



JavaOne™

java.sun.com/javaone

Ajax and JSF: Natural Synergy

Kito D. Mann, Principal Consultant

TS-6482

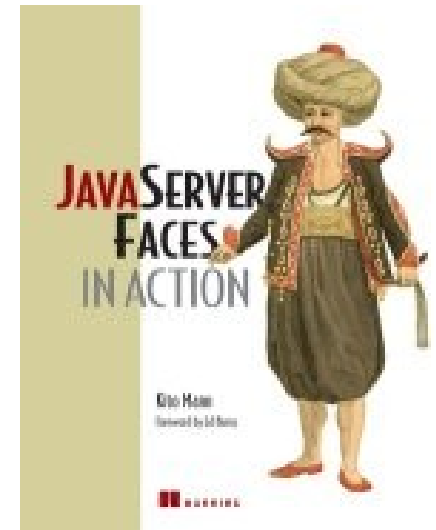
Virtua™
technical excellence. it's that simple.



Learn how JSF transparently supports Ajax development.

Kito D. Mann

- Virtua, Inc
 - Training, consulting, architecture, mentoring, JSF product development
- Author, JavaServer Faces in Action
- Founder, JSF Central
 - <http://www.jsfcentral.com>
- Internationally recognized speaker
 - JavaOne, JavaZone, Javapolis, NFJS, etc.
- JCP Member
 - JSF 2.0, WebBeans, Design-Time API for JavaBeans, Design-Time Metadata for JavaServer Faces Components, Portlets 2.0



Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

Agenda

- **JavaServer Faces from 1,000 feet**
- **Ajaxian Faces**
- **Sprinkling on some Ajax**
- **Ajax inside**
- **Ajax on the outside**
- **Which one?**
- **Rolling your own**
- **Future directions**
- **Summary**
- **Q&A**

What is JavaServer Faces?

- Standard web user interface (UI) framework for Java
 - JSF 1.0/1.1: Standardized through Java Community Process (JCP) in 2004 (JSR 127)
 - JSF 1.2: Standardized through JCP in 2006 (JSR 252)
 - Part of Java EE 5.0

What is JavaServer Faces?

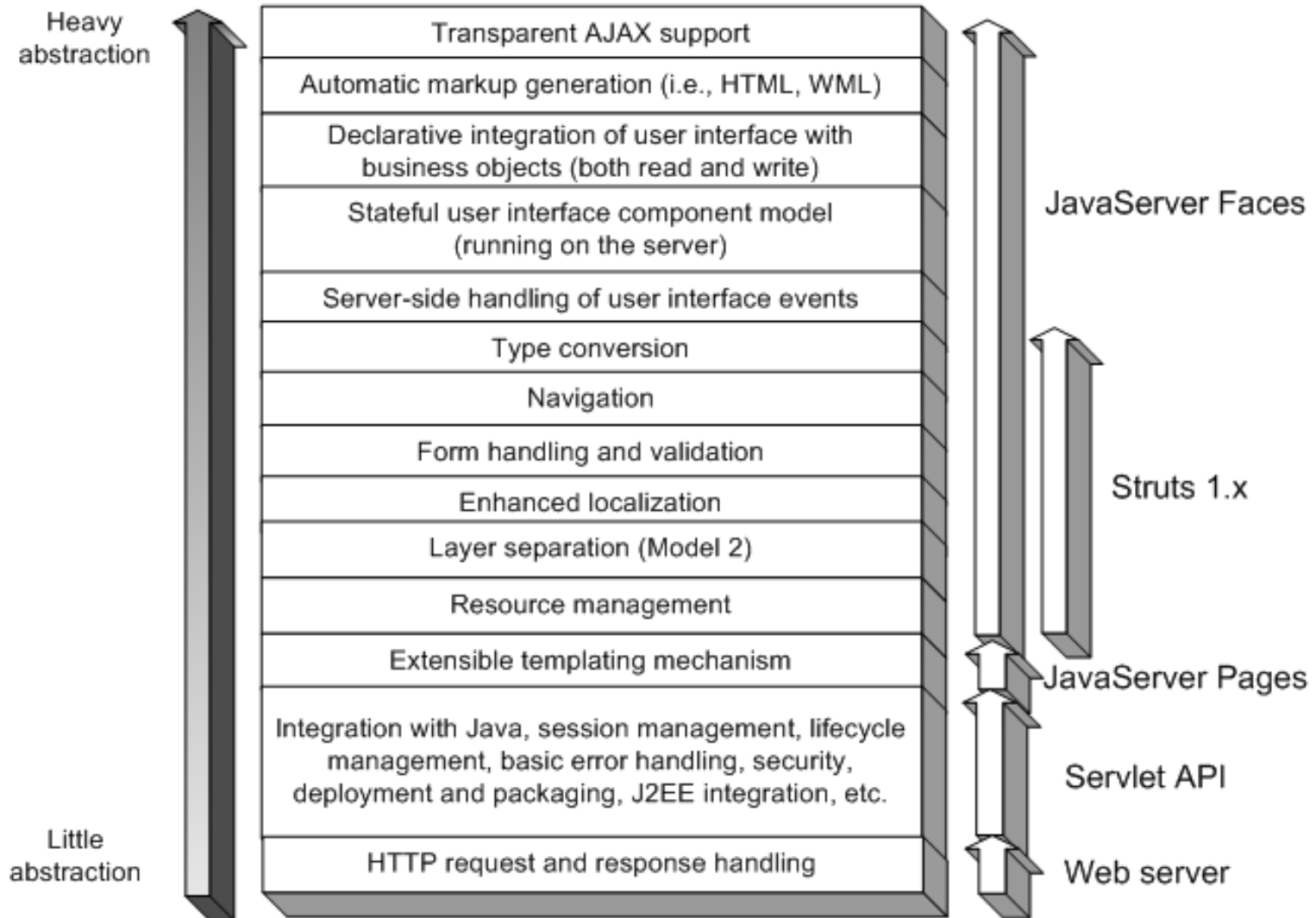
➤ Specification consists of:

- Server side UI component and event model
- Set of basic UI components
- Basic application infrastructure
 - Can automatically synchronize UI components with application objects
 - Includes basic Dependency Injection container

What is JavaServer Faces?

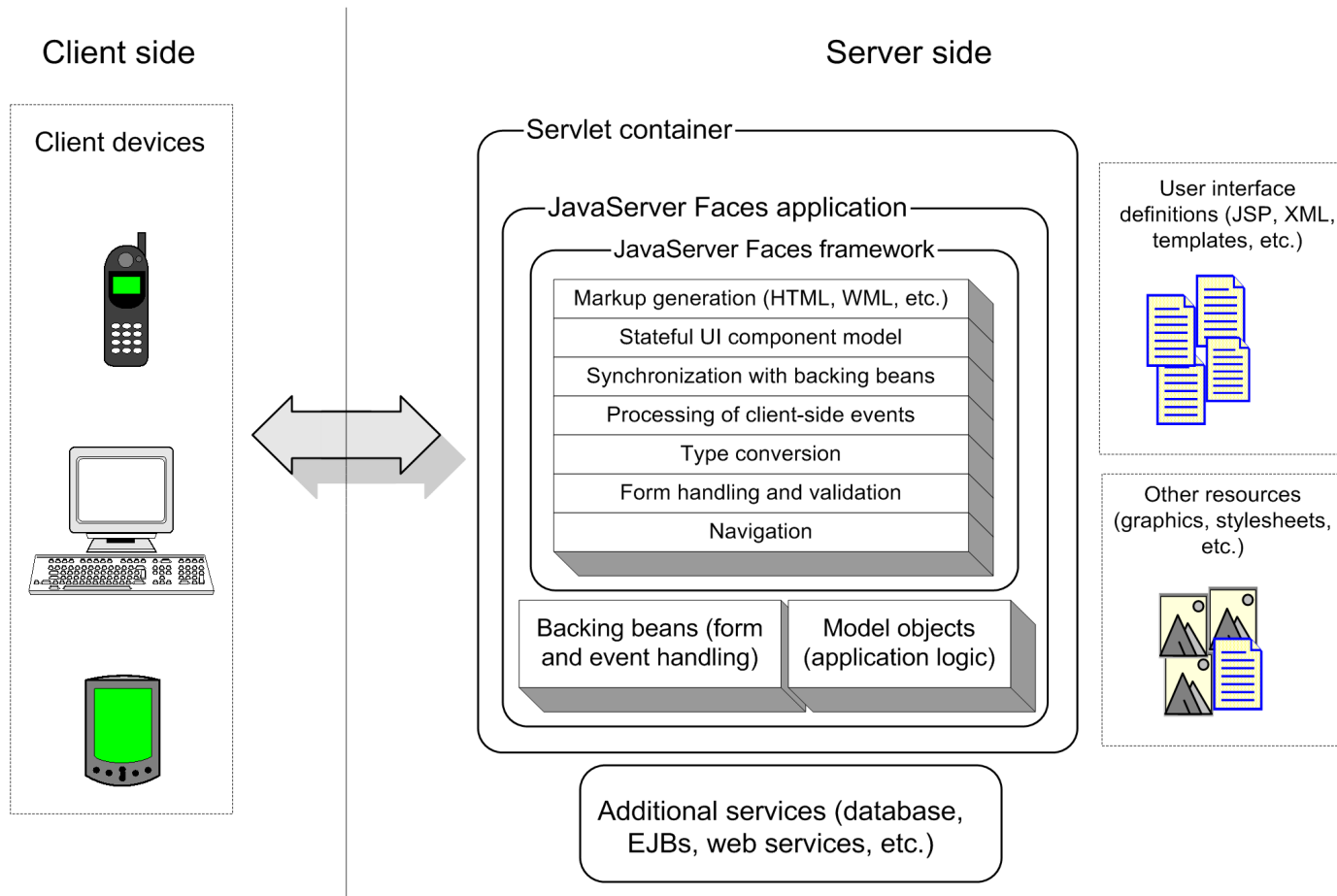
- Extensive tool support
 - Sun, Oracle, IBM, BEA, Exadel, Borland, JetBrains, Genuitec, others
- Enables RAD-style approach to Java web development
- Facilitates third-party UI component market
- Built on top of Servlet API
- Works with JSP, but does not require it

JSF and Struts



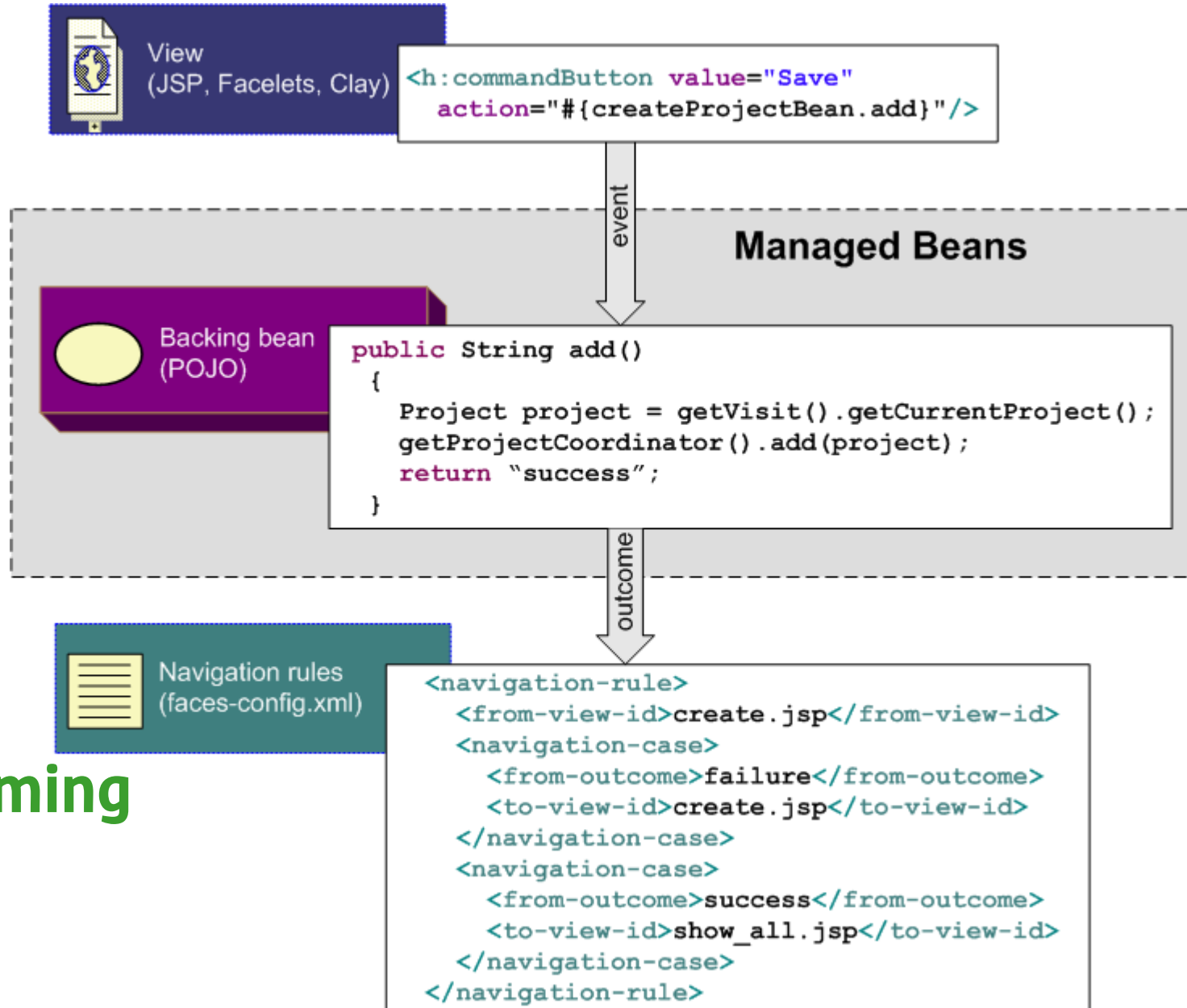
What is Java Server Faces?

JSF Application Architecture



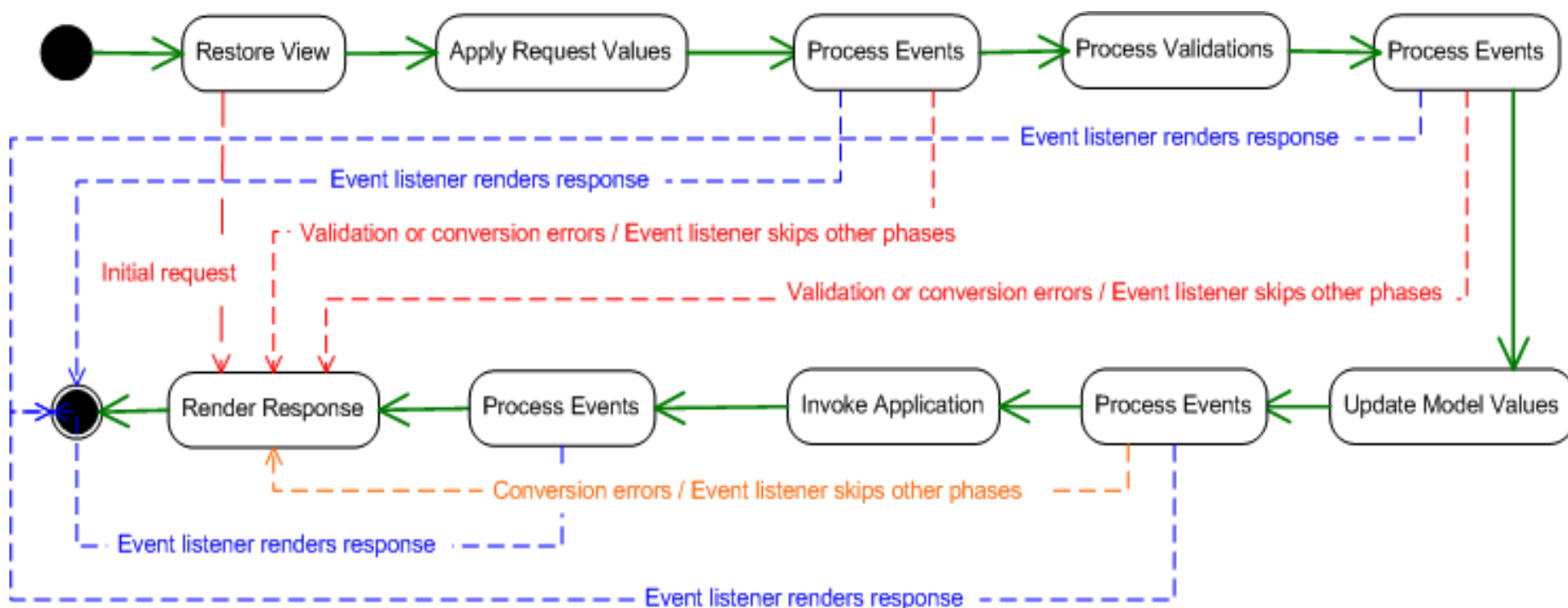
The IDE Effect

- Designed to facilitate easy-to-use IDE integration
 - Since JSF is a standard, vendors more likely to create tools
- All major IDE vendors support JSF
- Level and style varies
- Caveat:
 - Generated artifacts
- Does not *require* tools



JSF Programming Model

The Request Processing Lifecycle



Pluggable Extension Points

- ELResolver
- ViewHandler
- NavigationHandler
- ActionListener
- StateManager
- RenderKits

Agenda

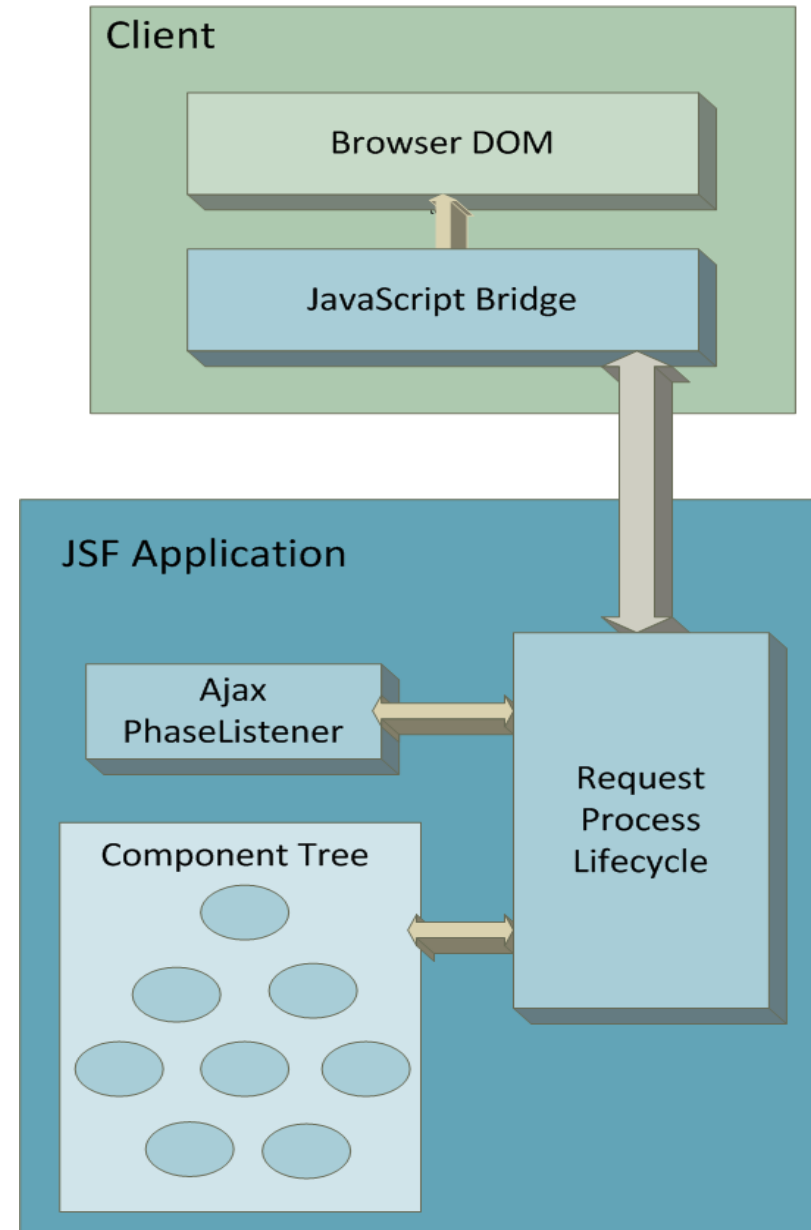
- JavaServer Faces from 1,000 feet
- **Ajaxian Faces**
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

Ajaxian Faces

- JSF's architecture makes it easy to add Ajax support
 - Components and renderers can be separate
 - PhaseListeners can modify request processing
 - Can also use either a Filter or a separate Servlet
 - Pluggable classes can be used to change behavior
 - Enables *transparent* Ajax support
 - Same programming model with or without Ajax

Ajaxian Faces

- JavaScript Bridge sends request
- PhaseListener sends changes
- JavaScript Bridge updates page



Ajaxian Faces

- Ajax-enabled UI components may not be compatible with each other
- No standard for:
 - JavaScript Bridge
 - Determining which components to update
 - Resource resolution

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- **Sprinkling on some Ajax**
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

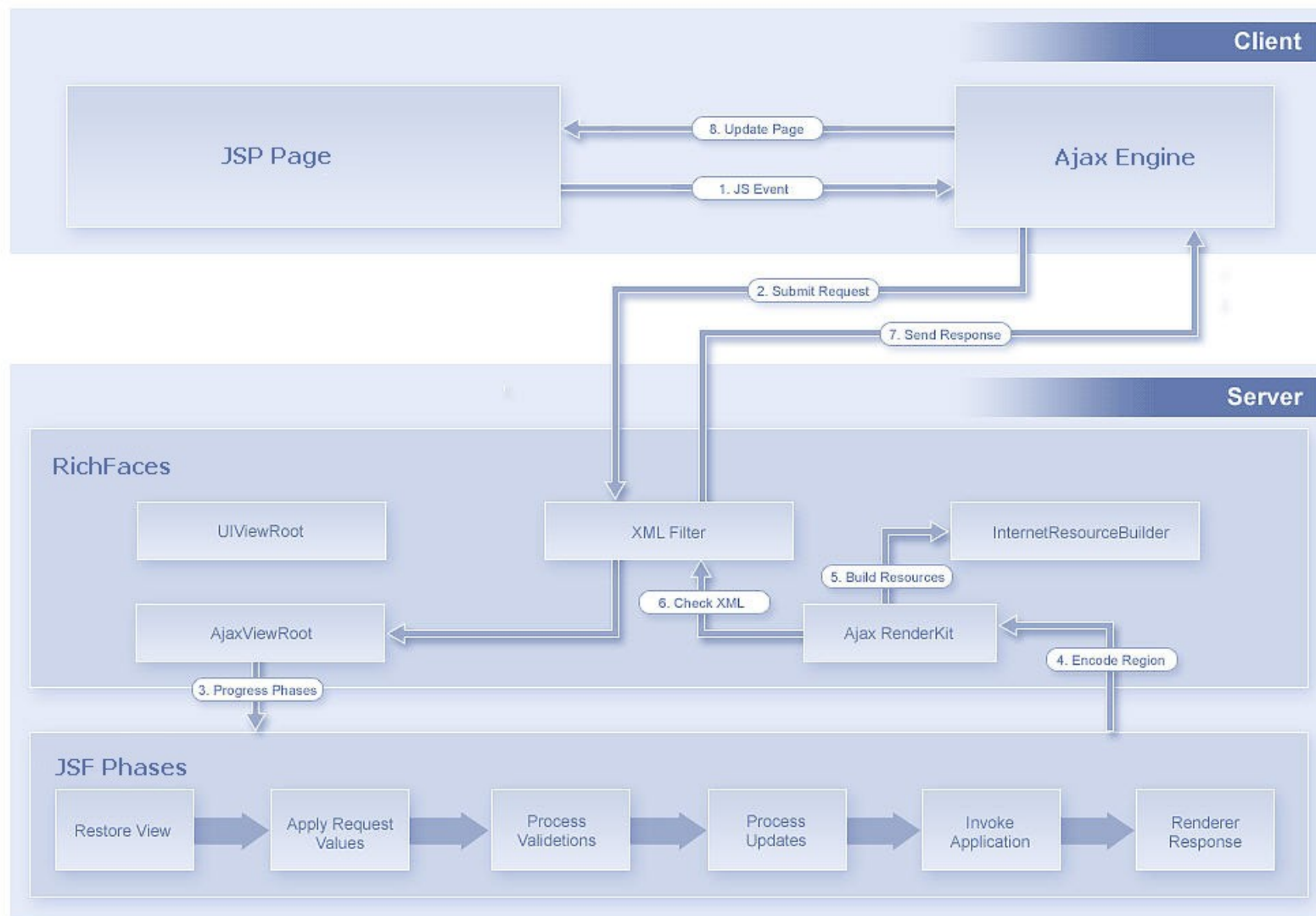
Sprinkling on some Ajax

- Add Ajax to ordinary JSF components
- Specify specific zones or components that need to be refreshed
- JSF event listeners executed asynchronously
- Examples:
 - JBoss RichFaces (formerly Ajax4jsf)
 - AjaxAnywhere
 - DynaFaces
 - Ajax support in MyFaces Sandbox

JBoss RichFaces (Ajax4Jsf)

- Adds Ajax support to any JSF view
- Features:
 - Can add Ajax support to any JSF component with JavaScript events
 - Can define regions of the page (or entire page) that can be updated via Ajax
 - ActionListeners and ValueChangeListeners invoked via Ajax
 - Also features AjaxListeners
- Also has a suite of components

RichFaces



Demo

Ajax4Jsf

Ajax Inside

- Most JSF UI component libraries integrate Ajax support
- Have their own JavaScript and server layers
- Examples:
 - Simplicia ECruiser Ajax Suite for JSF
 - Infragistics NetAdvantage for JSF
 - JBoss RichFaces
 - ICEsoft ICEfaces
 - TeamDev QuipuKit
 - Backbase Java Edition
 - Sun Project Woodstock
 - Apache MyFaces Trinidad

ECruiser Ajax Suite for JSF

- Suite of basic JSF components
 - DataTable, GridTable, Tree, Tab, Menu, etc.
- Integrated Ajax support
 - Lazy loading of data sets
 - Full set of Ajax-aware basic input components
 - Ability to execute any backing bean method for JavaScript events

Demo

ECruiser Ajax Suite for JSF

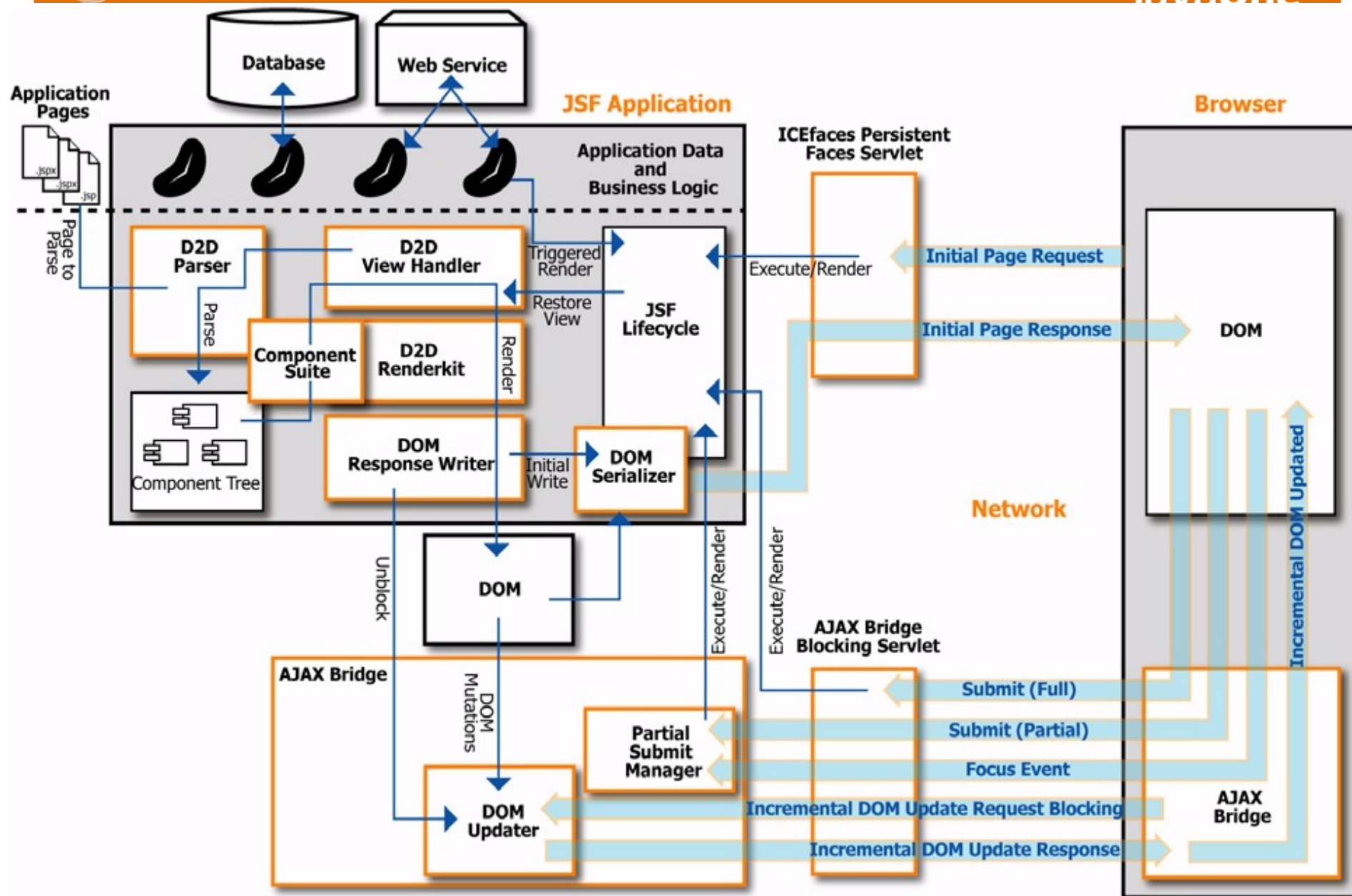
ICESoft ICEfaces

- **Innovative take on Ajax browser/server integration**
 - Direct-to-DOM: maintains copy of DOM on server which is synchronized with DOM on the client
 - Sends only changes when the page changes
 - Requires a persistent connection between client and server
 - Supports Ajax Push/Comet

ICEsoft ICEfaces

➤ Three elements:

- Framework
 - Custom ViewHandler, ResponseWriter, and RenderKit
 - Custom Servlet for blocking requests
- Client-side JavaScript Ajax bridge
- Component suite
 - Chart, Data Table, DynamicImage, Collapsible Panel, Modal Popups, Tabs, TreeTable, Input controls



Demo

ICEsoft ICEfaces

Infragistics NetAdvantage for JSF

- Suite of bread-and-butter components:
 - Tree, Table, Menu, Tabs, MenuBar, Input controls
- Full Ajax support
 - Dynamic loading of data sets
 - Updating part of the page with a selection
 - All components can send Ajax requests and update other components dynamically

Demo

NetAdvantage for JSF

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- **Ajax on the outside**
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

Ajax on the outside

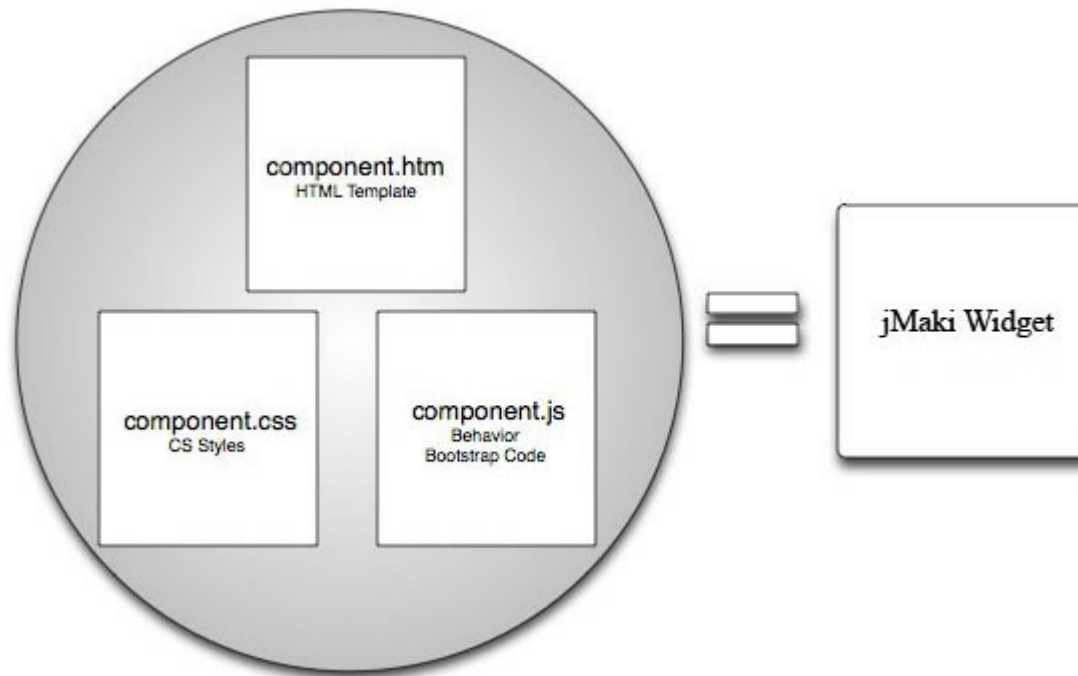
- What about all of those great pure Ajax widgets?
 - Wrap them with JSF components
- Examples:
 - jMaki
 - YUI4JSF
 - DojoFaces
 - Mojarra Scales

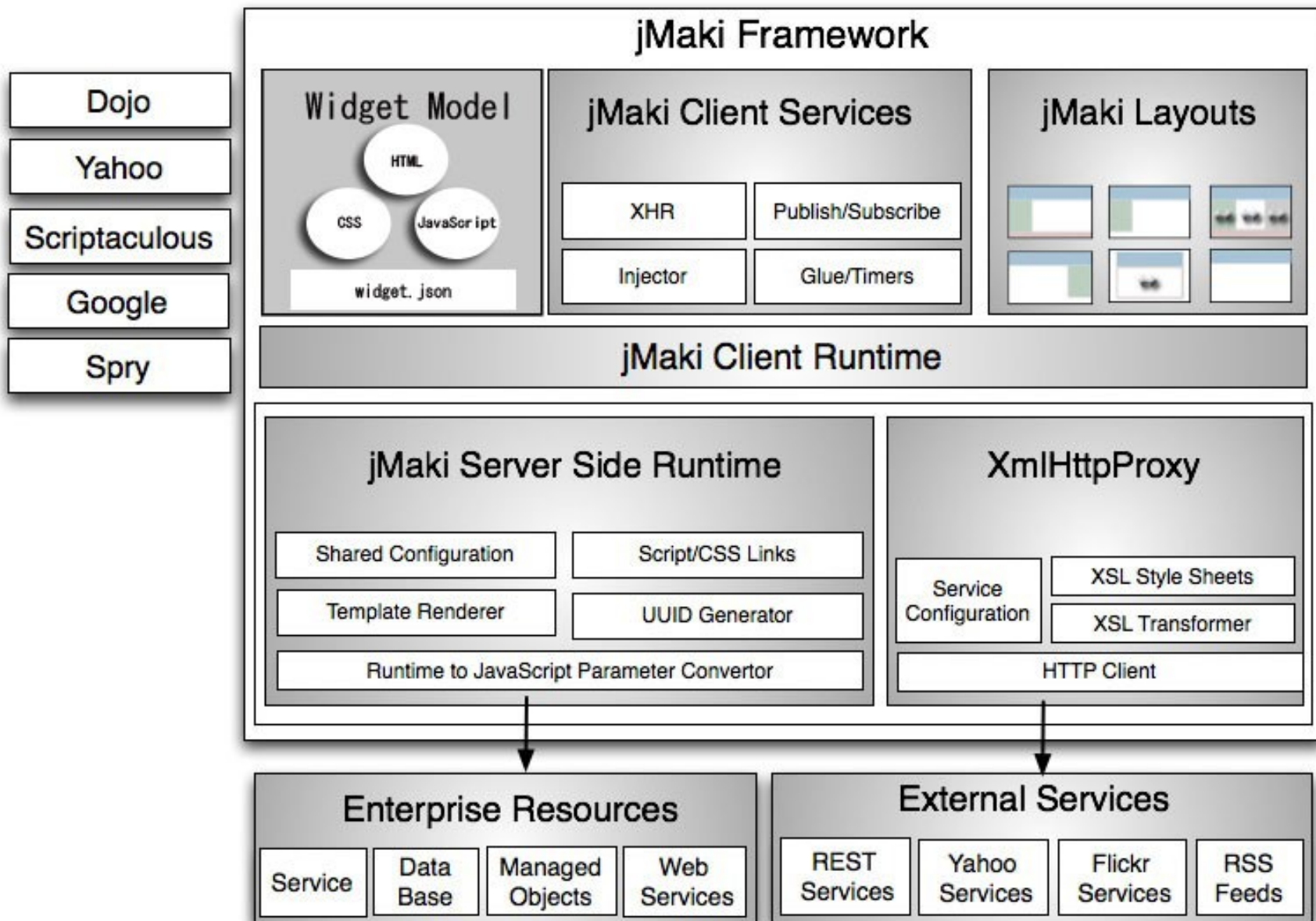
jMaki

- Provides wrappers for popular Ajax widgets
 - Common interface and data model for different libraries
- Supports widgets from:
 - Dojo, Yahoo!, Flickr, Google
- Not as tightly integrated with JSF as other solutions

jMaki

- Easy to create new widgets





Demo

jMaki

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

Which one?

➤ Pick a component suite

- Tight integration with JSF
 - No incompatibility problems within the same suite
- Well documented and supported
- Choose as few as possible (preferably just one)
- How much Ajax do you need?
 - Other suites (such as WebGalileo Faces and Apache MyFaces Tomahawk have some Ajax support as well)

Which one?

- To add support to existing JSF components
 - Use RichFaces/Ajax4jsf
 - Works well with Apache MyFaces Tomahawk
- If you need Ajax push, choose ICEfaces
 - Well documented
 - Open Source
 - Enterprise edition provides support and enhanced scalability
- Use jMaki for eye candy or Web 2.0 components
- Don't forget tools support!

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- **Rolling your own**
- Future directions
- Summary
- Q&A

Rolling Your Own

➤ Loose integration with client-side JavaScript

- Apache Shale Remoting
- Seam Remoting
- Apache MyFaces Trinidad

➤ Custom Ajax components

- Use existing toolkit:
 - JBoss RichFaces CDK
 - Apache Trinidad Maven Plugins
 - JSF Toolkit

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- **Future directions**
- Summary
- Q&A

JSF 1.2

- One problem with JSF 1.1 Ajax support:
 - The request processing lifecycle
- JSF 1.2 improves Ajax support
 - Support for additional Lifecycle instance
 - invokeOnComponent method

JSF 2.0

- Targeted for late 2008
- Part of Java EE 6
- Will incorporate features from:
 - DynaFaces
 - Single, integrated API for Ajax component developers
 - Will include JavaScript bindings
 - Apache Shale
 - JBoss Seam
 - Facelets

JSF 2.0

➤ Key features

- Standardized support for Ajax
- Resource resolution and versioning
- Simplified component development
- Bookmarkable pages
- Page description language (PDL) a la Facelets
- Better UI component interoperability
- New scopes
- Less configuration (annotations)

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- **Summary**
- Q&A

Summary

- **JavaServer Faces is the standard Java EE web framework**
 - Component-based framework
 - Simple, cohesive programming model
- **JSF's architecture lends itself to Ajax development**
- **Ajax development with JSF doesn't require JavaScript!**
 - Same programming model with or without Ajax

Summary

- **Ajaxian Faces**
 - JavaScript layer
 - JSF integration
 - Partial rendering
- **Three approaches:**
 - Adding Ajax to existing components
 - Integrating Ajax into components
 - Wrapping JavaScript widgets

Summary

- Many different component suites and solutions exist currently
- JSF 1.2 makes Ajax integration easier for component developers
- JSF 2.0 will provide an Ajax integration platform for component vendors

Agenda

- JavaServer Faces from 1,000 feet
- Ajaxian Faces
- Sprinkling on some Ajax
- Ajax inside
- Ajax on the outside
- Which one?
- Rolling your own
- Future directions
- Summary
- Q&A

For More Information

- Official JSF site
 - <http://java.sun.com/javaee/javaxserverfaces/>
- JSF Central product directory
 - <http://www.jsfcentral.com/products>
- Ajax component suite comparison
 - <http://www.theserverside.com/tt/articles/article.tss?l=JSFComparator>
- Ajax-JSF Comparison Matrix
 - <http://www.ajaxjsf.net/>

