Webコンピューティングにおけるソフトウェア作成手法の比較評価

6W−01

田渕 梢　菊池 史典　小泉 寿男
東京電機大学 理工学部

1 はじめに

近年，インターネットユーザの急激な増加に伴い，Web コンピューティングによる電子商取引システムが注目を集めている．Web システムの構築には，J2EE(Java2 Platform Enterprise Edition)をベースとした方法が増えつつある[1]．J2EEを用いたソフトウェア開発手法には様々な形態があり，その開発手法は未だ研究中であると言える．

本稿では，Web コンピューティングにおけるソフトウェア作成手法について，J2EE に準拠した技術を用いて典型的システム例を 3 層クライアントサーバ上に構築，その作成手法の比較評価を行った．

2 比較評価対象と項目

Java を基本とし，J2EE に準拠した技術(Servlet,JSP, JavaBeans など)を使用したサーバーサイドの Web におけるアプリケーションソフトウェア作成手法について，システムの典型的例を構築し，その作成手法において比較評価を行う．比較の対象は次の二つの形とする．

方法1：JSP/Selvlet ＋ JavaBeans
JSPを表示，Servletを制御，JavaBeansをロジック記述に使用．

方法2：Servlet ＋ JavaBeans
Servlet を表示と制御，JavaBeans をロジック記述に使用．

比較項目は以下の通りとする．

- ロジックの記述の違い
- レスポンスの違い
- 記述するプログラムの記述法，長さの違い
- それぞれのプログラムの役割の違い

3 評価システムの構築

3.1 構築環境

システム構築環境を図 1に示す．ここでは 3 層クライアントサーバシステムを用いて，グループ内簡易発注取りまとめシステムを上記方法1，方法2の形態で構築した．

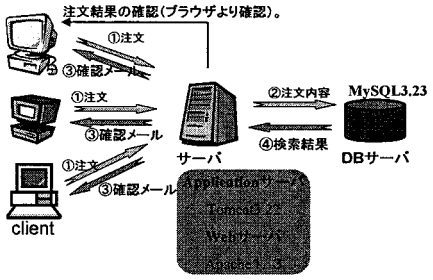


図 1 構築環境

本システムは，ある集団において，一つの店舗へのまとまった注文を行う際の利便性を向上させる事を目的としている．図1においてシステムの流れは，次の通りである．

①グループの各個人がブラウザにて，オンラインショッピング形式で注文を行う．②ユーザーが自分で注文を確認した後，自動的にDBに情報を格納 ③各ユーザーへ，注文の内容と代金の合計のメール送信④取りまとめ者がブラウザにて合計を確認，注文を行う．⑤各自がブラウザより，再度注文の確認を行う．

3.2 ソフトウェア作成方法

方法1：JSP/Selvlet ＋ JavaBeans

構成図の主要部分を図 2に示す．

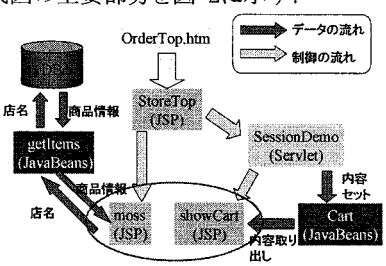


図 2 JSP/Servlet+JavaBeans

図中右半分でショッピングカートの処理を行っている．ショッピングカートは，SessionDemo(Servlet)から Cart(JavaBeans)に情報をセット，その後表示部分となるshowCart(JSP)で Cart にセットされた情報を取り出し，表示している．左半分はカタログの表示部分である．

A Comparison Evaluation of Software Creation Techniques in Web Computings
Kozue Tabuchi,Fuminori Kikuchi,Hisao Koizumi
Department of Computers and Systems Engineering, Tokyo Denki University

方法2:Servlet ＋ JavaBeans

システムの構成図を図 3に示す.

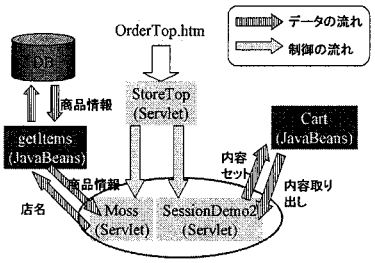


図 3 Servlet+JavaBeans

図中右半分はショッピングカートの処理であるが, 図 2と多少異なる. ショッピングカートへの内容のセット, 内容の取り出し, 画面への表示を一つの Servlet プログラムが行い, 図 2での二つのプログラム(SessionDemo,showCart)が, 一つの Servlet プログラム(SessionDemo2)になっている. 表示部分も全ての構築を Servlet で行っている.

4 評価と考察

4.1 構築作業における比較評価

(1) それぞれのプログラムの役割の違い

方法1では, 画面表示部分を JSP に, 動作内容を JavaBeans に, またその統括を Servlet が行っている. それに対して, 方法2の構築方法では見た目の部分と全体の統括とを共に Servlet が行っている. 役割の違いによって, 後述のロジック等の違いも現れる事がわかった.

(2) ロジックの記述の違い

基本的に, 業務処理を行っているロジックに関しては JavaBeans によって共通に行われているので異なることはない. 記述に違いが出るとすれば, それは構造によるものである.

まず, 方法1ではいくつかのプログラムに機能を分散させている. 従って, その連携のために記述される部分が数多くある. 特に, Servlet から JSP へ処理を切り替えるための記述は, 例外処理などを伴うため, 最低でも3行ほどの記述が必要となる. 対して方法2ではそう言った部分は少ない.

(3) プログラムの記述法, 長さの違い

同じ役割を行っている moss+SessionDemo と SessionDemo2 を比較すると, 記述に関してはまず, 前者には処理切り替えの為の記述がある.

Servlet で表示部分を記述する場合, 必ず登場するのが, println などと言った, 出力用の命令である. したがって, 画面出力をする部分が多ければ多いほど, 書かねばならないプログラムが増え, エスケープ文字などの処置を施さねばならず, プログラムを作る際には大変な手間となる. 対して, JSP を用いて画面出力を行う際には HTML 部分は通常の HTML と同様に記述を行えば良いので, プログラムを作る人間はエスケープ文字などの処理にとまどう事無く, また, println などという命令も書かずに済む為, 全体としてすっきりとしたプログラムになる.

全体として, 長さに関しては前者が 107＋46＝153 行, 後者が 125行と, 前者の方が多少長くなる事がわかる. 一般に, J2EE に準拠した技術は, 方法1の様な形で使用するのが最も良いとされてきた. 今回, それとは別の方法で構築を行い比較を行った. プログラムの全体的な長さとしては方法2の形の方が短くなるが, 後のデバックや構築前の設計のしやすさを考えると方法1の構築方法の方が優れていると言える.

4.2 システム動作のレスポンスの違いの評価

カタログページより, カートの中身を変更する手続き毎に計測を行った. その結果を表 1に示す.

表 1 実行 2 回目以降のレスポンス時間

	変更	削除
方法1[ms]	1583	1342
方法2[ms]	1212	1262

方法1では処理切り替えの時間の分, 方法2より時間がかかると予想していた. 一回目の実行の際は方法1による出力の方が多少時間がかかる. しかし, それ以降のレスポンスに関してはどちらの手法もさほど時間的な違いはなく, 結果的に構築の際に容易である方法を選んだ方が良いという事がわかった.

5 まとめ

本稿では, J2EE の技術を用い, Web アプリケーションの様々な方法での構築を行った. 更に, その比較, 評価を行った結果, 作成方法による特徴を知ることが出来た.

＜参考文献＞

[1] 宮崎 肇之, 他 “Web アプリケーション開発スタイル標準化 ”情報処理学会第63回全国大会講演論文集(1), pp133－134