



플랫폼 독립성이 확산 원동력

현재 인터넷 서비스를 제공하는 다양한 서버들은 많은 데이터의 관리를 데이터베이스 관리 시스템(DBMS)에 의존하고 있다. 이것은 자바를 통해 데이터베이스에 있는 정보를 어떤 형태로든 활용할 필요가 생김을 의미한다. 그러면, Java를 통해 데이터베이스를 활용하는 프로그램을 작성하는 방법에는 어떤 것들이 있을까? 이 글에서는 이러한 궁금증을 출발점으로 삼아 데이터베이스에서의 Java 수용 현황에 대해서 살펴 보고, 현재 이루어지고 있는 Java를 통한 DB 접근 방안에 대해서 살펴 본다.

■ 장성우/ 한국 오라클 제품기획실 서버기술팀장
(swchang@kr.oracle.com)

연재 순서

- 1 Java와 DB의 통합 - 이번호
- 2 오라클8i 상에서의 Java에 기반한 DB 프로그래밍 방안

최

근 인터넷에 기반한 컴퓨팅 환경이 기업 컴퓨팅 환경에서도 빠른 속도로 받아들여지고 있다.

인터넷은 상태 정보가 유지되지 않은 채로 비연결 정보 교환만을 지원하던 정적인 차원에서, 동적이고 연결 안전성을 지원하는 트랜잭션의 기능을 제공하는 형태로 발전함으로써 기업에게는 비용 구조의 감소 및 고객 저변의 증가와 수요자와 공급자 사이의 긴밀한 유대의 확대를 통하여 기업을 확장할 수 있는 기회를 제공한다. 이런 기회를 인지한 선두 업체들은 인터넷과 인터넷의 협력을 통한 기업 규모의 응용 프로그램의 개발과 활용에 총력을 경주하고 있다.

그러나 Java 프로그래밍의 출현 이전에는 인터넷은 사업으로서의 대상과는 거리가 있었다. 인터넷 콘텐츠를 관리하기가 어려웠고, 또한 이러한 콘텐츠를 접근하는 응용 프로그램의 작성 및 운용/유지가 어려웠었다.

최근까지도 응용 프로그램 개발자는 수많은 다양한 종류의 개발 언어와 툴과 다른 특성을 가지는 많은 서버와 미들웨어들을 인터넷 프로그램을 개발하기 위해서 학습을 해야 했다. 그러므로 인력 관리에 많은 비용을 지출해야 했고 생산성도 낮았던 것이 사실이었다.

최근 Java가 인터넷에서의 표준 언어로 자리잡게 되었다. Java는 특정 하드웨어나 운영체제에 종속적이지 않은 플랫폼 독립성(platform independence) 특징을 갖추고 있어서, 한 번 작성되면 어떤 플랫폼 상에서도 수행될 수 있는 특징을 가지는데, 이것이 Java의 확산에 결정적인 계기가 되었다.

또한, 다른 객체 지향 언어에 비해 단순한 개념 및 문맥을 제공

하여 상대적으로 프로그래밍이 쉽고 또한 보안에 대한 지원이 잘 되어 있는 등 인터넷 환경에서 유용하게 활용될 수 있는 다양한 장점들을 가지고 있어서 인터넷 환경에서의 프로그래밍에 있어서 주요 개발 언어로서 자리잡게 되었다.

DB에서의 Java 활용 방안

Java를 통해 데이터베이스를 활용하는 방안은 다음과 같은 2가지 형태로 구분할 수 있다.

- 데이터베이스 외부에서 Java로 프로그램을 작성하고, 이 프로그램 안에서 데이터베이스에 접근하는 방안 : 이것은 Java 프로그램과 데이터베이스 간의 연결을 통한 정보 접근에 중점을 두는 방안으로서 개방되고 표준화된 고수준의 응용 프로그램 인터페이스(API)를 통해 어떤 데이터베이스에도 접근하여 질의를 수행할 수 있도록 지원하는 것에 초점을 맞추고 있다. Java 프로그램은 데이터베이스 밖에 존재하는 Java 수행 환경, 즉 JVM(Java Virtual Machine) 상에서 동작하게 되며, Java 프로그램이 데이터베이스 서버와는 논리적으로 분리되어 별도로 존재하므로 Java 프로그램과 데이터베이스 사이에는 약결합(loosely coupled) 되어진다.



- 데이터베이스 내부에서 Java로 데이터 처리 프로그램을 직접 작성하는 방안 : 이 방안은 데이터베이스 엔진에 Java 수행 환경인 JVM이 존재하여 Java 프로그램이 직접 데이터베이스 엔진 안에서 수행될 수 있도록 지원한다. 따라서, Java 프로그램과 데이터베이스는 밀결합(tightly coupled) 되어진다.

전자의 방안으로는 JDBC와 SQLJ를 통한 데이터베이스 접근 방법이 있고, 후자의 방법으로는 Java 내장 프로시저, EJB, Java로 작성한 CORBA 서버 모듈 등이 있다.

가. DB 외부에서의 Java를 통한 DB 접속 방안

JDBC

JDBC(Java DataBase Connectivity)는 Java 프로그램 상에서 개방된 API를 통해 어떤 데이터베이스에도 접근하여 질의를 수행하고 결과를 받아 처리할 수 있도록 지원하는 개방된 API의 집합으로서, Java에 기반한 표준화된 데이터베이스 접근을 위한 프레임워크를 제공한다. 프로그래머는 Java 프로그램 상에서 1) 데이터베이스와의 접속, 2) 데이터베이스 상에서 실행되어지기를 원하는 명령 전달, 3) 처리 결과 집합의 수신 및 이의 활용 등의 단계로 작업을 수행하게 된다.

JDBC를 활용할 때 중요 이슈는 Java 프로그램 상에 존재하는

객체들을 목표 데이터베이스에 맞게 사상시켜 주는 작업이다. JDBC는 현재 관계형 모델로의 사상만을 지원하기 때문에 Java 프로그램 상에서의 복잡한 객체를 분해하여 관계형 테이블의 필드로 사상시켜 주어야만 한다. 또, 반대로 관계형 테이블 필드로부터 정보를 모아 객체를 구성할 수 있도록 해 주어야 한다.

SQLJ

SQLJ는 Java로 작성된 프로그램 안에서 곧바로 SQL 문장을 사용할 수 있도록 지원한다. 즉, Java 언어에 SQL 문장을 곧바로 포함시켜 프로그램을 작성할 수

있도록 지원하는 전통적인 내재 언어(embedded language)가 된다. 이것은 JDBC가 제공하는 다양한 API를 일일이 지정하여 프로그램을 작성하는 복잡함 내지는 번거로움을 해결하여 주기 위해 제안된 방안으로서, SQLJ를 활용하여 작성된 프로그램은 전처리기(preprocessor)를 통해 JDBC를 통한 데이터베이스 접근 루틴으로 자동 변환되어지게 된다.

그런 후, 변환된 루틴이 Java 컴파일러에 의해 수행되어지게 된다. 자동적으로 변환된 JDBC 호출은 어떤 JDBC 드라이버에 의해서도 수행될 수 있다. SQLJ는 오라클을 비롯하여 IBM, Tandem, Sybase, JavaSoft 등에 의해 정의되어 표준안으로 제출

되었으며, ANSI에 의해 표준(X3.135.10-1998)으로 지정되었다.

나. DB 내부에서의 Java를 통한 DB 프로그래밍 방안

Java 저장 프로시저(Stored Procedures)



(그림) 데이터베이스 내부에서의 Java 지원 구조

저장 프로시저는 확장된 SQL을 이용해서 비즈니스 로직을 구현해야 하고 또 이 루틴이 데이터베이스 엔진 안에서 수행되어 빠른 수행 성능을 얻을 필요가 있을 때 사용된다. Java 저장 프로시저는 별도로 개발된 Java 프로그램이 데이터베이스 안에

서 저장 프로시저의 형태로 저장 및 실행될 수 있는 환경을 제공한다.

프로그램 내부적으로는 Java 저장 프로시저 역시 자료 접근에 있어서 JDBC나 SQLJ를 사용한다. 하지만, 이 루틴이 데이터베이스 서버 내에서 실행되기 때문에 자료의 접근은 네트워크를 사이에 두고 다른 기계에서 자료를 가지고 오는 경우보다 훨씬 빠르게 된다. Java 저장 프로그램은 여러 환경에서 수행될 수 있는데, 사용자 정의 함수나 SQL 질의에서 곧바로 호출되어 사용될 수 있다.

그러므로 저장 프로시저 개발자는 SQL을 Java를 이용해서 확장할 수 있게 된다. Java 저장 프로그램은 SQL과 Java 타입간의 표준 JDBC 매핑을 지원한다. Java 저장 프로그램은 사용자가 데이터베이스 서버에서 개발된 프로그램 언어를 사용하도록 한다. Java 저장 프로그램은 ANSI/ISO 규격을 따르고 있다.

Enterprise Java Beans

EJB(Enterprise Java Beans) 컴포넌트는 Java 응용 프로그램을 클라이언트가 아닌 서버 상에서 컴포넌트 모델에 기반하여 정의할 수 있도록 지원한다.

EJB가 서버상에서 수행되는 컴포넌트이기 때문에 데이터베이스 서버 상에서 수행될 필요가 생기므로, Java에 기반하여 DB 내부에서 수행될 수 있어야 하게 된다.

EJB는 많은 종류의 서비스를 제공하는 서버의 인프라에서 효과적으로 수행되는 명확한 인터페이스를 갖는다. EJB는 Java 응용

프로그램 프로그래머에게 매우 편리하고 생산성 있는 컴포넌트 모델을 제공한다. 이 컴포넌트는 비즈니스 로직을 구현하고 있으며 코드의 재 사용성과 다중 계층 응용 프로그램 모델을 가능하게 한다.

CORBA 기반 서버 프로그램

CORBA(Common Object Request Broker Architecture)는 분산 객체 환경의 지원을 위한 구조로서 객체를 가지는 클라이언트와 서버 프로그램이 네트워크를 사이에 두고 상호간의 객체에 접근할 수 있는 인프라를 제공한다. CORBA에 기반한 서버 프로그램이 데이터베이스 상의 정보에 접근할 필요가 있을 때 CORBA 서버 프로그램이 DB 내에서 수행될 필요가 생긴다. CORBA 서버는 클라이언트와 인터넷 상에서 IIOP 프로토콜을 통해 상호 호출을 수행할 수 있다. 따라서, DB는 IIOP 프로토콜 역시 지원할 수 있어야 한다.

사례 : 오라클의 Java 지원

Oracle은 인터넷 콘텐츠를 하나의 데이터베이스 서버에서 관리하고, 응용 프로그램이 그 콘텐츠에 보다 쉽게 접근할 수 있도록 지원하기 위해 개방/표준화된 수단을 통해 작성되어지고, 응용 프로그램이 관리하기 거북한 많은 수의 클라이언트에 전개되는 것보다는 전문적으로 관리되고 확장적이고 성능이 월등한 적은 수의 애플리케이션 서버에 전개되어야 한다는 전략을 따라가고 있다.

이런 원칙 하에서 오라클은 앞서 기술한 모든 Java 지원을 제공한다. 오라클8까지는 JDBC에 기반한 Java 접근을, 오라클8부터는 SQLJ 및 DB 내장 JVM에 기반한 Java 프로그램 작성을 지원한다.

오라클은 JDBC 2.0을 지원한다. 또, 오라클은 다음과 같은 3가지의 JDBC 드라이버들을 제공한다.

- JDBC/OCI : Thick JDBC라고 불리는 것으로서 드라이버가 데이터베이스 서버 상에서 수행되며, 수행 제어는 OCI(Oracle Call Interface)를 통해 수행되어지게 된다.
- Thin JDBC : 애플릿 형태로 구성된 JDBC 드라이버로서 사용자 기계로 다운로드 되어 수행되어지게 된다.
- JDBC/KPRB : Java 저장 프로시저 내부에서 사용되어지는 JDBC 드라이버이다.

오라클은 SQLJ 1.0 표준을 지원하며, SQLJ 처리를 위한 전처리기가 제공된다. 또, Java 저장 프로시저를 지원한다.


오라클은 EJB를 위한, 확장성이 뛰어난 고성능의 실행 환경을 제공한다. 이는 EJB 1.0 스펙에 순응하며, 여러 가지 EJB의 서비스를 제공한다. Java 트랜잭션 서비스를 지원하도록 확장된 내장 JDBC 드라이버를 이용한 Java 트랜잭션 서비스를 제공한다. 오라클의 Java 트랜잭션 서비스는 EJB에 트랜잭션 속성을 부과하며 LDAP 기능이 구현된 JNDI(Java Naming and Directory Interface)를 제공한다. EJB 컴포넌트는 JNDI를 통해서 접근이 되는 디렉토리 서비스에 위치한다. 이것은 데이터베이스의 인증 기술 및 다단계의 접근 제어 방법과 함께 인터넷 표준인 SSL(Secure Socket Layer)을 이용한 강력한 보안성을 제공한다.

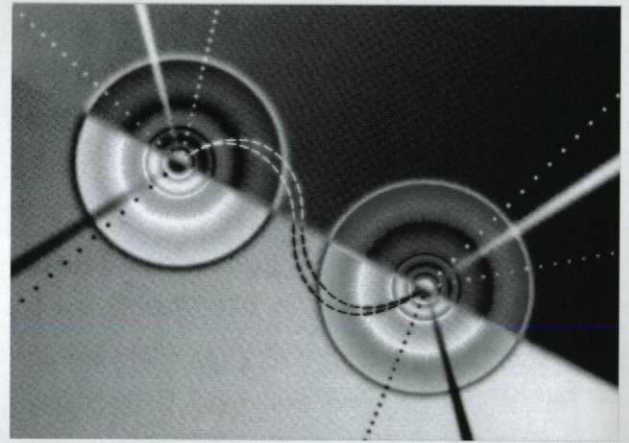
IIOP와 ORB기능을 활용하여 CORBA 서버 개발에 필요한 여러 CORBA 서비스를 제공한다.

- CORBA Services : 객체 트랜잭션 서비스를 지원하도록 확장된 내장 JDBC 드라이버를 이용한 CORBA 객체 트랜잭션 서비스를 제공한다
- Java to CORBA Service : 데이터베이스는 Java 프로그래머에게 CORBA 서비스를 쉽게 개발할 수 있도록 여러 가지의 기능을 제공한다. Caffeine을 이용하면 Java에서 IIOP로의 직접 매핑이 가능한데 이는 IDL정의의 필요성의 제거, 표준 객체의

지원 기능이 있다. 또한 java2iiop, idl2java, java2idl등의 툴의 지원 등을 통해 응용 프로그램 개발을 단순화한다.

요약 및 결론

이번호에서는 Java를 통해 데이터베이스에 있는 정보에 접근하여 활용할 수 있는 Java와 데이터베이스의 통합 방안에 대해 설명하고, 하나의 사례로서 오라클8i 상에서의 Java 지원 방안에 대해서 설명하였다. 다음호에서는 실질적으로 오라클8i를 사례로 하여 세부적인 프로그래밍 방안에 대해서 살펴 보기로 한다. 



정기구독안내

1. 구독신청방법

1. 일단, 02-725-3751(ext 118)번으로 전화하여 안내를 받으실 수 있습니다.
2. 아래의 은행구조로 구독료를 입금하신 다음 데이터베이스월드 담당자와 통화하시면 됩니다.
3. 구독자 또는 구독기관명, 구독기간, 책을 받아보실 주소, 신청인 주소와 전화번호 등을 적어서 02-725-3750번 팩스로 넣어주셔도 정기구독자로 등록됩니다.

2. 구독료 입금계좌

조흥은행 수송동지점 390-03-003978
 국민은행 세종로지점 023-25-0008-729
 ※ 예금주 : 한국DB진흥센터

3. 정기구독료

6개월 : 24,000원
 1년 : 44,000원
 2년 : 88,000원
 ※ 권당 가격은 4,000원입니다.
 ※ 정기구독을 신청하시면 편안히 책을 받아보실 수 있습니다.

재단법인 한국데이터베이스진흥센터
 110-755 서울시 종로구 수송동 146-1 이마빌딩 6층

