

웹 기반 컴포넌트 뷰에 관한 연구

정대성*, 임종혁*, 한정수*

*천안대학교 정보통신학부

e-mail: fl5cc, jshan, bestofbest99@cheonan.ac.kr

A Study of Web based Component View

Jone-Hyuk Lim*, Dae-Sung Jung*, Jung-Soo Han*,

* Division of Information and Communication, Cheonan University

요 약

CBD 개발 방법론의 발전으로 많은 컴포넌트가 개발되면서 소프트웨어의 빠른 기술변화와 더불어 상용컴포넌트 사용이 활성화되고 컴포넌트의 재사용이 빈번해 지면서 UML 이 각광받고 있다.. 본 논문에서는 웹상에서 컴포넌트의 클래스 다이어그램과 명세를 보여주는 뷰(View)를 구현하였고, 이를 위해 각 정보를 DB에 분류 저장하여 애플릿으로 구현된 뷰어에서 검색하여 사용할 수 있도록 하였다.

1. 서론

소프트웨어 개발방법이 상용컴포넌트, 표준 컴포넌트 등을 사용하면서 새로운 패러다임으로 진행되고 있다. 컴포넌트 개발 방법론은 재사용이 가능하기 때문에, 기존보다 빠른 개발을 할 수 있는 장점을 가진다.[1][2] 컴포넌트 기반 개발 방법을 사용하면서 UML의 사용이 빈번하게 대두되면서, 각 컴포넌트의 클래스 다이어그램, 시퀀스 다이어그램 등을 웹에서 능동적으로 보여줄 수 있고 필요에 따라서 해당 다이어그램을 다운로드 받을 수 있는 프로그램 개발을 유도하고자 한다.

본 논문은 자바 애플릿을 이용하여 웹 뷰어를 개발 하였고 MySQL을 이용하여 데이터베이스를 구축하였고 자바 JDBC를 이용하여 데이터베이스를 연동하였다.

2. 컴포넌트 뷰

2.1 시스템 구조

개발시스템 운영체제는 윈도우2000 플랫폼을 이용하였으며, UML 툴로는 투게더(Together 6.0)을 이용하였고 개발언어는 자바 1.3.1 를 이용하였다. <그림1은> 시스템에서 컴포넌트 접속시 어플리케이션 에서 제공되는 뷰(View) 포인터를 통하여 해당 컴포넌트의 클래스 다이어그램과 소스(Source)를 보여줌으로써, 해당 컴포넌트 정보를 보다 상세하게 알 수 있다. 뷰는 컴포넌트 다이어그램과 소스를 보여주게 되며, 컴포넌트와 관련된 정보는 데이터베이스를 이용한다. 데이터베이스는 이와 관련된 저장된 객체이름과 객체 데이터형태 정보를 담고 있어야 하며, 저장 객체에는 파일이름과 버전, 정의

등이 저장되어야하고, 데이터 타입에는 데이터 형태가 저장된다.

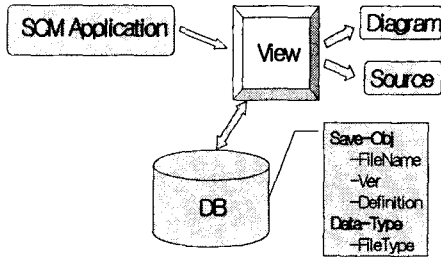


그림 1. 시스템 구조

2.2 뷰어(Viewer)

데이터베이스 테이블 내 각각 필드에서 파일명은 버전으로도 사용할 수 있으며, 정의는 컴포넌트 이름(기능) 등을 수반하게 된다. <그림 2> 는 이를 기반으로 한 DB Table구조를 보여준다. 데이터타입에서 파일타입은 각 컴포넌트의 파일형태를 의미하며 파일 형식뿐만 아닌 개발된 언어를 포함할 수 있다.

```
<?xml version="1.0" encoding="EUC-KR"?>
<!DOCTYPE Component [
  <!ELEMENT Component (Save_Obj, Data_type) >
  <!ELEMENT Save_obj (File_Name, Ver, Definition*)>
  <!ELEMENT File_name (#PCDATA)>
  <!ELEMENT Ver (#PCDATA)>
  <!ELEMENT Definition (#PCDATA)>
  <!ELEMENT Data_type (#PCDATA)>
  <!ELEMENT File_type (#PCDATA)> ]>
```

그림 2. DB 저장구조 DTD

```
try{
  Class.forName("org.gjt.mm.mysql.Driver").
  newInstance();
}
catch(Exception e) {
  e.printStackTrace();
}
```

```
String dbURL=
"jdbc:mysql://localhost/java?user=root&pass
word=1012";
Connection conn =
DriverManager.getConnection(dbURL);
System.out.println("Connencted Database
Server");
```

그림 3. JDBC Driver 로딩

시스템에 사용된 데이터베이스는 MySQL로 자바의 JDBC를 사용하여 연동하였고, <그림 5> 는 MySQL의 JDBC Driver를 로딩하기 위한 코드이다. 그림에서는 클래스 Class의 forName()메소드를 이용하여 드라이버를 직접 선택하여 로딩하고 JDBC 에서는 데이터베이스를 사용하기 위해 드라이버를 제공, 이렇게 로딩된 데이터베이스는 <그림 3> 과같이 Connection 처리를 해줌으로써 데이터베이스에 접근하여 자원을 이용할 수가 있게 된다.[3] 그림에서는 지정된 User의 아이디와 패스워드로 Connection 객체를 생성 해주고 있다. 이렇게 생성된Connection 객체를 통해서 접속이 이루어지면 모든 작업들은 이미 생성된 객체를 지속적으로 사용할 수 있다.

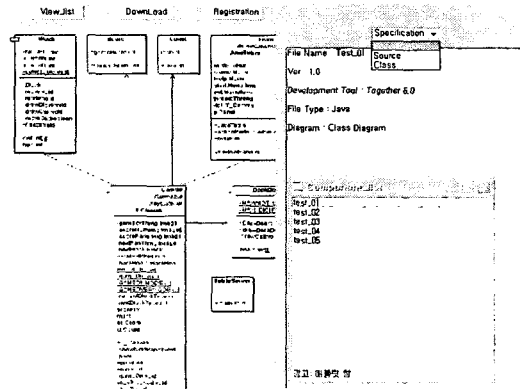


그림 4. 프로그램 구동

<그림4> 는 구현된 애플릿 화면을 보여주고 있다. 애플릿 화면을 통해 웹상에서 데이터베이스에 접근, 원하는 컴포넌트를 검색하여 사용할 수 있으며, View_list 는 DB에 저장된 컴포넌트 목록을 보여주고, Download는 원하는 컴포넌트를 다운을 받아 사용할 수 있도록 해주고 있고, Registration은 컴포넌트를 등록 기능을 수행하며, 컴포넌트를 선택했을 경우 Choice 버튼을 이용하여 Specification, Source, Class중에 하나를 선택하면, 관련 정보를 텍스트 필드 창에 나타내 준다.

그림 5. 컴포넌트 등록

컴포넌트 등록 예시는 <그림5> 와 같은 방법으로 등록이 이루어진다. 이는 Registration 수행시 출력되는 화면이며, 이때 컴포넌트 등록에 명시되는 내용으로는, 작성자, 제목, 파일명, 버전, 그리고 컴포넌트 내용등이 기술되며, 일반적인 게시판과 같은 형식의 등록을 예제로 들었다.

컴포넌트 등록은 데이터베이스에 저장소에 직접적으로 기록되는 부분으로써 sql 문을 이용하여 데이터베이스에 저장된다. 화면은 컴포넌트 정보를 입력받고 컴포넌트를 입력받아서 넘겨주는 창을 보여주며, Submit 버튼을 이용하

여 쿼리를 서버에 넘겨준다.[4]

3. 결론

본 논문에서는 컴포넌트 기반 개발방법을 보다 효율적으로 운용하기 위해 기존의 오프라인 방식을 탈피하여 웹상에서, 컴포넌트 조립시 웹상에 등록된 컴포넌트를 보다 적절하고 쉽게 시스템에 적용시키기 위한 방법을 제안하였고, 데이터베이스에 등록된 컴포넌트를 웹상에서 애플릿을 이용하여 컴포넌트 정보를 다이어그램으로 보여주고, 관련된 명세를 함께 보여줌으로써, 컴포넌트 정보를 보다 정확하게 전달하여 시스템에 보다 적절한 컴포넌트를 선택하고 선택된 컴포넌트를 다운로드 받아 사용할 수 있도록 하였다.

참고문헌

- [1] Ivica Crnkovic, Peter Funk, Magnus Larsson, "Processing Requirements by Software Configuration Management", In euromicro 99, proceedings of the 25th EUROMICRO conference Milano, Italy, Sep 1999.
- [2] Magnus Larsson, Ivica Crnkovic, "Component Configuration Management", In ECOOP Conference, Workshop on Component Oriented Programming Nice, France, June 2000.
- [3] Li Gong, "Java Security Architecture(JDK1.2), Sun Microsystems, 1997.
- [4]"Designing Codasyl database programs using JSP" School of Computing and Informatics, Newcastle-upon-Tyne Polytechnic, Newcastle-upon-Tyne NE1 8ST, UK Available online 10 June 2003