Comparing Web Frameworks

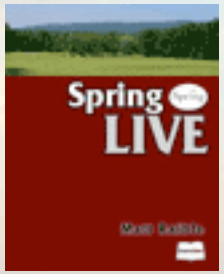
Struts, Spring MVC, WebWork, Tapestry & JSF

Matt Raible
mraible@virtuas.com

VIRTUAS

Who is Matt Raible?

- Developing websites since 1994 - Developing J2EE webapps since 1999
- Committer on several open source projects: AppFuse, Roller Weblogger, XDoclet, Struts Menu, Display Tag
- J2EE 5.0 and JSF 1.2 Expert Group Member
- Author: Spring Live (SourceBeat) and contributor to Pro JSP (Apress)



VIRTUAS

My Experience

- **Struts**: used since June 2001 - same time 1.0 was released.
- **Spring MVC**: used since January 2004 - before 1.0 was released.
- **WebWork**: used since July 2004.
- **Tapestry**: used since July 2004.
- **JSF**: used since July 2004 - both Sun's RI and MyFaces.

VIRTUAS

Meet the Candidates

Struts

- **Pros**:
  - The "Standard" - lots of Struts jobs
  - Lots of information and examples
  - HTML tag library is one of the best
- **Cons**:
  - ActionForms - they're a pain
  - Can't unit test - StrutsTestCase only does integration
  - Project has been rumored as "dead"
- Action and JSP Examples →

VIRTUAS

Spring MVC

- **Pros**:
  - Lifecyle for overriding binding, validation, etc.
  - Integrates with many view options seamlessly: JSP/JSTL, Tiles, Velocity, FreeMarker, Excel, XSL, PDF
  - Inversion of Control makes it easy to test
- **Cons**:
  - Configuration intensive - lots of XML
  - Requires writing lots of code in JSPs
  - Almost too flexible - no common parent Controller
- Controller and JSP Examples →

VIRTUAS

WebWork

- **Pros**:
  - Simple architecture - easy to extend
  - Tag Library is easy to customize - backed by Velocity
  - Interceptors are pretty slick
- **Cons**:
  - Small Community
  - Documentation only recently written, few examples
  - Client-side validation immature
- Action and JSP Examples →

Tapestry

- **Pros**:
  - Very productive once you learn it
  - Templates are HTML - great for designers
  - Healthy and smart user community
- **Cons**:
  - Documentation very conceptual, rather than pragmatic
  - Steep learning curve - very few examples
  - Long release cycles - major upgrades every year
- Page Classes and Template Examples →

VIRTUAS

JSF

- **Pros**:
  - J2EE Standard - lots of demand and jobs
  - Fast and easy to develop with
  - Rich Navigation framework
- **Cons**:
  - Tag soup for JSPs
  - Immature technology - doesn't come with everything
  - No single source for implementation
- Page Beans and JSP Examples →

VIRTUAS

Let's Compare!

VIRTUAS

Evaluation Criteria

- **List Screens**: How easy is it to code pageable, sortable lists?
- **Bookmarkability**: Can users bookmark pages and return to them easily?
- **Validation**: How easy is it to use and does it support client-side (JavaScript) validation?
- **Testability**: How easy is it to test Controllers out of container?

VIRTUAS

Evaluation Criteria, cont.

- **Post and Redirect**: How does the framework handle the duplicate post problem?
- **Spring Integration**: Does the framework support using Spring in the middle tier; how easily?
- **Internationalization**: How is i18n supported and how easy is it to get messages in Controllers?
- **Page Decoration**: What sort of page decoration/ composition mechanisms does the framework support?

VIRTUAS

Evaluation Criteria, cont.

- **Tools**: Is there good tool (particularly IDE) support for the framework?
- **Marketability of Skills**: If you learn the framework, will it help you get a job?
- **Job Count**: What is the demand for framework skills on dice.com?

VIRTUAS

List Screens

- How easy is it to integrate a sortable/pageable list of data?
  - Struts, Spring MVC and WebWork can all use Tag Libraries like the Display Tag
  - Tapestry has a contrib:Table component
  - JSF has a dataTable with no sorting - have to write your own logic if you want it

VIRTUAS

Bookmarking and URLs

- Using container-managed authentication (or other filter-based security systems) allow users to bookmark pages. They can click the bookmark, login and go directly to the page.
  - WebWork has namespaces - makes it easy
  - Struts and Spring allow full URL control
  - Tapestry has ugly URLs - difficult to segment the app for different roles
  - JSF does a POST for everything

VIRTUAS

Validation

- Validation should be easy to configure, be robust on the client side and either provide good out of the box messages or allow you to easily customize them.
  - Struts and Spring MVC use Commons Validator - a mature solution
  - WebWork uses OGNL for powerful expressions - client-side support very new
  - Tapestry has very robust validation - good messages without need to customize
  - JSF - ugly default messages, but easiest to configure

VIRTUAS

Testability

- Struts - can use StrutsTestCase
- Spring and WebWork allow easy testing with mocks (e.g. EasyMock, jMock, Spring Mocks)
- Tapestry appears difficult to test because page classes are abstract, Tapestry Test Assist to the rescue
- JSF page classes can be easily tested and actually look a lot like WebWork actions

VIRTUAS

Post and Redirect

- The duplicate-post problem, what is it?
- Easiest way to solve: redirect after POST
- Is there support for allowing success messages to live through a redirect?
  - Struts and Spring are the only framework that allows success messages to live through a redirect
  - WebWork requires a custom solution
  - Tapestry requires you to throw an Exception to redirect
  - JSF requires a custom solution, i18n messages difficult to get in page beans

VIRTUAS

Spring Integration

- All frameworks have integration with Spring
  - **Struts**: ContextLoaderPlugin and Base classes
  - **WebWork**: SpringObjectFactory
  - **Tapestry**: override base engine, grab from servlet context, put into global map
  - **JSF**: DelegatingVariableResolver or JSF-Spring Library

Internationalization

- JSTL's <fmt:message> tag makes it easy
- No standard for getting i18n messages in controller classes
- Struts, Spring and JSF encourage one ResourceBundle per locale
- WebWork and Tapestry advocate separate files for each page/action

Page Decoration

- Tiles Experience: used since it first came out
- SiteMesh is much easier to setup and use
- Tiles can be used in Struts, Spring and JSF
  - Requires configuration for each page
- SiteMesh can be used with all frameworks
  - Requires very little maintenance after setup
- Use both for powerful decoration/composition

VIRTUAS

Tools

- Struts has a lot of IDE support and even has frameworks built on top of it (i.e. Beehive's PageFlow)
- Spring has Spring IDE - only does XML validation, not a UI/web tool
- WebWork has EclipseWork
- Tapestry has Spindle - great for coders
- JSF has many, all cost money and hook into proprietary app servers

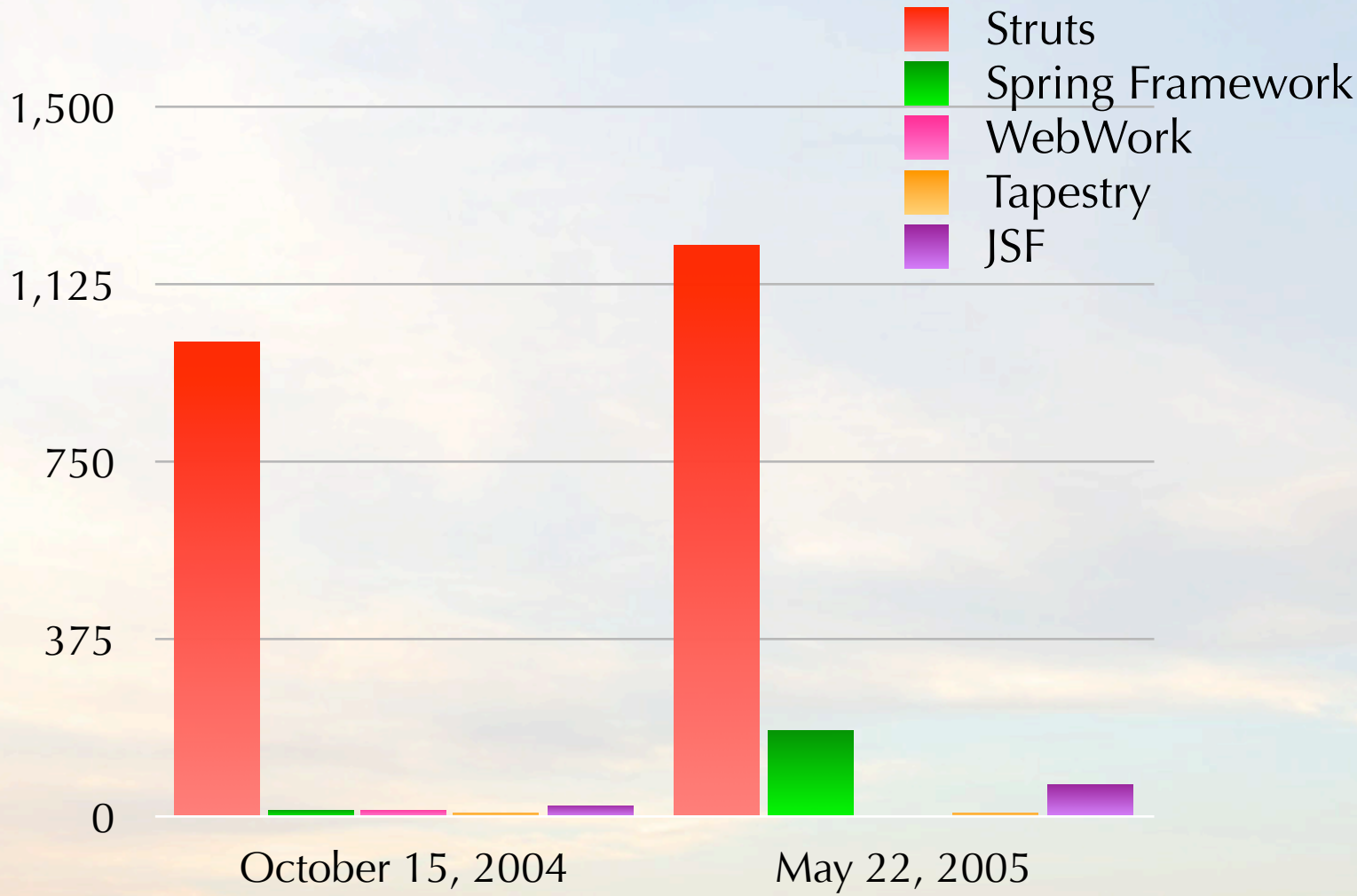
VIRTUAS

Marketability of Skills

- Struts is still in high-demand and widely-used
- Spring is getting more press, but mostly due to the framework's other features
- WebWork is gaining ground, but very scarce on job boards
- Tapestry is even more scarce - needs more marketing
- JSF is quickly becoming popular

VIRTUAS

Dice Job Count



VIRTUAS

My Opinion

- Struts is fast to develop with because most problems have been solved. HTML tag library the best of the bunch.
- Spring is nice, but lack of form tags drops it down a notch or two. JSP 2.0 tag files exist in issue tracker.
- I like WebWork a lot, but lack of of good client-side validation support is a killer.
- Tapestry - I haven't finished the learning curve.
- JSF - needs to listen to developers to see what they want instead of tools vendors.

VIRTUAS

Which would I choose?

- Quick and dirty project?
  - Whichever one needs the most work in AppFuse ;-)
- Massive enterprise project?
  - Tapestry for its reusable components
- If I got a job as an open source developer?
  - WebWork because using it requires you to dig into the frameworks

VIRTUAS

Resources

- Download sample apps for this presentation
  - http://equinox.dev.java.net/framework-comparison
- Struts - http://struts.apache.org
  - **StrutsTestCase**: http://strutstestcase.sf.net
- Spring MVC - http://www.springframework.org
  - **Spring IDE**: http://www.springide.org
  - **Gaijin Studio**: http://gaijin-studio.sf.net
- WebWork - http://opensymphony.org/webwork
  - **Eclipse Plugin**: http://sf.net/projects/eclipsework
  - **IDEA Plugin**: http://wiki.opensymphony.com/display/WW/IDEA+Plugin

VIRTUAS

Resources, cont.

- Tapestry - http://jakarta.apache.org/tapestry
  - **Spindle**: http://spindle.sourceforge.net
- JSF - http://java.sun.com/j2ee/javaserverfaces and http://myfaces.apache.org
  - **Java Studio Creator**: http://sun.com/software/products/jscreator
  - **MyEclipse**: http://myeclipseide.com
- **IDEA**: http://www.jetbrains.com/idea
- **SiteMesh**: http://opensymphony.com/sitemesh

VIRTUAS

Resources, cont.

- Testing Frameworks
  - **JUnit**: http://junit.org
  - **EasyMock**: http://easymock.org
  - **jMock**: http://jmock.org
  - **jWebUnit**: http://jwebunit.sourceforge.net
  - **Canoo WebTest**: http://webtest.canoo.com
  - **Tapestry Test Assist**: http://howardlewisship.com/blog/2004/05/tapestry-test-assist.html
- **XDoclet** - http://xdoclet.sourceforge.net
- **AppFuse** - http://appfuse.dev.java.net

VIRTUAS

Books

- **Struts in Action**, Ted Husted and Team
- **Struts Live**, Rick Hightower and Jonathan Lehr
- **Spring Live**, Matt Raible
- **Pro Spring**, Rob Harrop and Jan Machacek
- **Spring in Action**, Craig Walls and Ryan Breidenbach
- **Professional Java Development with Spring**, Rod Johnson, Juergen Hoeller and Team

VIRTUAS

Books, cont.

- **WebWork Live**, Matthew Porter
- **WebWork in Action**, Patrick Lightbody and Team
- **Tapestry Live**, Warner Onstine
- **Tapestry in Action**, Howard Lewis Ship
- **Core JSF**, David Geary and Cay Horstmann
- **JSF in Action**, Kito Mann

VIRTUAS

The End

...or is it just the beginning...

what about ajax?

VIRTUAS