같이 삽입된 Web Datablade 태그는 WebExplode 함수에서 런타임시에 처리 결과로 대치되어 웹 브라우저로 출력된다. Web Datablade 에서는 웹 기반 데이터베이스 응용 프로그램을 개발하기 위해 HTML 페이지만을 작성하면 되므로 쉽게 응용 프로그램을 개발할 수 있다는 장점이 있으나 HTML 태그와 유사한 형태의 Web Datablade 태그로는 범용 프로그래밍 기능을 제공하지 못한다는 문제가 있다.

2.3 Oracle Application Server

Oracle Application Server (OAS) 는 Oracle 의 Oracle7/8 을 위한 웹-데이터베이스 게이트웨이이다. Oracle Application Server 는 Data Server, Application Server, Thin Client 로 구성되는 3-tier 아키텍처를 가지고 있다. 3-tier 아키텍처는 기존의 2-tier 아키텍처의 중간에 하나의 계층을 더 두는 것이다. 3-tier 아키텍처는 Application Server 에서 프로그램 수행 환경에 따라 Thin Client 나 Data Server 의 작업을 분담하므로 Thin Client 나 Data Server 의 오버헤드를 줄여준다. 또한 사용자는 Cartridge 라는 서버 확장 기법을 사용하여 Application Server 에 프로그래밍 인터페이스를 추가할 수 있다 [Ora98]. Cartridge 에는 PL/SQL, C/C++, Perl, Servlet Cartridge 등이 있으며 이러한 Cartridge 를 설치하면 Cartridge 에서 제공하는 프로그래밍 언어를 사용할 수 있게 된다.

OAS 는 웹과 데이터베이스의 연동 방식으로 전용 서버 방식을 사용한다 [Ora98]. OAS 에 포함되어 있는 Oracle Web Server 로 들어온 사용자 요청은 OAS 로 전달되고, OAS 는 각 Cartridge 마다 별도의 프로세스를 유지하면서 웹 서버에서 받은 사용자 요청을 Cartridge 프로세스로 전달해서 처리한다. 따라서 OAS 는 사용자의 요청을 빠르게 처리할 수 있다는 장점이 있으나 Oracle Web Server 이외의 다른 웹 서버를 사용할 수 없다는 문제점을 가지고 있다 [Ora98].

OAS 는 프로그래밍 인터페이스로 범용 프로그래밍 언어 방식을 제공한다 [Ora98]. 사용자는 설치된 Cartridge 가 제공하는 범용 프로그래밍 언어를 통해서 응용 프로그램을 작성한다. OAS 는 이러한 Cartridge 방식을 채택하여 다양한 프로그래밍 환경을 제공한다. 그러나 범용 프로그래밍 언어 방식은 다른 방식과 달리 HTML 페이지 내에 응용 프로그램을 삽입하는 방식이 아니기 때문에 HTML 페이지 구성이 더 어렵다는 단점을 가지고 있다.

3. 웹-게이트웨이의 설계 및 구현

본 장에서는 오디세우스를 위한 웹-데이터베이스 게이트웨이의 설계 및 구현 방법에 대해서 설명한다. 본 장에서 설계하고 구현한 웹-데이터베이스 게이트웨이의 목적은 사용자의 요청을 빠르게 처리하면서 프로그래밍의 편의성과 범용 프로그래밍 기능을 동시에 지원하는 것이다.

본 장의 구성은 다음과 같다. 제 3.1 절에서는 웹과 데이터베이스의 연동 방법에 대해서 설명한다. 제 3.2 절에서는 제공하는 프로그래밍 인터페이스에 대해서 설명한다. 제 3.3 절에서는 범용 프로그래밍 기능의 지원 방법에 대해서 설명한다.

3.1 웹과 데이터베이스의 연동

본 논문에서는 웹과 데이터베이스의 연동 방식으로 서버 전용 API 방식을 사용한다. CGI 방식은 매 사용자의 요청마다 CGI 프로세스가 생성되기 때문에 사용자의 요청이 급증할 경우 처리 성능이 저하되는 반면 서버 전용 API 방식은 매 사용자의 요청마다 쓰레드가 생성되므로 사용자의 요청이 급증하더라도 성능 저하가 그리 심하지 않다 [Kim96]. 본 논문에서는 서버 전용 API 중 Netscape Enterprise Server 에서 제공하는 Netscape Server API (NSAPI) 를 사용하여 웹-데이터베이스 게이트웨이를 구현한다.

본 논문에서 구현한 웹-데이터베이스 게이트웨이의 아키텍처는 그림 1 과 같다. Netscape Enterprise Server 는 NSAPI 를 사용하여 데이터베이스 응용 프로그램의 수행 요청이 들어오면 웹-데이터베이스 게이트웨이 라이브러리를 동적 링크하여 라이브러리 내부에 있는 함수를 호출하도록 수정된다. 또한 웹-데이터베이스 게이트웨이는 오디세우스 DBMS 라이브러리를 동적 링크하여 라이브러리 내부에 있는 DBMS API 를 호출한다. 그림 1 에서 보는 바와 같이 웹-데이터베이스 게이트웨이 내의 함수와 오디세우스 DBMS API 는 웹 서버 프로세스

내에서 수행된다. 그림 1 과 같은 아키텍처는 각 모듈간의 통신을 위해 프로세스간 통신(Interprocess Communication: IPC)을 사용하지 않고 단지 라이브러리 함수 호출을 사용하므로 사용자의 요청을 빠르게 처리할 수 있다.

본 논문에서 구현한 웹-데이터베이스 게이트웨이에서 사용자의 요청을 처리하는 과정은 다음과 같다. (1) 수행하고자 하는 응용 프로그램을 URL 을 통해 지정한다. (2) 사용자의 요청을 받아 웹-데이터베이스 게이트웨이의 함수를 호출한다. (3) 사용자가 요청한 응용 프로그램을 데이터베이스로부터 가져오는 질의식을 수행한다. (4) 데이터베이스로부터 응용 프로그램을 읽어서 웹-데이터베이스 게이트웨이에 전달한다. (5) 응용 프로그램의 수행 결과로 나오는 HTML 페이지를 웹 서버에 전달한다. (6) HTML 페이지를 웹 브라우저로 전달한다.

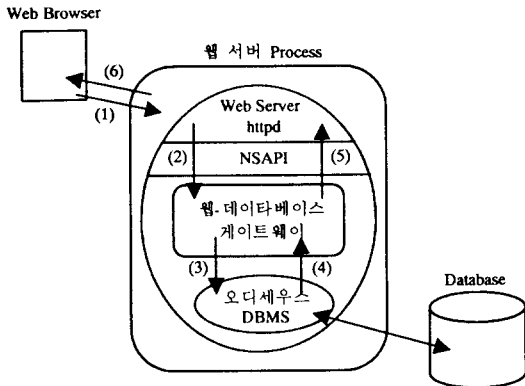


그림 1. 웹-데이터베이스 게이트웨이 아키텍처

3.2 프로그래밍 인터페이스

본 논문에서 구현한 웹-데이터베이스 게이트웨이는 프로그래밍 인터페이스로 태그 방식을 지원한다. 태그 방식을 채택한 이유는 사용자들이 가장 쉽게 응용 프로그램을 작성할 수 있는 인터페이스를 제공하기 위해서다. 태그 방식은 HTML 페이지에 확장된 태그를 삽입하는 작업만으로 별도의 프로그래밍 작업 없이 데이터베이스 응용 프로그램을 작성할 수 있으므로 사용자가 쉽게 응용 프로그램을 작성할 수 있다.

프로그래밍 인터페이스에서 지원하는 확장된 태그의 종류는 5 가지이며 크게 데이터베이스 액세스 기능과 변수 처리, 조건 분기, 에러 처리 등의 범용 프로그래밍 언어에 포함된 기능을 지원한다. 태그의 기능면에서 본 웹-데이터베이스 게이트웨이에서 제공하는 태그의 기능은 제 2.2 절에서 설명한 Web DataBlade 태그들의 기능을 모두 포함한다. 본 웹-데이터베이스 게이트웨이에서 제공하는 태그의 기능들은 아래와 같다.

- 오디세우스의 질의어를 수행하고 질의 결과를 출력한다
- HTML 페이지에서 변수를 정의하고 변수의 값을 출력한다.
- HTML 페이지에서 조건이 참이면 블록을 수행한다.
- HTML 페이지에서 조건이 참이면 블록을 반복 수행한다.
- 태그 수행시 발생한 에러를 처리한다.

응용 프로그램을 작성하는데 있어 가장 핵심적으로 사용되는 태그는 오디세우스의 질의어[Woo95]를 수행하는 태그이다. 이 태그는 수행하고자 하는 오디세우스 질의식을 태그의 속성 값으로 입력 받고, 속성 값으로 전달된 질의식을 오디세우스 DBMS 를 사용하여 수행한다. 그리고 질의 수행 결과를 HTML 페이지 내에서 사용할 수 있도록 질의 결과와 특정 시스템 변수를 바인딩 시켜주는 방법을 제공한다.

3.3 범용 프로그래밍 기능

프로그래밍 인터페이스로 태그만을 제공하면 태그에서 지원하지 않는 기능은 응용 프로그램에서 사용할 수 없다. 프로그래밍 인터페이스에서 제공하는 태그에는 웹 기반 데이터베이스 응용 프로그램 작성에 필수적인 기능만이 포함되어 있기 때문에 C/C++, Java, PASCAL 등과 같은 범용 프로그래밍 언어에 비해 표현력이 떨어지고 범용 프로그래밍 기능이 결여되어 있다. 이로 인해 웹-데이터베이스 게이트웨이로 작성할 수 있는 응용 프로그램의 범위에 제한을 받을 수 있으며 이는 점차 응용 프로그램의 요구가 다양해지고 있는 현실점에서 심각한 문제가 될 수 있다.

본 논문에서 구현한 웹-데이터베이스 게이트웨이는 범용 프로그래밍 기능을 지원하기 위해 C 언어로 작성된 외부 함수를 호출할 수 있는 구조를 지원한다. 사용자는 외부 함수를 사용하기 위해서 C 언어로 함수를 작성하고 컴파일하여 공유 라이브러리를 만든다. 그리고 함수 이름, 함수의 파라미터 정보, 공유 라이브러리의 위치 등 외부 함수의 정보를 데이터베이스에 저장한다.

사용자는 HTML 페이지에 확장된 태그를 삽입해서 외부 함수를 호출할 수 있다. 외부 함수가 호출되면 데이터베이스에서 외부 함수에 관한 정보를 읽은 다음 공유 라이브러리를 동적 링크하여 라이브러리에 포함되어 있는 외부 함수를 수행한다. 외부 함수 수행시 IPC 불 없이 수행 성능을 높일 수 있도록 외부 함수는 웹 서버와 동일한 프로세스 내에서 수행된다. 외부 함수의 수행 결과는 웹-데이터베이스 게이트웨이로 다시 전달되어 HTML 페이지에 삽입된다. 따라서 사용자는 C 언어로 작성한 함수를 마치 웹-데이터베이스 게이트웨이에 내장된 함수와 전혀 다름없이 사용할 수 있다.

4. 결 론

본 논문에서는 한국과학기술원 전산학과 데이터베이스 및 멀티미디어 연구실에서 개발한 객체지향 멀티미디어 데이터베이스 관리시스템인 오디세우스를 위한 웹-데이터베이스 게이트웨이를 설계하고 구현하였다.

본 논문의 공헌은 다음과 같이 크게 세 가지로 요약할 수 있다. 첫째, 웹과 데이터베이스의 연동 방식으로 서버 전용 API 방식을 사용하므로 매 사용자의 요청을 보다 빠르게 처리할 수 있다. 둘째, 프로그래밍 인터페이스로 태그 방식을 지원하므로 사용자가 보다 쉽게 응용 프로그램을 작성할 수 있다. 셋째, C 언어로 작성한 외부 함수를 호출하는 구조를 지원하므로 범용 프로그래밍 기능을 사용할 수 있다. 이로서 본 논문에서 구현한 웹-데이터베이스 게이트웨이는 사용자의 요청을 빠르게 처리할 뿐만 아니라 프로그래밍의 편의성과 범용 프로그래밍 기능을 동시에 제공한다.

참고 문헌

[Ger97] Gerald Ehmayer, Gerti Kappel, and Siegfried Reich, "Connecting Databases to the Web: A Taxonomy of Gateways," DEXA 1997, pp 1-15, 1997 년.

[Inf98] Informix, Informix Web DataBlade Module User's Guide, Version 3.3, 1998 년.

[Kim96] 김평철, "웹을 위한 데이터베이스 통로의 분류체계," 제 3 차 World Wide Web 워크숍 발표 자료집, 부산, 1996 년 3 월.

[Net97] Netscape, NSAPI Programmer's Guide, 1997 년.

[Ora98] Oracle, Oracle Application Server 4.0 White Paper: Product Overview, 1998 년.

[Par94] 박종목, 심재근, 이종학, 우준호, 조완섭, 황규영, "ODYSSEUS: UNIX 용 다사용자 객체지향 데이터베이스 시스템," 1994 년도 춘계 정보과학회 학술발표 논문집, 제 21 권, 제 2 호, pp.31-34, 1994 년.

[Uni97] UniSQL, UniWeb 3.0 매뉴얼, 1997 년.

[Woo95] 우준호, ODYSSEUS 객체지향 데이터베이스 시스템을 위한 질의처리의 설계 및 구현, 석사 학위 논문, KAIST 전산학과, 1995 년.