

CORBA 를 이용한 STEP 데이터베이스의 검색 Retrieving STEP Database through CORBA

김 준환(한국과학기술원), 한 순홍(한국과학기술원)

J.H.Kim(KAIST), S.H.Han(KAIST)

요 약

기업 조직의 확대와 세분화, 제품의 다양화 및 전문화에 따라, 정보 공유와 상호간의 원활한 정보 교환이 필수 요구조건이 되고 있다. 그러나 이들 정보들이 서로 다른 형식과 구조, 서로 다른 기종의 컴퓨터 환경과 각기 다른 데이터 저장 방식 또는 운용 소프트웨어, 네트워크 구조 등의 문제로 인해, 문서 교환에 의해 상호간 정보를 공유하는 것이 현재의 실정이다. 이 글에서는, CORBA 라는 분산객체 공유표준과 연계되고, 제품 데이터의 표준인 STEP 과, Oracle 데이터베이스, 인터넷을 기반으로 한 PDM 의 구축을 소개한다.

Key Words: STEP, CORBA, 데이터 베이스, 인터넷, 자바 (JAVA)

1. 서론

1.1 연구의 배경

오늘날 인터넷의 활용이 일반화 되어 가고, 특히 웹은 네트워크 사용자가 전세계의 분산된 정보를 쉽고 빠르게 접근할 수 있도록 효과적인 정보 공유의 형태를 제공한다. 컴퓨터와 네트워크의 발달로 기업에서 인터넷을 통한 정보의 공유는 일반적인 현상이 되가고 있다. 제조회사에서는 제품을 설계하고 제조하는데 서로다른 기종의 컴퓨터 환경에서 인터넷을 이용하여 제품정보 데이터를 얻을수 있고, 또 그 데이터들을 실시간에 생성, 수정 할 수 있게 함으로써 생산성을 향상 시킬 수 있다.

기업활동은 조직의 확대와 세분화, 제품의 다양화 및 전문화에 따라, 각 부서간의 정보 공유는 물론 협력 업체, 부품 공급 업체 상호간의 원활한 정보 교환이 필수 요구조건이 되고 있다. 그러나 이들 정보들이 서로 다른 형식과 구조, 서로 다른 기종의 컴퓨터 환경과 각기 다른 데이터 저장 방식 또는 운용 소프트웨어, 네트워크 구조 문제로 인해, 문서 교환에 의해 상호간 정보를 공유하는 것이 현재의 실정이다. 또한 정보의 양이 많고 복잡하여, 정보관

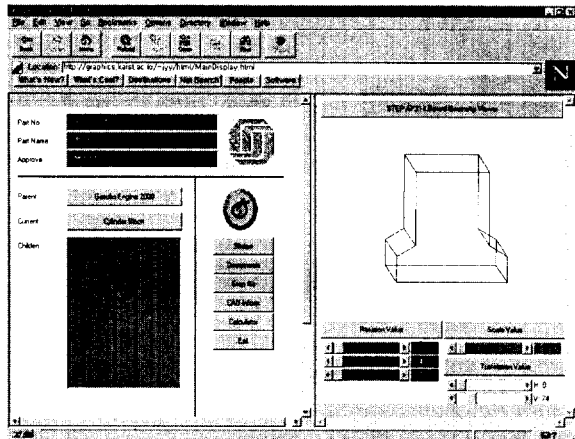
리에 있어 데이터베이스와의 연계는 필수적이다.

서로 다른 형식과 구조의 데이터의 교환문제를 해결하기 위해 미국을 비롯한 일본 및 선진 업체들은 CALS (Commerce At Light Speed), 즉 광속의 상거래라는 구상하에 제품 전체 수명주기 동안의 전과정에서 발생하는 정보를 표준화하여 정보네트워크를 통해 공유하려고 추진하고 있다. 특히 제품정보의 공유의 방안으로 제품정보를 어느 시스템에서나 인식이 가능하도록 중립형태의 표준 (Neutral format standard)으로 표현해 네트워크를 통해 전송하는 다양한 중립형태의 표준 중 STEP(Standard for the Exchange of Product Model Data)에 관한 연구가 활발히 진행중이다. STEP은 국제표준화기구 (ISO)에서 제정작업중인 제품정보의 교환에 관한 국제 표준이며 CALS에서도 이를 채택하고 있다.

서로 다른 컴퓨터 하드웨어나 운영체제에 상관없이 작동하는 생산성 있는 응용 프로그램을 개발하고, 파일이나 데이터베이스 처럼 다양한 데이터 저장공간으로 부터 정보를 얻고, 또 다른 사용자들과 자료를 공유하고 서로 통신할 수 있는 환경이 필요하다. 이를 극복하기 위한 기술로 OMG (Object Management Group)에서 제정한 이종의 분산된 환

경 하에서 응용 프로그램들을 서로 통합 할 수 있는 국제 표준인 CORBA (Common Object Request Broker Architecture) 가 있다.

STEP 표준을 이용하여 제품 데이터를 생성하고 CORBA 라는 분산객체 공유표준과 연계된 PDM (Product Data Management) 을 사용하여 위에서 나열한 정보공유에 있어서의 어려움을 극복 할 수 있다. [그림 1] 은 인터넷상에서 JAVA 언어를 이용하여 자동차 PDM 을 구현한 예를 보여준다.



[그림 1] 인터넷에서 JAVA 를 이용한 PDM

1.2 연구의 목표

데이터 교환에 있어서 데이터들이 서로다른 구조와 형식을 가지는 것을 해결하는 방안으로 STEP 을 채택하고, 복잡하고 많은 양의 데이터를 데이터의 중복성과 이질성을 극복하기 위해 데이터베이스와 연계가 필요하다. 서로 다른 컴퓨터 환경에서의 데이터 교환 응용 프로그램 개발을 위해 CORBA 를 채택하고, 구현된 시스템을 인터넷에서 이용하기 위해서 JAVA 를 활용한다. 즉 제품정보의 표현은 중립 형태 표준으로 새롭게 제시되고 있는 STEP 표준을 따르며, 분산 객체의 표준은 CORBA 를 따르고, 이를 자동차의 PDM 을 대상으로 적용하고자 한다.

본 연구는 인터넷상에서 STEP 파일을 데이터베이스에 저장하고, 그 정보를 분산 환경에서 이용할 수 있는 응용 프로그램을 개발하고, 그 응용 프로그램을 인터넷상에서 실행 할 수 있도록 구축 하는것을 목적으로 한다

2. STEP 과 인터넷 기술

2.1 STEP 의 개요

STEP 은 1984 년부터 국제표준화기구(ISO) 산하 기술위원회인 TC184 의 소위원회 SC4 를 중심으로 제정작업중인 국제 표준이다. STEP 은 제품의 형상 정보 뿐만 아니라 전 수명 주기에 걸친 모든 데이터를 포함하는 제품 정보를, 어느 시스템에서나 사용할 수 있도록 하는 중립 형태 표준이라는 점이 특징이며, 현재 가장 널리 사용되고 있는 중립 형태 표준인 IGES (Initial Graphics Exchange Specification)의 역할을 대체할 것이다.

STEP 은 여러개의 파트들로 구성되어 있다. 파트들은 크게 공통자원과 응용 프로토콜로 나누어진다. 공통자원에는 제품 정보를 표현하기 위한 서술방법으로 EXPRESS 에 관한 것이 10 번대에 있고, 구현 방법으로 물리적 파일 (Physical file)과 SDAI 에 관한 것이 20 번대에 있고, 통합자원이 40 번대와 100 번대에 있고, AIC (Application Interpreted Construct)가 500 번대에 있다. 산업 현장에서의 실제적인 적용을 위한 각 산업별 응용프로토콜 (AP : Application Protocol)은 통합자원을 마치 조립 블럭처럼 활용하여 구성된다. STEP 의 전체 구조는 [그림 3]과 같다. 이들중 파트 1, 11, 21, 31, 41, 42, 43, 44, 46, 101, 201, 203 은 1994 년 12 월에 국제 표준으로 확정되었고, 105 와 202 가 1996 년에 국제 표준이 되었다.

2.2 JAVA

JAVA 라는 프로그래밍 언어는 선 마이크로 시스템의 James Gosling 박사에 의하여 창안되었다. 안정적이면서도 작은 크기를 가지며 어떠한 컴퓨터 칩에서도 실행이 가능한 새로운 언어이다. JAVA 는 Netscape 와 같은 Web Browser 없이도 일반 프로그램 처럼 컴퓨터 자체 내에서 독자적으로 작동될 수 있다. 이렇게 작동되는 경우를 Application 이라고 하며, JAVA 를 지원하는 Web Browser 또는 Applet Viewer 상에서 작동되는 경우를 Applet 이라고 한다.

2.3 PDM

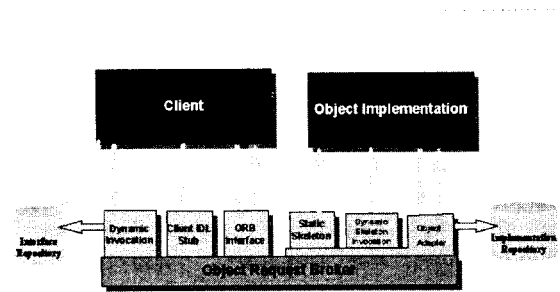
PDM 은 제품 전주기에 걸쳐 제품과 관련된 데이터를 관리하고 정보에 대한 추적성을 제공하며, 제품 개발 프로세스를 관리하는 제품 데이터 관리 체계이다. 신속하고 정확한 제품 정보 활용 체계를 갖춘

으로써 제품이 시장에 출하되기까지의 시간을 단축 하고자 하는 목적을 가지고 있다.

3. CORBA 를 이용한 응용 프로그램 개발

3.1 CORBA 의 소개

컴퓨터와 통신 기술의 발달로 분산시스템의 활용이 다방면에서 보편화 되어 가고 있다. 이들 분산 시스템은 대형 컴퓨터를 이용하지 않고 유닉스 워크스테이션에 일을 적절히 분배하여 활용함으로써 많은 효과를 얻을 수 있으며, 시스템의 신뢰도를 향상시킬 수 있다는 장점이 있다.



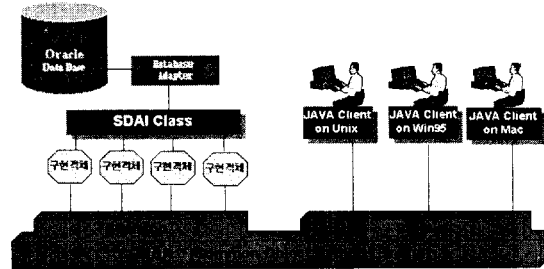
[그림 2] CORBA 의 구조

이러한 분산 시스템을 구성하는 대표적인 기술적 방법이 클라이언트/서버 기술이다. 서비스 의뢰자인 클라이언트는 서비스 제공자인 서버에게 원하는 작업을 요청하고 클라이언트는 서버에서 처리된 작업 결과만을 얻는 기술을 말한다.

특히 분산 소프트웨어의 오류의 상당수가 각 모듈 또는 프로세스 들간의 통신 및 동기화에 관련된 것들로써, 이러한 문제점들이 분산 시스템의 효과적인 활용에 걸림돌이 되어왔다. 이 문제에 대한 해결 방안으로 대두된 것이 OMG 에서 제정한 CORBA 표준안이다.

이제 까지 대부분의 시스템들은 중앙집중식 이었다. 이 방식은 시스템 중앙에 있는 컴퓨터에 접속하여 모든 작업을 호스트 중심으로 수행하였다. 그러나, 분산 시스템은 대형 컴퓨터를 이용하지 않고, 클라이언트가 서버에게 원하는 작업을 요청하면 클라이언트는 서버에서 처리된 작업 결과만을 얻는 기술이다.

CORBA ORB 를 사용하여 클라이언트 객체는 동일 머신 또는 네트워크 상에 존재하는 서버 객체의 메소드를 투명하게 호출할 수 있다. 즉, 클라이언트 객체는 요청을 실행 할 수 있는 객체, 즉 서버객체가 존재하는 장소, 프로그래밍 언어, 운영체제, 또는 객체의 인터페이스 부분이 아닌 다른 시스템 사양을 전혀 알고 있지 않더라도, ORB 가 제공하는 객체 서비스를 통하여 요청을 실현할 수 있다. 한편, CORBA ORB 상에 존재하는 객체들은 경우에 따라 클라이언트 또는 서버로서 동작 할 수 있다.



[그림 3] 클라이언트/서버의 구조

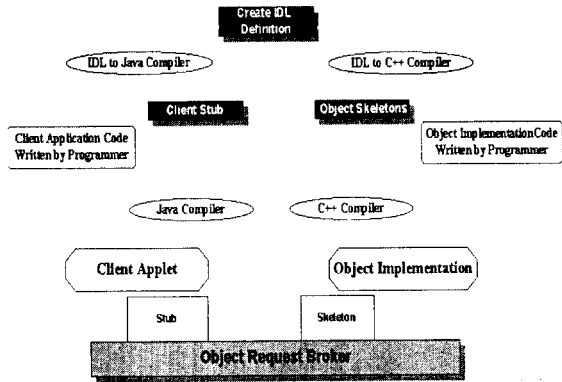
본 연구에서는 데이터 모델의 표준으로 ISO 의 STEP 을 이용하고 분산 객체 인터페이스의 표준으로는 OMG 의 CORBA 를 이용한다. 본 연구에서 제안하는 분산 통합 시스템의 전체 구조는 다음과 같다. 분산 통합 시스템의 구조는 인터넷 웹 브라우저 상에서 JAVA 클라이언트를 통하여 데이터 베이스에 저장된 STEP 데이터를 서버에 있는 객체구현을 실행하여 검색 할 수 있다.

3.2 CORBA 를 이용한 분산 소프트웨어의 개발 과정

CORBA 를 이용하여 소프트웨어를 개발하는 과정은 그림 4 에 보인 바와 같다.

- 단계 1: IDL (Interface Definition Language)을 설계 하고, IDL 을 사용하여 객체 클래스를 정의한다.
- 단계 2: IDL 언어의 skeleton 과 stub 을 얻는다.
- 단계 3: 단계 2 에서 얻어진 skeleton 에 객체 구현 코드를 첨가하여 객체 구현을 완성한다.
- 단계 4: 생성된 stub 코드를 이용하여 Client 를 구현 한다.
- 단계 5: 클래스의 정의를 인터페이스 저장소와 binding 한다.

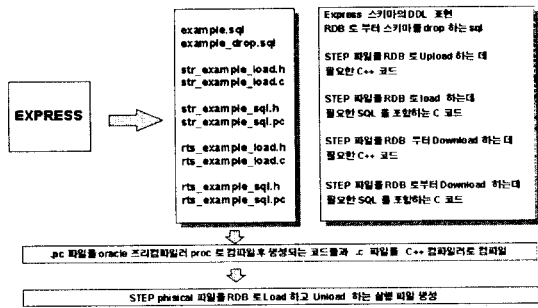
단계 6: 객체 구현 클래스로부터 객체를 생성한다.
 단계 7: 단계 6 에서 생성된 객체를 구현 저장소에 등록한다.



[그림 4] CORBA 응용 프로그램 개발 과정

4. CORBA 를 이용한 STEP DB 검색 시스템 구현

4.1 ST-ORACLE 과 테이블 구조

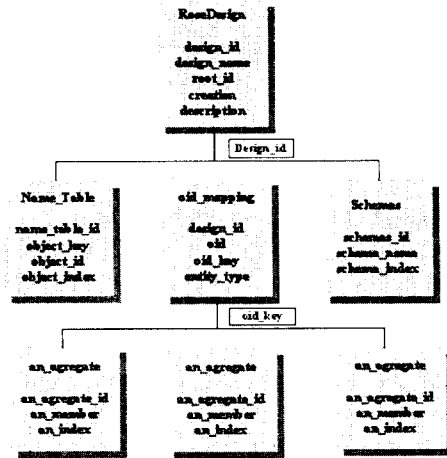


[그림 5] STEP 파일의 RDB 로의 Load 와 Unload

ST-Oracle 은 STEP 데이터를 관계형 데이터베이스 시스템인 Oracle 상으로 옮겨 관리하기 위한 STEP 과 Oracle 사이의 인터페이스 툴이다. 여기에 포함된 EXPRESS to RDB 변환기는 [그림 5]와 같이 EXPRESS 스키마로부터 DDL (Database Definition Language) 문장과 STEP 정보 모델을 데이터베이스로부터 읽기(Upload), 쓰기(Download) 위한 코드들을 생성해준다.

Oracle 은 전형적인 관계형 데이터베이스로 본 연구에서 STEP 데이터의 저장과 추출을 위한 데이터베이스로 사용된다. Oracle 은 데이터 조작언어로 SQL

(Structured Query Language)을 채택하고 있으며, Oracle 사용자 프로그램의 하나인 SQL*Plus 는 사용자 대화형 SQL 의 실행을 가능하게 해주고 있다.



[그림 6] 관계형데이터의 스키마

EXPRESS 는 객체 지향형의 서술 구조를 가지며 인터페이스 과정에서 발생하는 STEP 데이터도 EXPRESS 의 특성을 물려 받게 된다. 그러나, Oracle 은 전형적인 관계형 데이터베이스로서 EXPRESS 가 제공하는 다양한 형태의 데이터 구조를 지원하지 않으며, STEP 데이터의 저장시에 별도의 처리과정을 포함해야 한다. 처리과정의 주요 관점은 데이터의 저장을 위한 테이블의 구조이다. [그림 6]는 EXPRESS 파일이 어떻게 RDB 데이터에서 저장되는가 하는 모습을 보여준다. 상위 오브젝트의 정보 모델을 저장하기 위해서 각각의 테이블들은 상위 오브젝트의 인식자(ID)를 갖게 된다.

4.2 시스템의 구조와 개발 환경

시스템의 개발환경은 다음과 같다.

CORBA 의 구현객체;

Language	: C++
OS	: Sun Solaris
CORBA 제품	: Visibroker for C++
Database System	: Oracle 7.3.2

CORBA 의 Client;

Language	: JAVA
OS	: Sun Solaris
CORBA 제품	: Visibroker for Java

IDL을 컴파일하여 Stub 코드와 Skeleton 코드가 나오는데 이는 각각 클라이언트와 서버를 구현하는데 쓰여지며, 이 코드들에는 프로그래머가 수정을 가할 필요가 없다. 코드들이 생성되면 CORBA IDL로부터 얻은 Skeleton 코드를 기반으로 하여 ST-Oracle에서 생성된 코드들을 이용하여 CORBA의 객체를 구현한다.

CORBA IDL 컴파일러로부터 생성된 Skeleton 코드는 클라이언트와의 통신 부분을 담당하는 코드이다. ST-ORACLE에서 생성하여 주는 코드와 CORBA의 Skeleton 코드를 이용하여 CORBA의 구현 객체를 만들었다. 클라이언트 부분도 역시 기존의 프로그램을 CORBA의 객체로 만드는 작업을 수행한다. 웹 상에서 STEP 물리적 파일을 읽어서 정보를 출력하는 기존의 애플리케이션을, CORBA의 구현객체로부터 데이터를 받아서 출력시키도록 입력부분을 수정하여 CORBA의 클라이언트로 만들었다.

5. 결론

STEP 파일을 DB에 저장하기 위해 EXPRESS 파일을 RDB 스키마로 변환하였고, STEP표준을 이용하여 제품 정보를 DB에 저장하였다. 그리고 서로 다른 컴퓨터 하드웨어, 운영체제에 관계 없이 작동이 가능한 소프트웨어 개발 환경인 CORBA를 이용하여 응용 프로그램 작성하였다. 클라이언트는 JAVA를 이용하여 구현하여 인터넷의 웹 상에서 데이터베이스 정보를 검색 수 있도록 하여 자동차 PDM에 적용 시켜 보았다.

추후에 연구되어야 할 사항으로는 STEP의 PART 26 (SDAI의 IDL 바인딩)을 이용하여 IDL 설계에서부터 STEP의 EXPRESS의 정의와 연관되어 작업을 수행하는 것이다.

참고 문헌

1. Oracle Corporation, SQL Language Reference Manual, 1990
2. STEP Tools Inc., "SDAI Library Reference Manual", 1993
3. 김기환, "표준 생산 데이터베이스 인터페이스 구현", 인하대학교 석사학위논문, 1995년 2월
4. 유상봉, 서효원, "STEP을 이용한 통합 제품 정보 모델 (IPIM) 개발", 산업공학회지, Vol 21,

No3, 1995년 9월

5. 김태식, "STEP 표준 AP203을 이용한 제품 설계 정보 시스템", 한국과학기술원 석사학위논문, 1996년 2월
6. 권용국, "STEP 데이터 인터페이스를 위한 다중 데이터베이스 시스템 구현", 인하대학교 석사학위논문, 1996년 2월
7. 한순홍, 배두환, 신용재, 외 5인, 연구주관기관 한국과학기술원, "STEP 표준을 이용한 조선소 CAD 정보의 초고속 전송", 정보통신부 1996년 4월
8. "분산객체 시스템의 표준, CORBA를 정복하자" 마이크로 소프트웨어 1996년 4월
9. STEP 연구회, "제 14회 STEP 연구회", 1996년 5월
10. 한순홍, 서효원, 유상봉, 외 6인, "STEP 적용을 위한 연구", 생산 기술 연구원 1996년 11월
11. Jon Siegel, "CORBA", John Wiley & Sons, Inc., 1996
12. Robert Orfali 외, "The Essential Distributed Objects Survival Guide", John Wiley & Sons, Inc., 1996
13. STEP Tools Inc, "ST-Oracle Reference Manual", John Wiley & Sons Inc., 1996
14. STEP Mosaic
<http://www.rdr.rcpi.edu/niiip/NSF96.iita.html>
15. 정운용, "STEP AP214에 근거한 자동차 PDM의 구현 방안", 한국과학기술원 석사학위논문, 1997년 2월
16. 현석진, "CORBA를 이용한 PDM 환경에서의 이질적인 데이터 및 서비스 객체의 통합", 포항공과대학교 석사학위논문, 1997년 2월
17. 오유천, "인터넷에서 3차원 STEP 형상정보의 가시화", 한국과학기술원 석사학위논문, 1997년 2월
18. Visibroker for C++ Programmer's Guide,
http://www.visigenic.com/techpubs/vbroker++/prog_gd/startme.htm
19. CStar Project
<http://bikini.scra.org:8080/cstar2/noframe.htm>
20. STEP 연구회, "STEP - 제품 모델 정보 교환을 위한 국제 표준", 성안당, 1996년 9월