COMPARING JVM WEB FRAMEWORKS

Matt Raible
http://raibledesigns.com
Introductions

- Your experience with web development?
- Your experience with Java EE development?
- What do you want to get from this session?
- Experience with Spring MVC, GWT, Rails, Grails, Wicket or Struts 2?
Who is Matt Raible?

Blogger on raibledesigns.com

Founder of AppFuse

Web Framework Connoisseur

Father, Skier, Cyclist

Montana Native
Session Agenda

- The Problem with Web Frameworks
- The Candidates
- Comparison Points
- The Matrix
- Conclusion
- Q and A
The Problem
How do you choose?
ELIMINATE, DON'T INCLUDE
I've got a new way for you to build web apps!
... while I'd *love* to see life made simpler for Java web developers, and a lot of the things happening in Struts2 are going that way -- it won't be me doing it.

I've gone over to the dark side :-) and much prefer to develop in Rails -- for the conciseness mentioned above, but also because I don't ever have to do a "build" or "deploy" step during my development cycle any more. But you guys and gals need to be reminded that *this* is the kind of thing you are competing against if you expect to attract Rails developers ... or to avoid even more "previously Java web developer" defectors like me :-).

-- Craig McClanahan, 10/23/2007

http://markmail.org/thread/qfb5sekad33eobh2
Choosing a Framework

- Choose a short list of frameworks to prototype with.
- Create an application prototype with each framework.
- Document findings and create a matrix with important criteria.
- Create presentation to summarize document.
- Deliver document, presentation and recommendation.
2007: 6 Important Factors

- Request, Component or RIA Framework
- Ease of Development
- Project Community
- Project Future and Roadmap
- Maintenance
- Technical Features
Request, Component or RIA

Are you developing...
– ... a consumer facing site?
– ... a desktop-like, application?
– ... a media-rich site?

If the framework is built for state-fulness, should you use it in a stateless architecture?
Types of Applications

- High-traffic, internet facing, infinite scalability
- Intranet-based, behind the firewall, few users
- Products, to be maintained for 5-10 years
- Legacy Backend
- Others?
Types of Frameworks

- Request Based Frameworks
  - Struts 2, Spring MVC, Rails, Stripes

- Component Based Frameworks
  - JSF, Tapestry, GWT

- Rich Internet Applications
  - Flex

- One Size Fits All
  - Grails
Do they matchup?

- High-traffic, internet facing, infinite scalability
  - Request-based frameworks

- Intranet-based, behind the firewall, few users
  - Component-based frameworks

- Products, to be maintained for 5-10 years
  - Largest Community, Most Vendor Support

- Legacy Backend
  - Same Language as backend
2010: Now There’s 20

1. Developer Productivity
2. Developer Perception
3. Learning Curve
4. Project Health
5. Developer Availability
6. Job Trends
2010: Comparison Points

7. Templating
8. Components
9. Ajax
10. Plugins or Add-Ons
11. Scalability
12. Testing Support
2010: Comparison Points

- i18n and l10n
- Validation
- Multi-language Support (Groovy / Scala)
- Quality of Documentation/Tutorials
- Books Published
- REST Support (client and server)
2010: Comparison Points

- Mobile / iPhone Support
- Degree of Risk
Comparison Matrix

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Struts 2</th>
<th>Spring MVC</th>
<th>Wicket</th>
<th>JSF</th>
<th>Tapestry</th>
<th>Stripes</th>
<th>GWT</th>
<th>Grails</th>
<th>Rails</th>
<th>Flex</th>
<th>Vaadin</th>
<th>Lift</th>
<th>Play</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer Productivity</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Developer Perception</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning Curve</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Health</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Developer Availability</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Job Trends</td>
<td>1.00</td>
<td>1.00</td>
<td>0.00</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Templating</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Components</td>
<td>0.00</td>
<td>0.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ajax</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plugins or Add-Ons</td>
<td>0.50</td>
<td>0.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scalability</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>i18n and I10n</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Validation</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multi-language Support (Groovy / Scala)</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality of Documentation/Tutorials</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Books Published</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REST Support (client and server)</td>
<td>0.50</td>
<td>1.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Mobile / iPhone Support</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>Degree of Risk</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td><strong>Totals</strong></td>
<td><strong>14.5</strong></td>
<td><strong>17</strong></td>
<td><strong>14.5</strong></td>
<td><strong>12.5</strong></td>
<td><strong>13.5</strong></td>
<td><strong>14</strong></td>
<td><strong>17</strong></td>
<td><strong>16.5</strong></td>
<td><strong>17</strong></td>
<td><strong>13.5</strong></td>
<td><strong>11.5</strong></td>
<td><strong>11.5</strong></td>
<td><strong>13</strong></td>
</tr>
</tbody>
</table>

Matrix Results

- Spring MVC (17)
- GWT (17)
- Ruby on Rails (17)
- Grails (16.5)
- Wicket (14.5)
- Struts 2 (14.5)
Pros and Cons
**Pros**

- Easy Configuration with Annotations and Conventions
- Integrates with many view options seamlessly: JSP / JSTL, Tiles, FreeMarker, Excel, PDF, JSON
- Excellent REST Support

**Cons**

- Instant reload not built-in, need JRebel or Spring Roo
- No open development process, need to be SpringSource
- Ajax requires 3rd-party library (can be a good thing!)
GWT

**Pros**
- Write Java => Produces Optimized JavaScript
- Easy to learn and develop with standard Java Tools
- Vibrant Community

**Cons**
- You have to know Java
- Slow to compile, difficult to test
- More like a JSP Tag Library than a web framework
Ruby on Rails

**Pros**
- Easy to learn and understand for Web Developers
- Lots and lots of documentation
- Passionate Community

**Cons**
- Slightly less performant by default
- Dynamic language means more tests
- Development Tools and Debugging
Grails

Pros

- Easy dynamic language transition for Java Developers
- Groovy
- Plugins for all types of applications

Cons

- Groovy learning targets Java Developers
- Stack traces are horrendous
- Knowledge of underlying frameworks not required, but helpful
Wicket

**Pros**
- Great for Java Developers
- Tight binding between pages and views
- Active community - support from creators

**Cons**
- No Jobs or Developers
- Stateful by default
- HTML Templates live next to Java code by default
Rails vs. Grails

- Grails: 40 requests per second, Rails: 32
Language Performance

Time (ms) per iteration

Java  C++  JRuby  Python  Groovy  PHP

Jobs in 2010 (US)
Developer Skills (US)

LinkedIn

- Spring MVC
- GWT
- Rails
- Grails
- Wicket

0
2,750
5,500
8,250
11,000
Developer Skills (World)

- Spring MVC
- GWT
- Rails
- Grails
- Wicket

LinkedIn

Number of Developers:
- 0
- 2,750
- 5,500
- 8,250
- 11,000
Pretty Graphs
Pretty Graphs

Spring MVC, GWT, Ruby on Rails, Grails, Wicket Job Trends

Job Trends from Indeed.com

Percentage of Matching Job Postings

Jan ’06 Jan ’07 Jan ’08 Jan ’09 Jan ’10

Indeed.com searches millions of jobs from thousands of job sites. This job trends graph shows the percentage of jobs we find that contain your search terms.
Pretty Graphs

Spring MVC, GWT, Ruby on Rails, Grails, Wicket Job Trends

Job Trends from Indeed.com

Scale: Absolute - Relative

Top Job Trends
1. Facebook
2. Virtualization
3. iPhone
4. Social Media
5. Sharepoint
6. Twitter
7. Certified Nurse Assistant
8. Patient Care Technician
9. Cloud Computing
10. Blogger
Pretty Graphs

Spring MVC, GWT, Ruby on Rails, Java Struts Job Trends

Job Trends from Indeed.com

Scale: Absolute - Relative

1. Facebook
2. Virtualization
3. iPhone
4. Social Media
5. Sharepoint
6. Twitter
7. Certified Nurse Assistant
8. Patient Care Technician
9. Cloud Computing
10. Blogger
Pretty Graphs

Spring MVC, GWT, Ruby on Rails, Java Struts, PHP Job Trends

Job Trends from Indeed.com

- Spring MVC
- GWT
- Ruby on Rails
- Java Struts
- PHP

Scale: Absolute - Relative

Email to a friend
Post on your blog/website

Top Job Trends
1. Facebook
2. Virtualization
3. iPhone
4. Social Media
5. Sharepoint
6. Twitter
7. Certified Nurse Assistant
8. Patient Care Technician
9. Cloud Computing
10. Blogger

Indeed.com searches millions of jobs from thousands of job sites. This job trends graph shows the percentage of jobs we find that contain your search terms.

Find Spring Mvc jobs, Gwt jobs, Ruby On Rails jobs, Java Struts jobs, PHP jobs
Pretty Graphs

ASP.NET, Ruby on Rails, Java Struts, PHP Job Trends

Job Trends from Indeed.com
- ASP.NET
- Ruby on Rails
- Java Struts
- PHP

Scale: Absolute - Relative

Top Job Trends
1. Facebook
2. Virtualization
3. iPhone
4. Social Media
5. Sharepoint
6. Twitter
7. Certified Nurse Assistant
8. Patient Care Technician
9. Cloud Computing
10. Blogger

Indeed.com searches millions of jobs from thousands of job sites. This job trends graph shows the percentage of jobs we find that contain your search terms.
* Spring MVC uses Forums, which don’t provide this data.
Books on Amazon

October 2010

- Spring MVC
- Rails
- GWT
- Grails
- Wicket
2010 Releases

As of October 28, 2010

Spring MVC: 4 releases
GWT: 0 releases
Rails: 8 releases
Grails: 8 releases
Wicket: 12 releases
StackOverflow

Tagged Questions (October 28, 2010)

Java  PHP  .NET  Python  Ruby
Spring MVC

- [http://www.springsource.org](http://www.springsource.org)
- Apache 2.0 License

Recommended Books:
- Expert Spring MVC
- Spring in Action
- Pro Spring
Google Web Toolkit

- [http://code.google.com/webtoolkit](http://code.google.com/webtoolkit)
- Apache 2.0 License
- Active Community @ [http://groups.google.com/group/Google-Web-Toolkit](http://groups.google.com/group/Google-Web-Toolkit)

Recommended Books:

- GWT in Action
- GWT in Practice
- Google Web Toolkit Solutions
Ruby on Rails

- [http://rubyonrails.org](http://rubyonrails.org)
- MIT License
- Active Community @ [http://rubyonrails.org/community](http://rubyonrails.org/community)

Recommended Books:

- Agile Web Development with Rails
- Rails for Java Developers
Grails

- [http://grails.org](http://grails.org)
- Apache 2.0 License
- Active Community @ [http://grails.org/Community](http://grails.org/Community)

**Recommended Books:**
- Getting Started with Grails
- Grails: A Quick-Start Guide
- Programming Groovy
Apache Wicket

- [http://wicket.apache.org](http://wicket.apache.org)
- Apache 2.0 License
- Active Community @ [http://wicket.apache.org/help/email.html](http://wicket.apache.org/help/email.html)

Recommended Books:
- Wicket in Action
- Pro Wicket
CONCLUSION

What if there is no “best” web framework?
How to Choose?

- Prioritize a list of features that are important to your application.
- Pick 3-4 frameworks and do a 1-week spike with each, developing the same application.
- Document and rank each framework against your list of features.
- Calculate and choose!
Questions?

Contact Information
http://raibledesigns.com
http://twitter.com/mraible

Download Presentation
http://slideshare.net/mraible