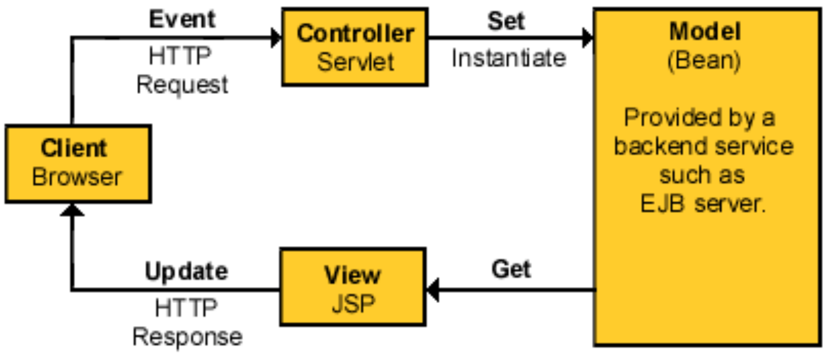
Simple MVC



An example: join.jsp

```
<%@ page language="java" %>
<%@ page import="business.util.Validation" %>
<%@ page import="business.db.MailingList" %>
<%
String error = "";
String email = request.getParameter("email");

// do we have an email address
if( email!=null ) {
    // validate input...
    if( business.util.Validation.isValidEmail(email) ) {
        // store input...
        try {
            business.db.MailingList.AddEmail(email);
        } catch (Exception e) {
            error = "Error adding email address to system.  " + e;
        }
```

join.jsp – part 2

```
if( error.length()==0 ) {
%>
            // redirect to welcome page...
            <jsp:forward page="welcome.html"/>
<%
        }
    } else {
        // set error message and redisplay page
        error = email + " is not a valid email address, try again.";
    }
} else {
    email = "";
}
%>
```

join.jsp – part 3

```
<html>
<head>
<title>Join Mailing List</title>
</head>
<body>
<font color=red><%=error%></font><br>
<h3>Enter your email to join the group</h3>
<form action="join.jsp" name="joinForm">
    <input name="email" id="email" value=<%=email%>></input>
    <input type=submit value="submit">
</form>
</body>
</html>
```

An example – comment

* Heavy HTML and Java coupling
The coder of the JSP file must be both a page designer and a Java developer. The result is often either terrible Java code or an ugly page, or sometimes both.

* Java and JavaScript blur
As the pages become larger, there can be a tendency to implement some JavaScript. When the JavaScript appears in a page, the script can get confused with the Java code. An example of a possible point of confusion is using client-side JavaScript to validate the email field.

* Embedded flow logic
To understand the entire flow of the application, you have to navigate all of the pages. Imagine the spaghetti logic on a 100-page Web site.

* Debugging difficulties
In addition to being ugly to look at, HTML tags, Java code, and JavaScript code all in one page makes it difficult to debug problems.

* Tight coupling
Changes to business logic or data means possibly touching every page involved.

* Aesthetics
Visually, in large pages, this type of coding looks messy.

Struts jsp tag libs

You write:

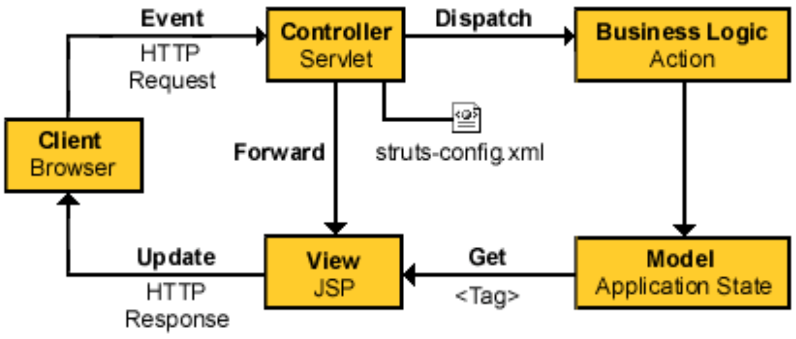
```
<form:form action="join.do" focus="email" >
    <form:text   property="email" size="30" maxlength="30"/>
    <form:submit property="submit" value="Submit"/>
</form:form>
```

Browser gets:

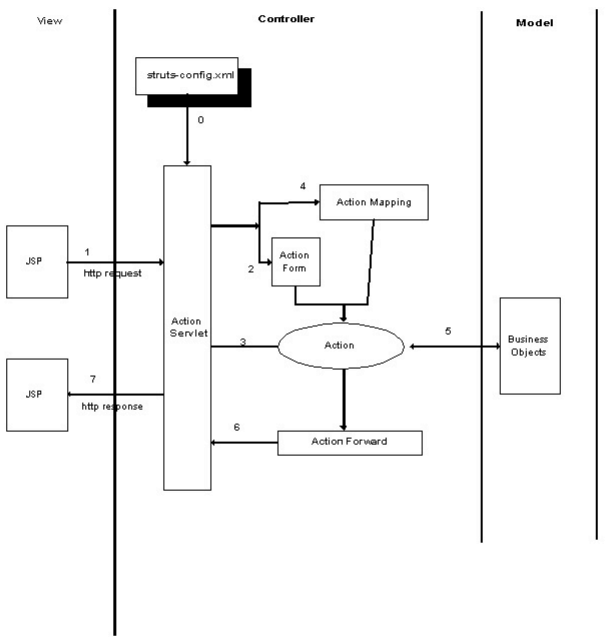
```
<form name="joinForm" method="POST"
   action="join.do;jsessionid=ndj71hjo01">
    <input type="text" name="email" maxlength="30" size="30"
   value="">
    <input type="submit" name="submit" value="Submit">
</form>
<script language="JavaScript">
<!--
    document.joinForm.email.focus()
// -->
</script>
```

The additional JavaScript sets the focus on the email address field.

Struts MVC



Event
HTTP Request
Controller Servlet
Dispatch
Business Logic Action
Client Browser
Forward
struts-config.xml
Update
HTTP Response
View JSP
Get
<Tag>
Model Application State



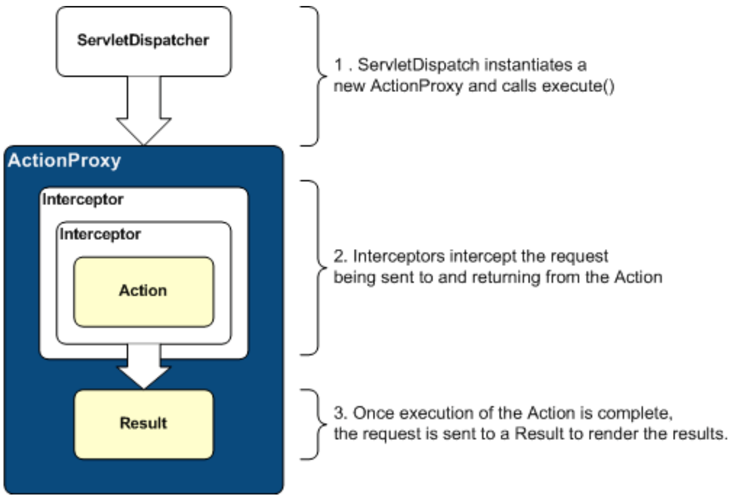
View
Controller
Model
struts-config.xml
0
4
Action Mapping
JSP
1
http request
2
Action Form
Action Servlet
3
Action
5
Business Objects
JSP
7
http response
6
Action Forward

implementation

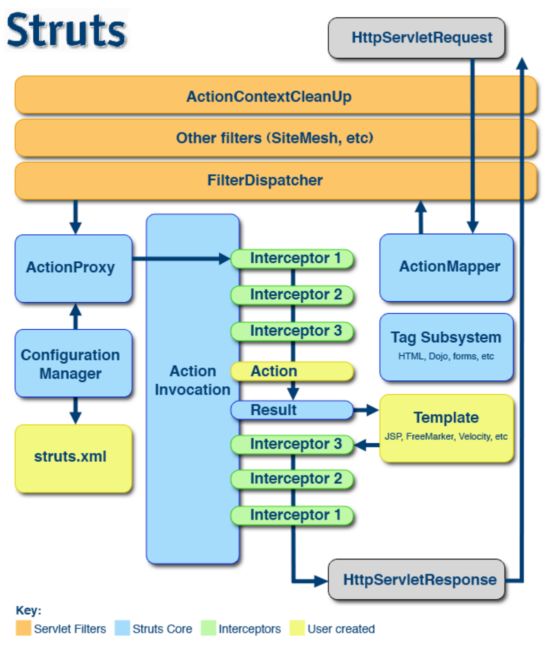
```
See code downloaded from

http://www-128.ibm.com/developerworks/ibm/library/j-struts/#download
```

Struts 2

http://struts.apache.org/2.x/docs/core-developers-guide.html


ServletDispatcher
ActionProxy
Interceptor
Interceptor
Action
Result
1 . ServletDispatch instantiates a new ActionProxy and calls execute()
2. Interceptors intercept the request being sent to and returning from the Action
3. Once execution of the Action is complete, the request is sent to a Result to render the results.

Struts 2


Struts
HttpServletRequest
ActionContextCleanUp
Other filters (SiteMesh, etc)
FilterDispatcher
ActionProxy
Configuration Manager
struts.xml
Action Invocation
Interceptor 1
Interceptor 2
Interceptor 3
Action
Result
Interceptor 3
Interceptor 2
Interceptor 1
ActionMapper
Tag Subsystem
HTML, Dojo, forms, etc
Template
JSP, FreeMarker, Velocity, etc
HttpServletResponse
Key:
Servlet Filters
Struts Core
Interceptors
User created