

UML을 이용한 CORBA기반의 분산 데이터베이스 연동의 설계 및 구현

(Implement and Desing Integration for Distributed Database which
Based on CORBA Environment with UML)

°채지연, 정은영, 박순철
전북대학교 대학원 정보통신 공학과

Ji-yon, Chae Eun-young, Jung Soon-cheol, Park
Dept. of Information and Communication Engineering, Chonbuk University

요 약

데이터의 통합과 공유 환경을 위해 탄생한 DBMS는 그 규모가 계속 커지고 있으며 형태도 복잡해지고 있다. 또한 Database가 네트워크 상에서 분산되어 있을 때, 제품이나 플랫폼이 다를 경우에는 서로간의 데이터를 공유하기가 어렵다. CORBA는 이종의 분산된 환경에서 여러 종류의 응용 프로그램들이 서로 통합될 수 있는 표준 분산 객체 기술이다. 본 논문에서는 이러한 특징을 가진 CORBA를 이용하여 여러 종류의 이질적인 데이터베이스 시스템을 논리적으로 통합시켜, 사용자가 동일한 관점에서 다양한 형태와 성질을 가진 정보를 접근할 수 있도록 해주는 것을 목표로 한다. 본 연구에서는 이러한 CORBA의 기술과 JDBC, ODBC를 이용하여 분산된 DBMS를 하나로 통합할 수 있는 환경을 구현해 보았다.

1. 서론

현재 많이 사용되어지는 DBMS는 Oracle, Cybase, Informix와 같은 관계형과 ORDBMS, O2 등과 같은 객체지향형이 있다. 그러나 같은 형의 DBMS라도 네트워크 상으로 분산되어 있거나 이종의 DBMS일 때는 서로간의 데이터를 공유하기가 어렵다. 한편 분산 시스템은 사용자에게 네트워크를 통해 연결된 모든 컴퓨터들과 그들이 갖고 있는 모든 자원들을 투명하게 사용할 수 있는 환경을 제공하는 것을 그 목표로 하고 있다.

본 논문에서는 통합이라는 용어가 자연스럽게 적용될 만한 기술에 객체지향 프로그래밍(OOP)과 분산 객체 시스템의 표준인 CORBA(Common

Object Request Broker Architecture)에 초점을 맞추어 분산 시스템과 데이터베이스의 변화에 보다 능동적으로 대처하고 효율적으로 통합 관리하는 시스템 설계를 목표로 한다.

CORBA는 네트워크 연결 및 관리에 대한 투명성, 다양한 시스템 호환성, 언어 독립성, 객체지향 기술 등의 장점을 가진 미들웨어로써 다양한 DB를 연결하여 데이터에서 최종 사용자까지의 최적의 루트 설계에 편의를 제공한다. 이러한 시스템을 설계하는 데 있어서 최적의 분석과 설계를 가능하게 하는 산업 표준 객체지향 모델링 언어인 UML(Unified Modeling Language)를 이용했다.

2. 본론

2-1 시스템 구조

CORBA는 여러 가지 기술이 혼합되어서 사용되거나 또는 여러 하드웨어에서 사용되는 클라이언트 서버 시스템을 구현할 때 유용하다.

분산환경에서 데이터베이스를 통합하는 기술로 인터페이스 부분을 CORBA 기술로 추상화시켜 클라이언트와 서버 사이의 투명성을 제공하고자 한다.

본 논문에서 제안하는 시스템의 구조를 간단하게 살펴보면, 서버는 사용자의 손쉬운 접속을 위해 웹문서를 제공한다. 사용자는 이러한 웹문서를 통해 서버에 접속하여 데이터베이스의 정보들을 선택, 추가, 삭제할 수 있는 권한을 얻는다.

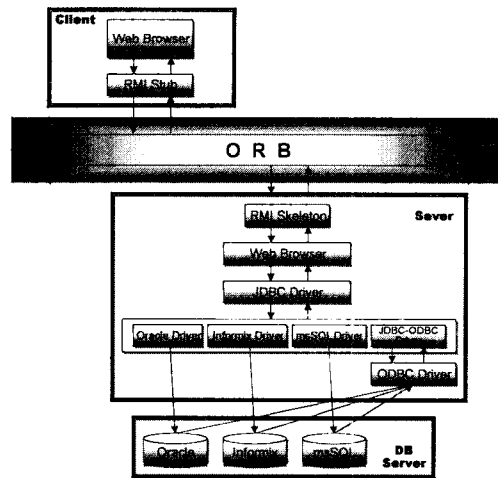
여기서 각 데이터베이스에 대한 컨트롤들을 추상화하기 위해 JDBC Driver를 이용하였다. 이를 이용하여 각종 데이터베이스 드라이버를 통합함으로써 다양한 데이터베이스들을 모두 사용할 수 있게 하였다. 본 연구에서 분산된 데이터베이스의 통합 예제로 쓰인 DBMS는 Oracle, Informix, msSQL이다.

C++로 매핑 되는 Orbix환경에서는 데이터베이스 통합에 ODBC를 이용하였고, Java로 구성되어지는 OrbixWeb환경에서는 JDBC를 사용하였다. ODBC와 JDBC의 상호 통신에는 JDBC-ODBC bridge를 이용하여 통신에 있어서 아무런 제약 없이 데이터베이스 어플리케이션들 간의 통합이 가능하게 했다.

본 연구의 시스템 전체 구조는 (그림 1)과 같다.

2-2 UML을 이용한 시스템 설계

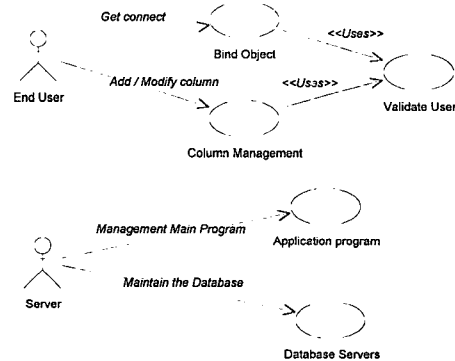
UML은 객체지향 분석과 설계를 위한 Modeling language이다. 이는 Booch, Rumbaugh, Jacobson 등의 객체지향 방법론에 관한 석학들이 내어놓은 방법론으로써 객체 기술에 관한 국제 표준화 기구인 OMG에서 이미 UML을 산업 표준 객체지향 모델링 언어로써 인정하고 있다. UML은 여러 개의



(그림 1) 시스템의 구조

Diagram으로 나뉘어 지는데, 본 연구에서는 컴퓨터 시스템과 사용자간의 상호작용을 보여주는 Use case Diagram과 객체들의 상호작용을 시간의 흐름에 따라 표현한 Sequence Diagram을 그려 시스템을 설계하였다.

먼저 (그림 2)의 Use Case Diagram을 살펴보면, 최종 사용자는 웹에서 Bind Object를 이용하여 서버에 bind한다. 이 과정에서 사용자는 인증된 사용자인지의 여부를 확인하는 과정을 거친 후 데이

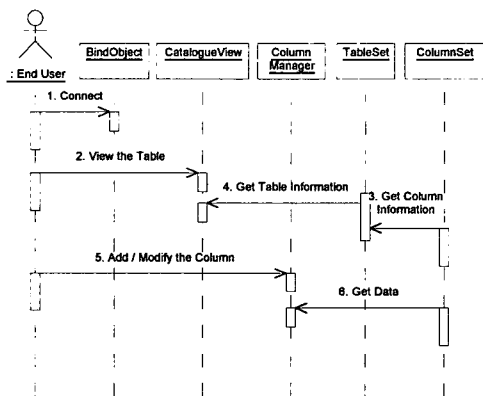


(그림 2) Use Case Diagram

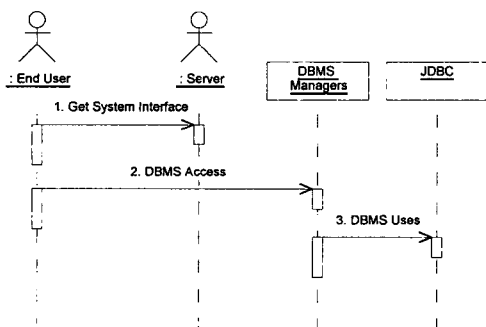
터베이스의 접근 권한을 갖는다. 권한을 가진 사용자는 Column Management를 이용하여 데이터의 삽입, 수정, 삭제의 작업을 한다. 분산된 데이터 베

이스를 하나의 인터페이스로 제공하는 서버는 데이터베이스 서버들의 정보를 가져와 하나의 애플리케이션으로 보여주는 Application program을 관리하고, 최종 사용자들이 참조하는 데이터베이스들을 가지고 있는 데이터베이스 서버들의 등록, 유지, 보수 등을 책임진다.

(그림 3)은 Sequence Diagram을 나타낸 것인데, 최종 사용자는 여러개의 객체들을 이용하여 해당 작업을 한다. 서버로의 Connect는 BindObject 객체를 이용하고, Table의 정보는 TableSet과 ColumnSet에서 가져온 Table의 정보를 CatalogView 객체를 통해 본다. 또한 Column의 Add/Modify/Delete는 ColumnSet에서 가져온 Column의 정보를 이용해 이루어진다.



(그림 3) Sequence Diagram



(그림 4) JDBC Interface Access

(그림 4)는 최종 사용자가 JDBC를 이용하여 DBMS에 연결되는 모습을 보여주는 Diagram이다. 최종 사용자는 Server에게서 Database을 접근할 수 있는 Interface를 얻어 JDBC를 이용해 해당 DBMS에 Connect한다.

2-3 시스템 구현 및 결과

사용자의 손쉬운 접속을 위해 클라이언트는 웹 환경으로 구현했고, 서버 접속 시 애플릿의 Security 문제를 해결하기 위해 Java RMI를 이용했다.

사용자는 웹을 이용해 네트워크로 분산된 시스템의 데이터를 조작(Manipulation)할 수 있다. 즉, 테이블이나 인덱스 따위의 데이터베이스 객체와 데이터베이스의 데이터를 선택, 삽입, 갱신등의 작업을 하는 데이터 조작(Data Manipulation)이 가능하다. 이러한 데이터 조작에 더하여 데이터 정의(Data Definition), 데이터에의 액세스(access)를 제어할 수 있는 데이터 제어(Data Control)등이 추가로 연구되어야 한다.

(그림 5)는 클라이언트가 웹을 이용하여 서버에



(그림 5) 웹을 이용한 클라이언트의 접속 환경

접속하는 환경을 보여주는 그림이다.

3. 결론

현재 데이터베이스들의 표준 인터페이스로 ODBC와 DAO등이 있지만, 이것은 역시 애플리케이션 개발과정에서 필요한 인터페이스일 뿐이다. 즉 인터페이스의 역할을 하더라도, 다른 종류의 데이터베이스들을 네트워크에 투명하게 적용하기 위해서는 역부족이다. 그러나 CORBA 환경을 이용한 통합 데이터베이스를 이용할 경우 이종의 시스템에서도 마치 단일 환경에서 작업하는 것처럼 데이터베이스 시스템을 이용할 수 있다. 그리고 관계형 데이터베이스에서는 어려웠던 음성, 영상정보의 저장 및 검색을 객체개념으로 취급함으로써 각각의 데이터를 이용할 수 있고, 또한 OLE통합을 통해 현재 개인 사용자가 가장 많이 사용하고 있는 Windows환경에서도 응용 애플리케이션의 개발이 용이해진다.

향후 과제로 위의 연구를 발전시켜, CORBA를 적용한 통합 데이터베이스 시스템과 현재 개발되어 있는 분산환경에서의 통합 데이터베이스 시스템과의 성능평가에 대해 연구할 예정이다.

참고문헌

- [1] 박재현 "Core CORBA", 영한 출판사, 1998
- [2] H. Onozawa "분산 오브젝트 지향기술 CORBA", 홍릉과학 출판사, 1997
- [3] Randy Otte, Paul Pratric and Mark Roy, "Under-standing CORBA", Prentice Hall PTR, 1996
- [4] Robert Orfali, Dan Harkey, "Client/Server programming with JAVA and CORBA", Wiley computer publishing, 1997
- [5] Robert Orfali, Dan Harkey, Jeri Edwards, "Instant CORBA", Wiley computer publishing, 1998
- [6] Grady Booch, Ivar Jacoson, James Rumbaugh, "Visual Modeling with Rational Rose and UML", Addison-wesley, 1998
- [7] Craig Larman, "Applying UML and Patterns", A Simon & Schuster Company, 1998

- [8] Matthew Siple, "자바 데이터베이스", 삼각형 출판사, 1998
- [9] IONA, "OrbixWeb3.0 Programming Guide", IONA Technologies Ltd.
- [10] IONA, "OrbixWeb3.0 Reference Guide", IONA Technologies Ltd.
- [11] Gary Cornell, Cay S. Horstmann, "Core JAVA", 영한 출판사, 1997