

Design Approaches of Web Application with Efficient Performance in JAVA

OhSoo Kwon and HyeJa Bang

Dept of Computer Science & Engineering,
Seoul National University of Science and Technology

Abstract

System implementation using Client/Server based on Web by spreading fast is becoming common. This kind of Web environment introduction solved many problems such as the dependency and distribution difficulty, high cost, etc. Web application and Web Service technology has more closely and effectively developed to maximize the positive point of Web environment. Based on the positive point, in this paper, we propose and implement Web application design method to implement it effectively. And also, we need to make it possible to implement on any platform based on Java. First, we have implemented reusable SQL Servlet DATABASE search, registration, update, delete, etc fast and convenient using SQL Servlet for only SQL service. Second, it writes business logic to the JSP file with Forward Servlet to prevent restarting the Web server, so it can prevent restarting the server when compiling from adding or changing Java file. These two methods provide easiness, and shortening the developing period, scalability and flexibility have improved, and it is very effective way for the maintenance.

Key words:

JAVA, Web Application

1. Introduction

Its early computational system had only one function which user device can access to the system in the way of connecting to the main frame terminal. Since then, the lowered price and advanced PC has come into wide use and Client/Server method appeared by developing communication technology. Since the advent of Web, many companies have chosen Web based method which can interchange not only text data in standardized environment, but voice, image, movie and multimedia. And all sort of server side languages, JavaScript and Flash made web page dynamically and it made various services possible. This kind of server based service is called Web Application defined by "Java Servlet Specification 2.3" which makes interface of online application made out of server source set. Server sources include Servlets, JSP, HTML page, Server Side Classes, etc.[1]

It is very effective for the maintenance by utilizing the server sources. We need to study and implement on effective Web Application which provides convenience

2. Design

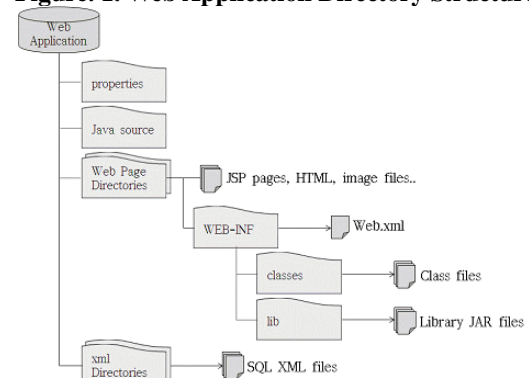
In this paper, the Web Application what we wanted to build was designed based on Web environment and was used many different open library based on Java. This library classes help us use various functions easily and conveniently. The used libraries were Struts, ibatis, Ajax, etc And also, the purpose of this paper is to implement using the C/S model's good point and offering similar UI and functionality compare to C/S model aimed at the system which replacing existing 2-Tier C/S model environment system to 3-Tier model Web based system. And it was designed the purpose of convenient development, reduction of development and maintenance easiness.[2]

And also, this paper proposes SQL Servlet for only SQL service and Forward Servlet for the maintenance and easiness.

2.1 Directory Structure

The actual Basic structure of the Web Application and sources includes static source directory, JSP file and directory, applications and WEB-INF directory of application related class file, subdirectory servlet files and related class files, JavaBeans class directory and lib directory of library files which is used in Web application. The actual Web Application directory location and structure are as follow.[1]

Figure. 1. Web Application Directory Structure



The descriptions of Web Application directory structure is as follow.

- Properties

It is an environment set file toward Web Application configuration. Basically, Web Application setting was set as default.

- Source

It is a directory for business logic Java files which will be used for Web Application.

- Web Page

It stores HTML, image, JavaScript file, etc to the base Doc directory of Web Application.

- WEB-INF

This directory can't be accessed with URL and includes every implemented application such as Servlet, JSP and class files which use those applications.

- Classes

This includes Servlet class files and related classes, JavaBean, etc. Even if this directory was not set to the classpath category, it registers to the container's classpath internally. But if it places servlet to a subdirectory then this servlet should have the directory name as the package name.

- lib

It is a directory which stores library files used in Web Application, and the file in this directory added automatically and internally by the container. The library in this directory should be in compressed format like jar, zip.

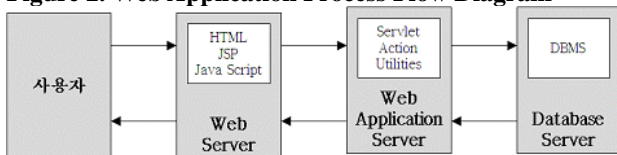
- XML

It is a directory which stores SQL DML statement used in Web Application written in XML format. It doesn't use SQL DML statement directly on Java code or JSP code, and it uses by reading DML statement written in XML.

2.2 Web Application Whole Process

The whole process of Web Application requests user's request to the servlet through Ajax engine, and at the same time, Servlet processes the logic, and then it responds with user's XML data. The whole Web Application process looks as follows.

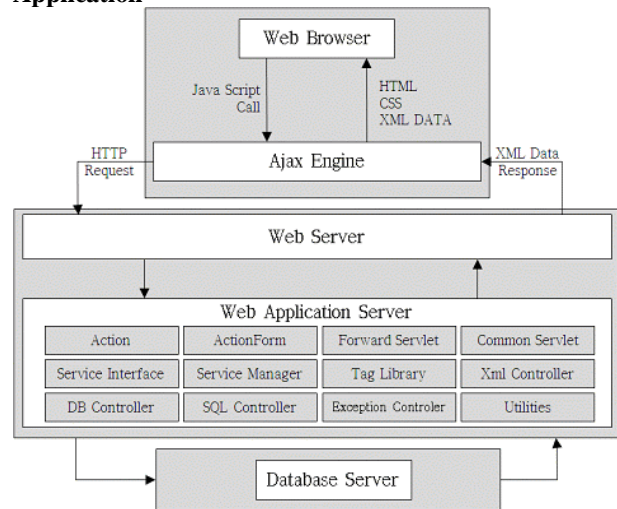
Figure 2. Web Application Process Flow Diagram



If we take a look at the Web Application process steps like the figure, it could be called JavaScript function by user's request. The called JavaScript function will request HTTP to the Web Server through Ajax Engine. At this time, JavaScript function will decide Parameter and Service type

which are needed for the request and Action Servlet which will be requested. If web server receives HTTP request, it forwards the request to the Web Application. The Web Application server processes the business logic based on received HTTP request. When business logic is in progress, it requests and send data to database server in case, As soon as process is done, it send the result to the web server in XML format. Web server forwards the received and processed result to the web browser through the Ajax Engine. Web browser forwards JavaScript function called Callback by processing received XML Data. The whole flow diagram shows as follows.

Figure 3. The whole process flow diagram of Web Application



2.3 Web Application client process flow diagram

First, client requests for the GUI for the presentation. This screen has used expanded JSTL Custom Tag Library. Every request is done by JavaScript function, and the JavaScript function checks the parameter and validation about the request. The request from the JavaScript function forwards to Web server through Ajax Engine, and the Web Server forwards the result to Client in XML Format. Forwarded XML Data parses with defined callback function, and the parsed XML displays on the screen.[12]. The client process flow diagram shows as follows.

2.4 Web Application server process flow diagram

First, the server loads the Web Application at the beginning, and then loaded Web Application will be loaded on the memory by loading stored code (codes used for the Web Application) in properties file and database to set the Web Application by LOAD class. As soon as the Web Server Client request arrives, if database use is needed after handling requested actions, Servlet will return the result in XML format after reading and processing SQL statements

from XML directory using XML Controller.

Figure 4. Web Application Client process flow diagram

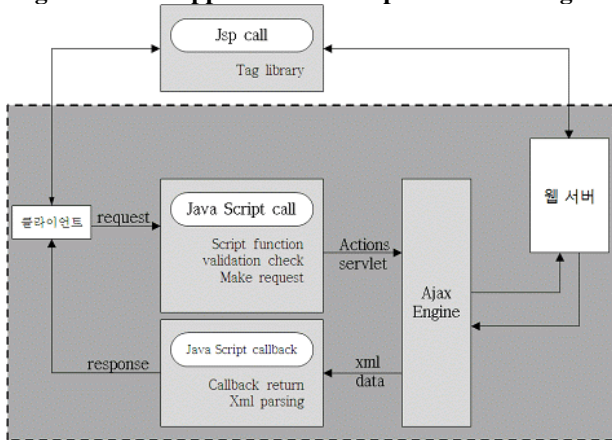
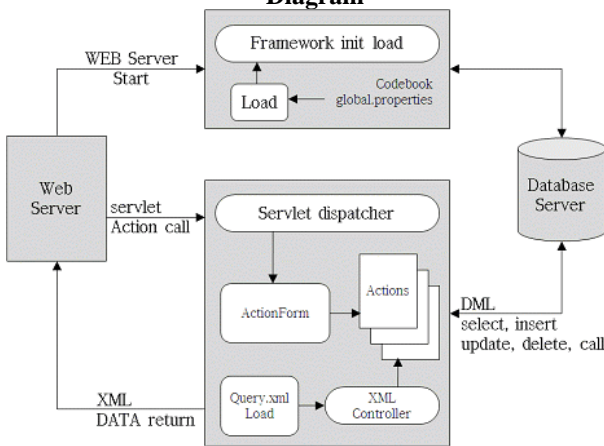


Figure 5. Web Application Web Server Process Flow Diagram



2.5 Web Application Compile Process Flow Diagram

We Application compile process uses ANT Build. ANT is a build tool based on JAVA offered by Apache. It is possible to use on any platform due to the platform independency if ANT is used. For example, compiled classes create jar compressed file by packaging, and have functions which moves jar compressed file to a particular directory.[5] The compile process of this paper also uses ANT and it moves in lib directory to the WEB-INF directory after the compile process.

2.6 Web Application SQL Servlet Process Flow Diagram

The proposed SQL Servlet is for only SQL service. It processes by reading XML document from the XML directory. It separates SQL ID (ID for SQL) by using delimiter to be able to handle multi SQL DML at the same

time. It processes in database by reading from XML document using each separated SQL ID and responds it by making XML format data. Then it doesn't add and create servlet anymore for the database process, and the service can be used by adding SQL DML statement in XML document, and it has an advantage of using service by adding SQL DML statement to the XML document. And also, it gives effective development of convenience. SQL servlet process flow diagram shows as follows.

Figure 6. Web Application Compile Process Flow Diagram

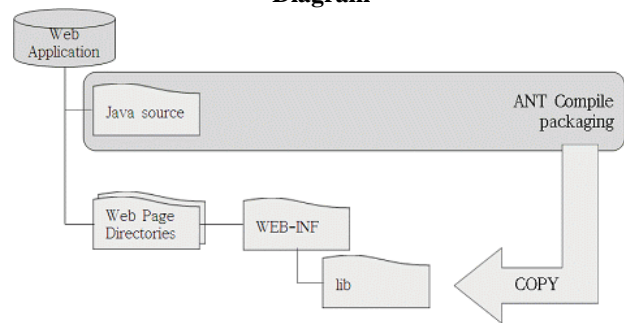
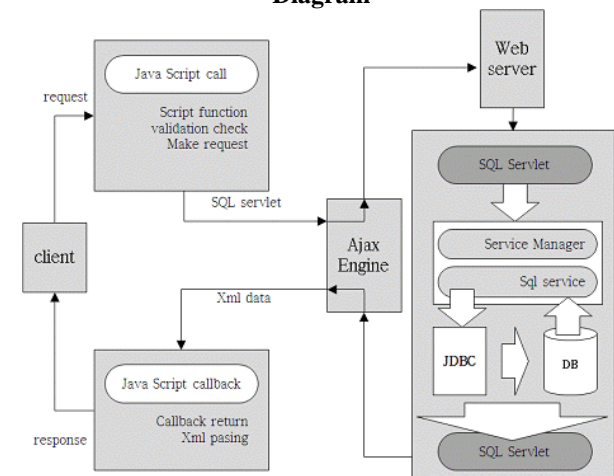


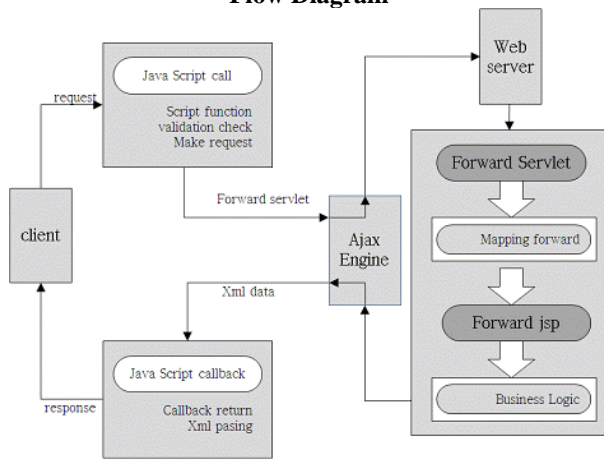
Figure 7. Web Application SQL Servlet Process Flow Diagram



2.7 Web Application Forward Servlet Process Flow Diagram

The proposed Forward Servlet in this paper is a servlet to process business logic in JSP. It is very effective for maintenance or change of implemented source. If it compiles when the source needed to change, web server need to be restarted. Because it could lead to service interruption, it forward the result in XML format after processing the JSP business logic which doesn't need to be compiled.

Figure 8. Web Application Forward Servlet Process Flow Diagram



3. Web Application Implementation

3.1 System Configuration Environment

Web Application was developed in following environment. Because it was implemented with JAVA, it could be implemented on other environments beside the implementation environment configured below.

- Web Server OS : Microsoft Window 7
- WAS Server OS : Microsoft Window 7
- Database Server OS : Microsoft Window 7
- Java Version : JDK 1.6
- WAS : Apache 2.0
- WAS : Tomcat 6.0.22
- DBMS : Oracle 10g
- Development tools : Eclipse 3.2
- Language : JAVA, JSP, JavaScript, XML
- library : apache ant, struts, ibatis


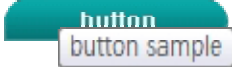




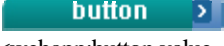
The connection between Java and Database was implemented using JDBC. Because it followed standard specification, it doesn't matter to use other database beside Oracle. Also web server, web application server and database server were implemented to work on different devices. Also, it was implemented model, view, controller by functionally separating using MVC pattern. It is effective on Web Application performance and load balancing, it guarantees independency by separating data access and transaction logic, and it is easy to expand, maintenance and reuse because it is irrelevant to the server and database.

Also we have placed emphasis on C/S method functionality and offering UI. To do this, we used Custom Tag Library expanded from JSTL (Java Standard Tag Library). The greatest advantage of Custom Tag Library is

that it can separate presentation and program implementation perfectly. Because it helps not to put messed script language code in HTML, it is very effective for maintaining JSP page.

3.2 Web Application Button Implementation Result

Next is the result of implementation of web application button. It changes the button style automatically by changing CSS style instead of changing source code. Also, it improved the convenience by using various attributes.

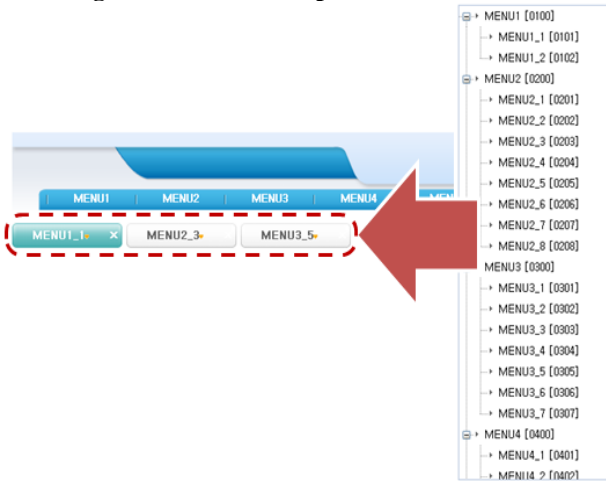
- value : 
`<webapp:button value="button"/>`
- title 
`<webapp:button value="button" title="button sample"/>`
- disable 
`<webapp:button value="button" disable="true"/>`
- styleClass 
`<webapp:button value="button" styleClass="img_btn"/>`
- classL 
`<webapp:button value="button" classL="img_btnL"/>`
- classM 
`<webapp:button value="button" classM="img_btnM"/>`
- classR 
`<webapp:button value="button" classR="img_btnR"/>`

3.3 The Implementation Result of Web Application Menu MDI

Next screen shows the implemented result of Web Application Menu by MDI. MDI is a window programming interface to make application program able to work with many document at the same time. Nature of the Web, it can't open and work multi screen in one screen.

To do this, this screen is showing the implemented result to work in one browser on each menu by applying Microsoft MDI functionality. Maximum numbers can be set by properties which are Web Application setting file, screen will be opening as many as the set.

Figure 9. The MDI Implementation Result



3.4 Web Application Tab Implementation Result

Next, It Web displays in Application Tab form on the screen. It can separate one menu which has similar function to many function screens.

```
<webapp:tab width="50" name="TAB_1,TAB_2,TAB_3"
></webapp:tab>
```



```
<webapp:tab width="50" name="TAB_1,TAB_2,TAB_3"
tabbox="true"></webapp:tab>
```



```
<webapp:tab width="50" name="TAB_1,TAB_2,TAB_3"
styleClass="blueTab"></webapp:tab>
```



```
<webapp:tab width="50" name="TAB_1,TAB_2,TAB_3"
styleClass="blueTab" tabbox="true"></webapp:tab>
```



3.5 The Implementation Result of Web Application GRID and TREE

Next, it is the screen which displays Web Application in GRID and TREE form. It was implemented similar as C/S method. We have implemented ROW click, double click, cell Change, etc by implementing events in GRID and TREE property.

- GRID implemented source code

```
<webapp:grid id="SAMPLEGRID" width="350"
height="400" no="true">
<tr event="O,D,C">
<td width="100" column="menu_id"
title="MENU ID"></td>
<td width="100" column="menu_nm"
title="MENU NAME"></td>
<td width="100" column="order"
title="ORDER"></td>
</tr>
</webapp:grid>
```

Figure 10. GRID implementation result

MUNU ID	MENU NAME	ORDER
0100	MENU1	1
0101	MENU1_1	1
0102	MENU1_2	2
0200	MENU2	2
0201	MENU2_1	1
0202	MENU2_2	2
0203	MENU2_3	3

- TREE implemented source code

```
<webapp:tree id="SAMPLETREE" width="350"
height="400" no="true">
<tr event="O,D,C">
<td width="100" column="menu_id"
title="MENU ID"></td>
<td width="100" column="menu_nm"
title="MENU NAME"></td>
<td width="100" column="order"
title="ORDER"></td>
</tr>
</webapp:tree>
```

Figure 11. TREE Implementation Result

- Events

- -Mouse left click : O
- -Mouse double click : D
- -Cell change click : C
- -Multiple Events : separate with ‘ ; ’

These events can be implemented to handle the events by reconstructing in JavaScript function. Left mouse click event could be implemented by overriding `grid_onClick(id, nRow)` function, and double click event could be implemented by overriding `grid_dbClick(id, nRow)` function, and Cell Change event could be implemented by overriding `grid_cellChange(id, nRow, nCol)` function.

4. Future Work

In this paper, we have researched on the effective way of implementing the Web Application. And, if we use proposed SQL servlet and Forward servlet in this paper, developer doesn't have to make extra business logic such as search, update and delete and it has good effect on developing conveniently and shortening the period of development. If JAVA program source change, it could have restarting problem, it can be solved by using Forward servlet processing the business logic.

We were able to increase the effectiveness of

implementation by making functionality of C/S model's GRID, TAB GUI and making GUI usable on Web Application. Requesting and responding data are still exposed with simple snooping program while communicating between servers. The research for the security should be continued for the client and server communication.

References

- [1] ORACLE, "API" <http://java.sun.com>, 2011
- [2] H. J. Moon, I. J. Whang, S.I. Kim, "Client/Server Environments and Computing Models.", Korea Information Processing Society review, Vol.4, No. 6, pp.4-15, 1997
- [3] Steve Burbeck, Ph.D., Applications Programming in Smalltalk-80: How to use Model? View? Controller", 1987
- [4] B.G. Choi, "MVC Prime Work mplementations." J.S.P., 2003
- [5] APACHE, "ANT", <http://ant.apache.org>, 2011
- [6] MySQL, Guide to Cost-effective Database Scale-Out using MySQL, http://www.mysql.com/why-mysql/mysql_power.html
- [7] Johnson R, Hoeller J, Expert One-on-One J2EE Development without EJB. 2004, Indianapolis: Wiley Publishing Inc
- [8] John Arthur, Shiva Azadegan, Spring Framework for rapid open source J2EE Web Application Development: A case study, IEEE Computer Society
- [9] Rod Johnson, Juergen Hoeller, The Spring Framework: Introduction to Lightweight J2EE™ Architecture, JavaOne conference
- [10] Servlet Technology, "Servlet", <http://java.sun.com/products/servlet/>
- [11] JavaServer Pages, "JSP", <http://java.sun.com/products/jsp/>
- [12] JSTL specification, "JSTL", <http://jcp.org/aboutJava/communityprocess/final/jsr052/>
- [13] Struts pages, "struts", <http://jakarta.apache.org/struts/>
- [14] The Model-View-Controller (MVC) Design Pattern <http://jakarta.apache.org/struts/userGuide/introduction.html#mvc>
- [15] Oracle10g Server Application Developer's Guide, Oracle 社, February 2008
- [16] Mark C. Reynolds, Andrew Wooldridg, ed. M.H. Kim, Using Java Script, Information Public Group,1997
- [17] Yong-Woo Park, Java Web Application Programming Bible, YongJin



OhSoo Kwon, MS Student, Seoul University of Science and Technology
Area of interest: JAVA, Android, Objective-C



HyeJa Bang, Professor, She received a Bachelor of computer science in Soongsil University and Master's degree in Computer Science in North Texas University, and Ph.D degree in Soongsil Univ. She was a visiting professor in Florida State University. She has been a full professor at department of computer science in Seoul National University of Science and Technology. **Area of interest:** Programming Language, XML, Formal Language Theory