



Portlet Development Frameworks

Cris Holdorph (holdorph@unicon.net)

Unicon, Inc.

<http://www.unicon.net/>

Agenda



Portlet Development Frameworks

- Background
- MVC a Review
- Portlets a Review
- Candidates
- Recommendation

Background

- Writing Portlets prevents vendor/portal lock in
- Using a standard framework allows for concentrating on the business logic
- uPortal 3 is Portlet based
- All uPortal basic administration channels will be rewritten as Portlets
- Frameworks did not initially exist or support Portlets
- ***Everything Changes***

MVC

- MVC stands for Model View Controller
- The “original” Design Pattern
- Clean separation of user interface presentation
- Web applications have adopted MVC as a way to keep “html” out of the (Java) “code”
- MVC framework
 - Model – The data structures to hold (and *modify*) data for the View and Controller (e.g., a Java Bean)
 - View – The file that includes all the markup for the rendered page, possibly including some dynamic sections (e.g., a JSP)
 - Controller – The main source code that gathers the data into the model and decides which View to use (e.g., a Servlet)

Portlets

- Portlets have certain traits that make them different than Servlets
- **URLs** – Portlets must use the Portlet container to create URLs
- **Render/Action** – Portlets define two phases
- **Modes** – Portlets support different modes
- **Window States** – Portlets support different window states

The Candidates

- MVCPortlet
- Struts
- Spring Portlet MVC
- Spring Web Flow
- JavaServer Faces
- Tapestry
- JSP (with JSTL and Portlet tag libraries)

MVCPortlet

- Built by Nabh Information Systems
- Open Source (GPL)
- XML Configuration
- JSP Views
- Custom Tag Libraries
- Supports Actions/Validators

- <http://www.nabh.com/>

Struts

- Most Popular Servlet based MVC framework
- No direct Portlet support in 1.x (latest is 1.2.9)
- Apache Portals Subproject – Portals Bridges
 - Portals Struts Bridge
 - Limits what can be done in struts
- <http://struts.apache.org/>
- <http://portals.apache.org/bridges/>

Spring Portlet MVC

- Spring Portlet MVC
- Part of the Spring 2.0 release (currently at Milestone-5)
- Simplifies many aspects of the main Portlet class
- Simplifies many aspects of the portlet.xml
- Supports various view technologies
- Custom JSP Tag (Spring Web MVC) libraries available

- <http://www.springframework.org/>
- <http://opensource.atlassian.com/confluence/spring/display/JSR168/Home>

Spring Web Flow

- Works on top of Spring's existing MVC frameworks (both Servlet and Portlet)
- Abstracts the Page Flow and Actions
- Easy to read and understand flow definition language
- Does not require any spring imports in your business logic code
- Flow definition and Java code can be identical for a Portlet and Servlet application (only configuration files such as web.xml, portlet.xml would change)
- <http://opensource.atlassian.com/confluence/spring/display/WEBFLOW/Home>

JavaServer Faces

- JSR-127 defined JavaServer Faces
- JSR-252 is defining JavaServer Faces 1.2
- Portlet support is through an addon JSF Portlet library
- Changes the traditional WWW MVC paradigm
- Analogous to rich user interface programming
- Define screens, fields, beans, validators
- Uses JSP Views with custom JSF tag libraries

- <http://myfaces.apache.org/>
- <https://javaserverfaces.dev.java.net/>

Tapestry

- Popular Servlet Development Framework
- Apache Jakarta project since 2001
- Version 4.0 tries to **fully** support the true Portlet specification (including action/render, modes and states)
- Some coding differences exist between portlet and servlet applications

- <http://jakarta.apache.org/tapestry/>
- <http://jakarta.apache.org/tapestry/tapestry-portlet/index.html>

WebWork

- Servlet Development Framework
- Started in early 2000 (opensymphony)
- Version 2.2 supports portlets
- *Some coding differences exist between portlet and servlet applications*
- <http://www.opensymphony.com/webwork/>
- <http://wiki.opensymphony.com/display/WW/Pportlet+Configuration>

Recommendation

- Avoid applications that run as both servlet and portlet applications at the same time
- Spring Web Flow and Spring Portlet MVC
- Spring Portlet MVC
- Use JSP for View Technology
 - JSTL 1.1
(<http://jakarta.apache.org/taglibs/doc/standard-doc/intro.html>)
 - JSP 2.0 / 2.1
 - Portlet Tag Library (JSR-168)

Future Considerations

- Struts Action Framework 2.0 (WebWork 2.3)
- Struts Shale Framework
- Wicket

- Revisit
 - MVCPortlet
 - JavaServer Faces (after JSR-286)

Future Considerations – JSR 286

- Expected to be finished May 2007
 - Some parts expected to be available for review as early as July 2006
- Features for JSR 286
 - Portlet to Portlet communication through events
 - Sharing of Portlet Session data ACROSS portlet web applications
 - Sharing of Render Parameters
 - Better support for development frameworks (especially JavaServer Faces)
 - AJAX support

Additional Considerations

- uPortal 3
 - Native portlet support
- Spring
 - Each web application must do its own Spring configuration at this time
- Portlet Development Frameworks
 - Portlet tag library (defined by JSR-168) provided by uPortal
 - All other libraries must be provided by the portlet web application

