



Java is a trademark of Sun Microsystems, Inc.



JavaOneSM

A Complete Tour of the JavaServerTM Faces 2.0 Platform

Ed Burns

Sun Microsystems, Inc.
<http://ridingthecrest.com/>

Roger Kitain

Sun Microsystems, Inc.
roger.kitain@sun.com

Presentation Goals

- > *Expose* you to nearly all of the new features in JSF 2.0
- > *Present* a cohesive overview of the current state of the JSF standard
- > *Spark* your interest for learning more



Some of the Faces of Faces



Alexandr Smirnov



Jacob Hookom



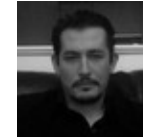
Matthias Weißendorf



Gavin King



Adam Winer



Çağatay Çivici



Keith Donald



Andy Schwartz



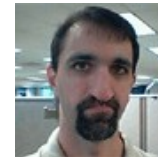
Ken Paulsen



Pete Muir



Craig McClanahan



Joe Ottinger



Dennis Byrne



Jeremy Grelle



Dan Allen



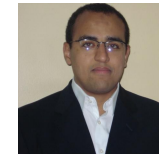
Martin Marinschek



Alexander Jesse



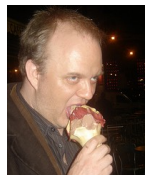
Kito Mann



Hazem Saleh



Roger Keays



Imre Oßwald



Yara Senger



Ted Goddard



Neil Griffin



Rick Hightower



Jason Lee



Amy Fowler



Stan Silvert



David Geary



Lincoln Baxter III




Mike Freedman



Max Katz

Our plan for your time investment

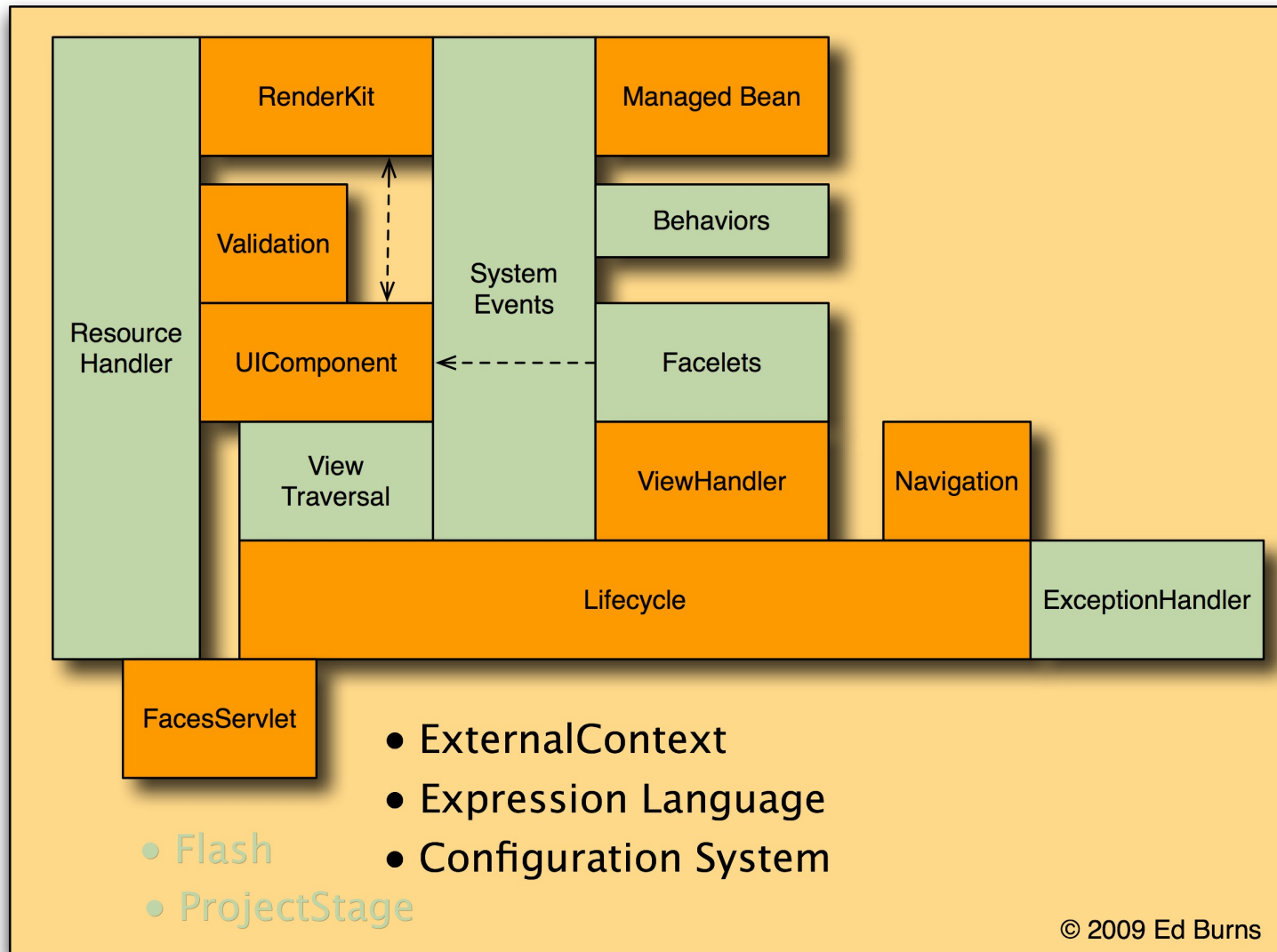
Also known as *The Agenda*

- > Touch on most of the new features in JSF2 in a sensible, organized manner
- > Give you breadth, **not** depth
- > There are no demos during this talk
- > Not enough time! 
- > For demos, see us at booth #567
 - Ed's booth times: Weds 15:00, Thurs 12:00
 - Roger's booth times: Thurs 10:00 13:00

What's New?



Everything!



Look for color coded JavaDocs

<http://javadoc.glassfish.org/javaee6/apidoc/>

Packages	
javax.faces	Top level classes for the JavaServer(tm) Faces API.
javax.faces.application	APIs that are used to link an application's business logic objects to JavaServer Faces, as well as convenient pluggable mechanisms to manage the execution of an application that is based on JavaServer Faces.
javax.faces.component	Fundamental APIs for user interface components.
javax.faces.component.behavior	APIs for attaching additional behavior to user interface components.
javax.faces.component.html	Specialized user interface component classes for HTML.
javax.faces.component.visit	APIs for traversing a user interface component view.
javax.faces.context	Classes and interfaces defining per-request state information.
javax.faces.convert	Contains classes and interfaces defining converters.
javax.faces.el	DEPRECATED Classes and interfaces for evaluating and processing reference expressions.
javax.faces.event	Interfaces describing events and event listeners, and concrete event implementation classes.
javax.faces.lifecycle	Classes and interfaces defining lifecycle management for the JavaServer Faces implementation.
javax.faces.model	Standard model data beans for JavaServer Faces.
javax.faces.render	Classes and interfaces defining the rendering model.
javax.faces.validator	Interface defining the validator model, and concrete validator implementation classes.
javax.faces.view	Classes for defining a View Declaration Language (VDL) for authoring JavaServer Faces user interfaces.
javax.faces.view.facelets	This package contains public classes for the Java code API of Facelets.
javax.faces.webapp	Classes required for integration of JavaServer Faces into web applications, including a standard servlet, base-classes for JSP custom component tags, and concrete tag implementations for core tags.

> New in 2.0

> Changed in 2.0

> Deprecated in 2.0 (JSP)

Categories of New Features

> *Foundational*

- Purpose built to support *Large* features

> *Large*

- Big Ticket Stuff

> *Medium*

- Very useful for advanced users

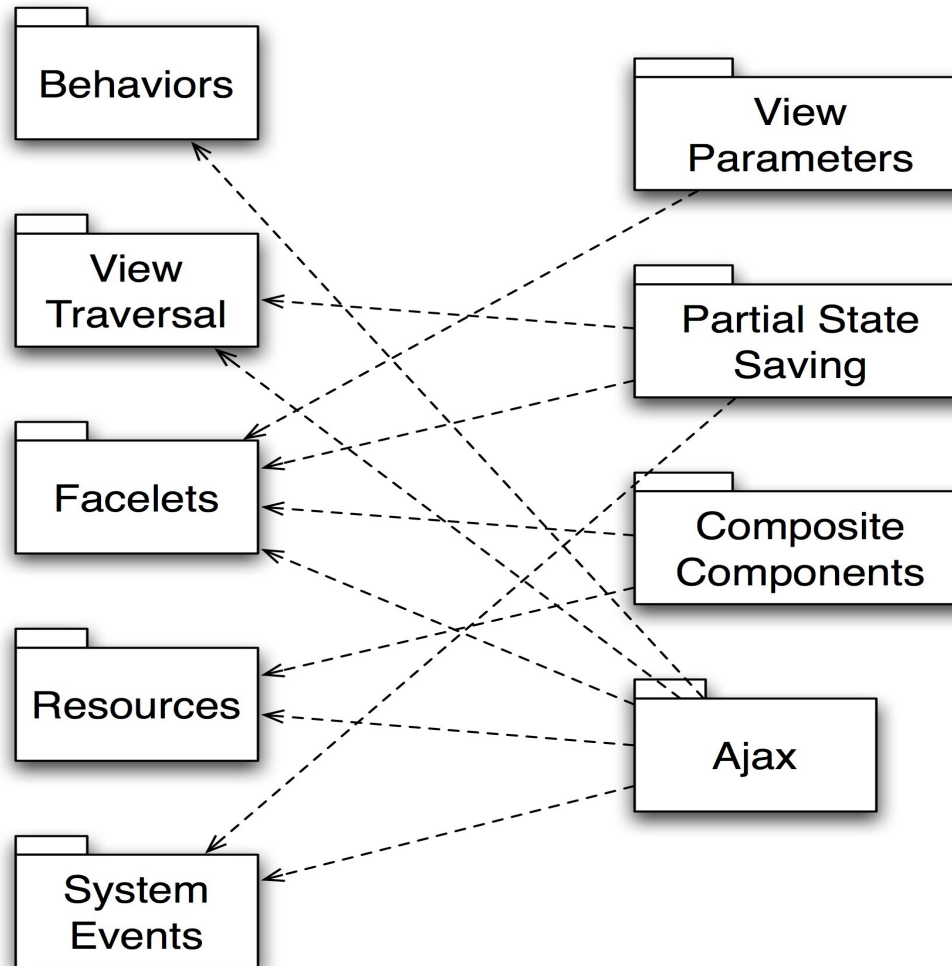
> *Small*

- Bug fixes and enhancements



Hi Level Categories of Features

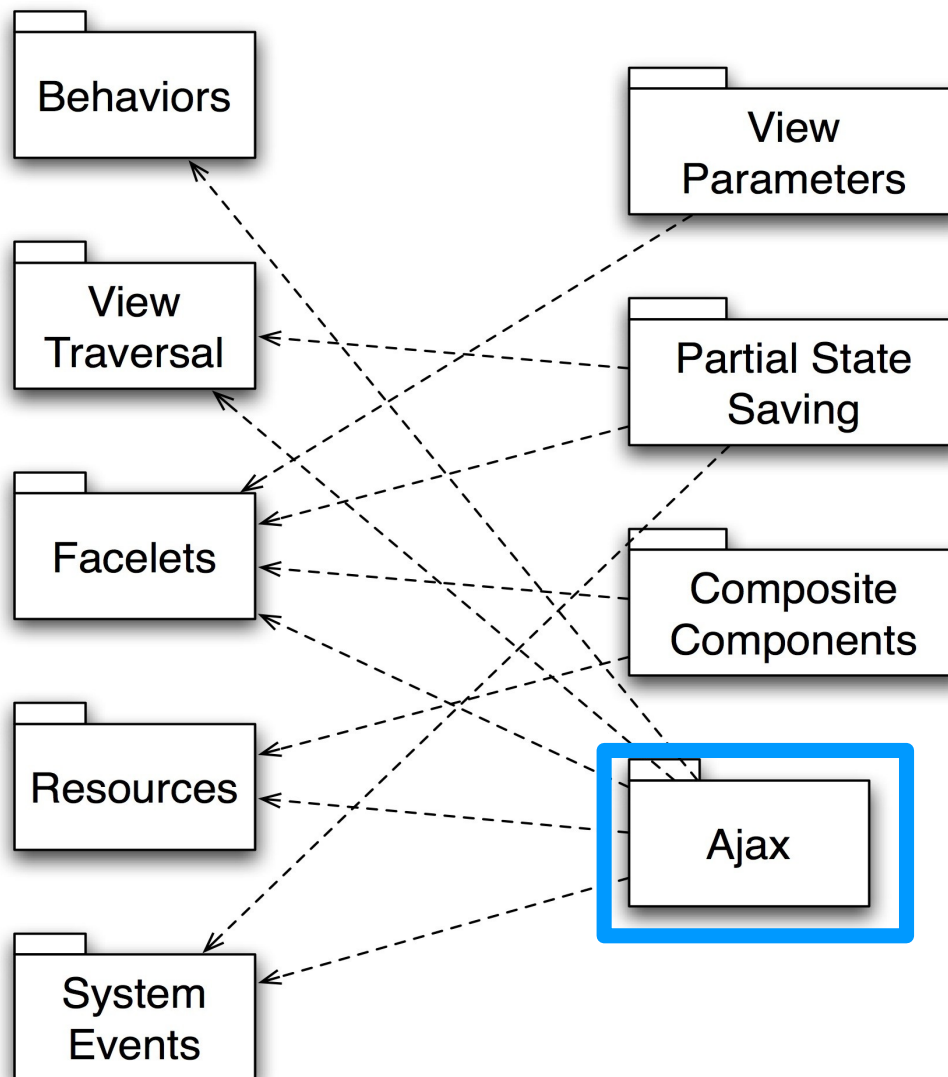
Foundational New
Features



Large New
Features

High Level Categories of Features

Foundational New
Features



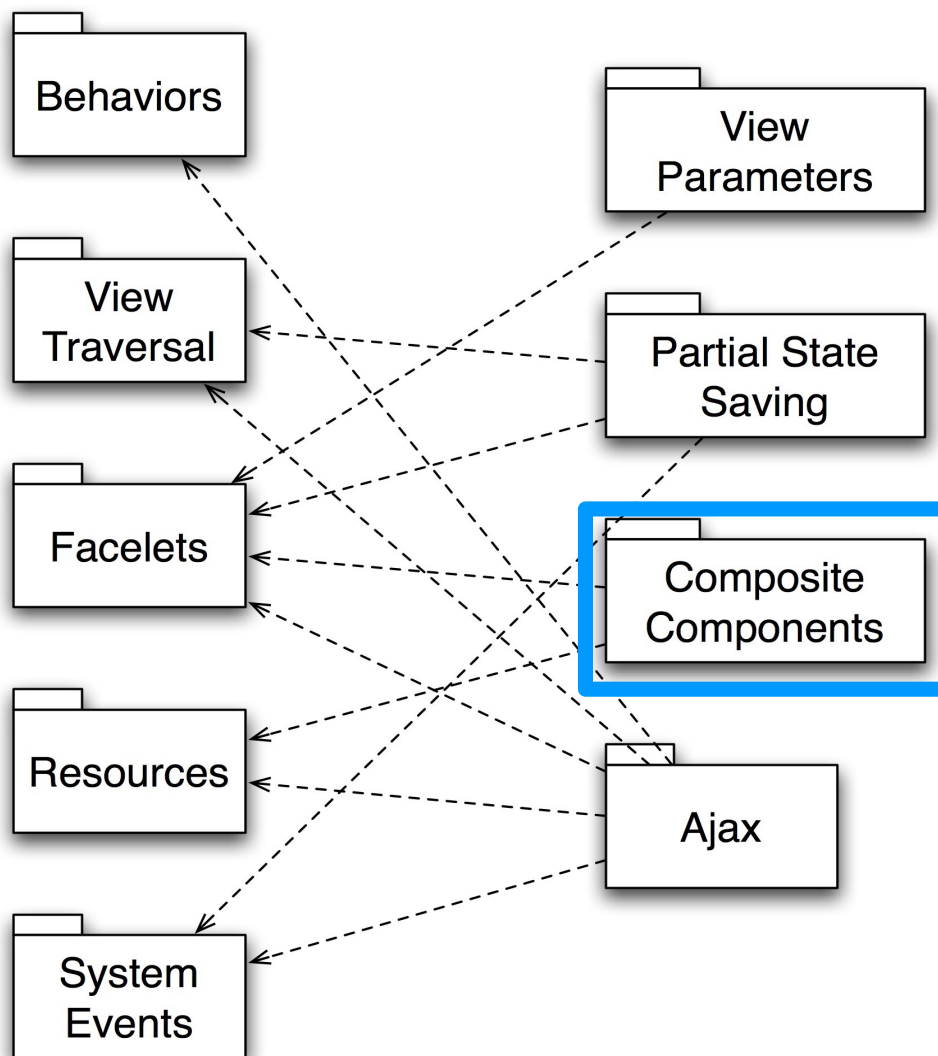
Large New
Features

Ajax

- > Inspiration: RichFaces, IceFaces, DynamicFaces, ADF Faces
- > Two entry points:
 - Declarative: **<f:ajax>** tag, uses **AjaxBehavior**
 - Programmatic ajax: resource library **javax.faces**, resource name **jsf.js**, JavaScript namespace **jsf.ajax**
- > Declarative ajax: useful to ajaxify non ajax pages
- > **AjaxBehavior** : great flexibility for most needs
- > Programmatic ajax API usage pattern
 - **jsf.ajax.request()** JavaScript method

High Level Categories of Features

Foundational New
Features



Large New
Features

Composite Components

>Enable True Abstraction

- Create a true, reusable, OO Black Box Component from an arbitrary region of a page
- Full support for using attached objects in the using page, and mapping them to arbitrary targets within the composite component
- Full support for Faces Event System

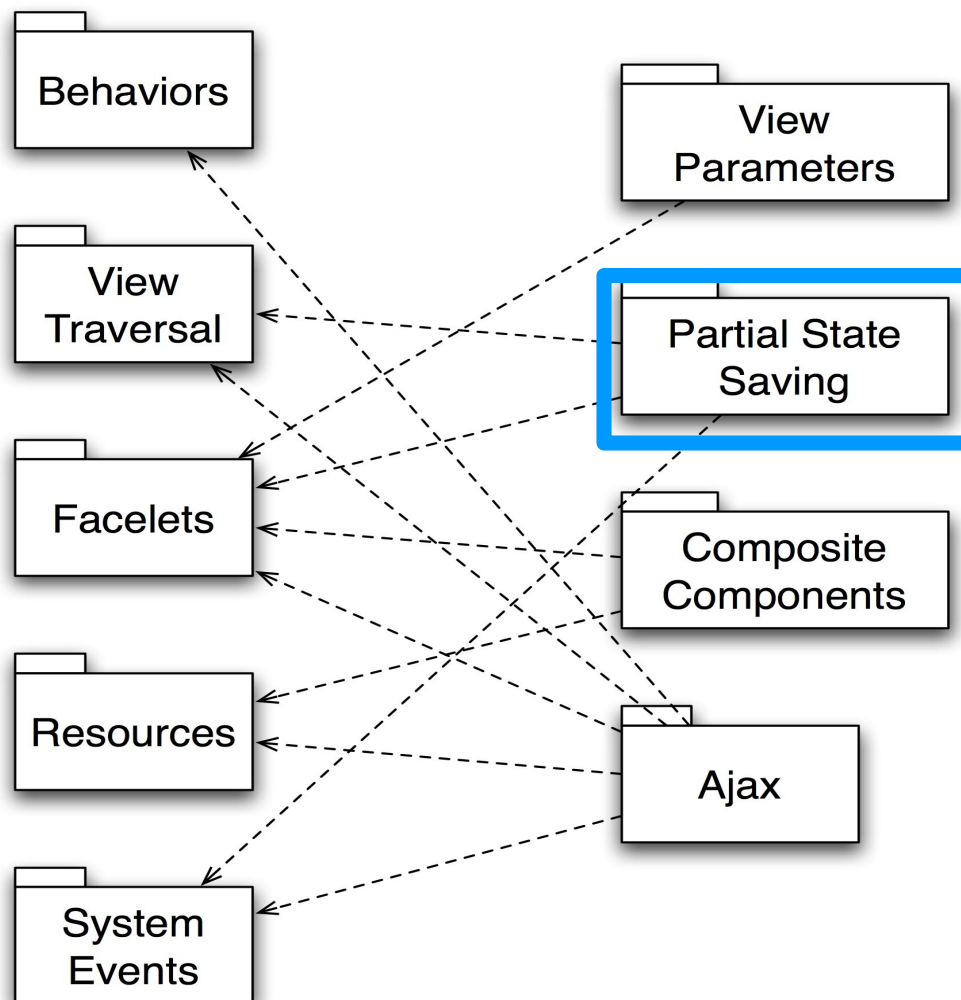
>Heavily leverages naming conventions

>Builds on top of Resources and Facelets

>Top Level Component may be written in script, Mojarra supports Groovy.

High Level Categories of Features

Foundational New
Features



Large New
Features

Partial State Saving

- > Default for pages written with Facelets for JSF 2.0
- > Of interest to Java custom component authors
- > All standard components implement this feature - your composite components will automatically take advantage of it.
- > Per-view state size up to 4X smaller than before
- > **PartialStateHolder** behavioral interface, extends existing **StateHolder**.
- > Inspired by Trinidad state saving
- > **StateHelper**, from **UIComponent**, does most of the work: replace ivars with keys within **StateHelper**.

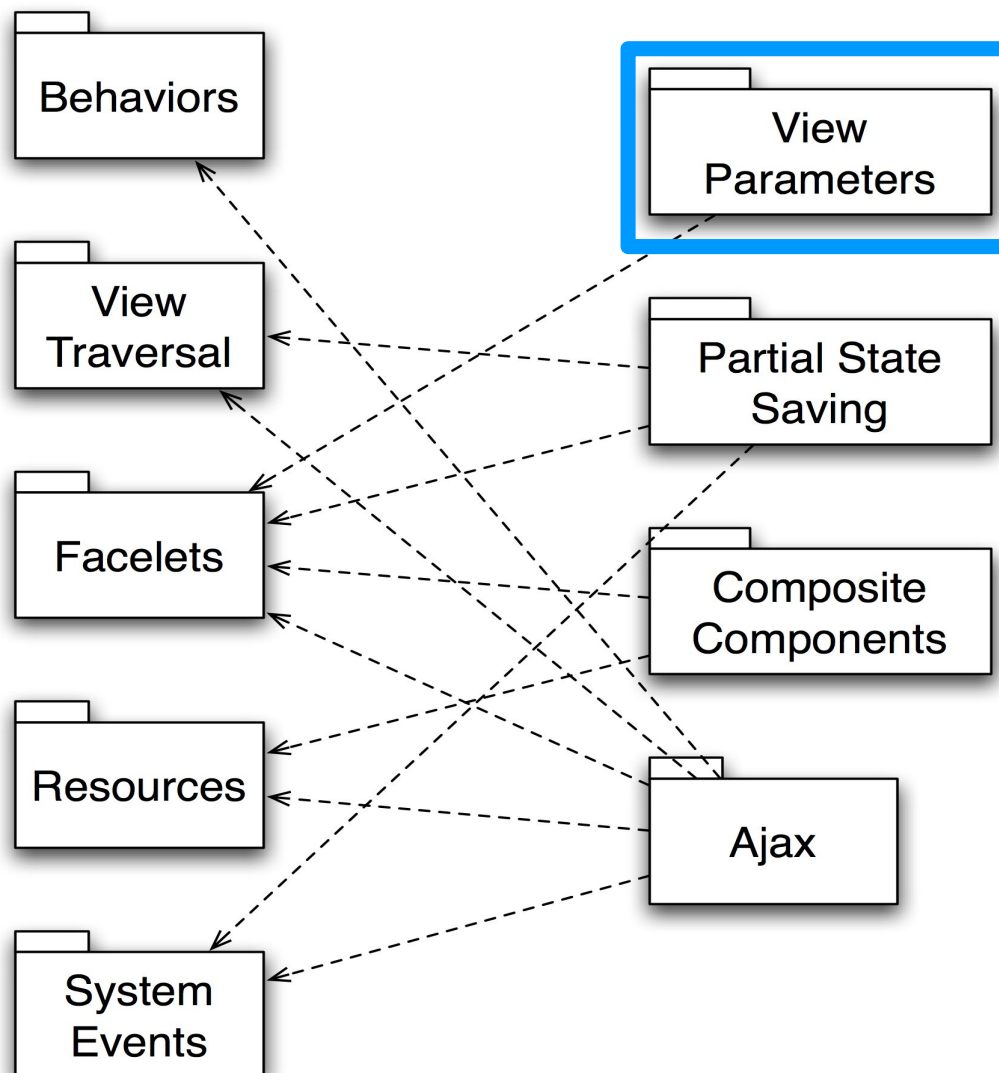
Simplifies custom components

```
public class MyInput extends UIComponentBase {  
    private enum Keys { name }  
    public String getName() {  
        return getStateHelper().get(name);  
    }  
    public void setName(String nameParam) {  
        getStateHelper().put(name, nameParam);  
    }  
}
```

High Level Categories of Features

Foundational New Features

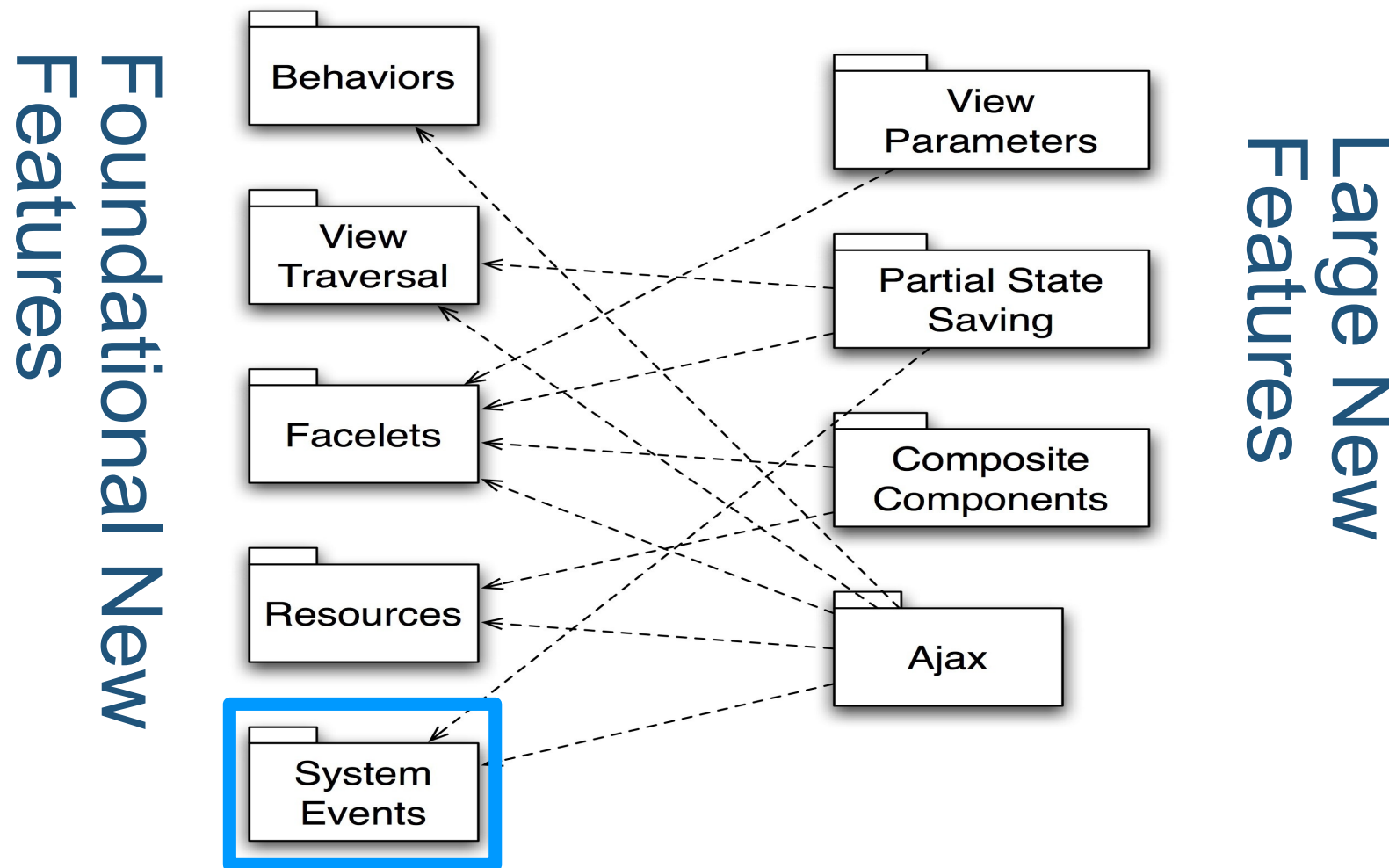
Large New Features



View Parameters

- > Inspired by Page Parameters from JBoss Seam
- > Provides a way to map request parameters to special components within the view
- > **<f:metadata><f:viewParam></f:metadata>**
- > **UIViewParameter** extends **UIInput**
- > All attached objects valid for **UIInput** are valid
- > Breaks the tyranny of POST
- > Lays the foundation for Bookmarkable pages

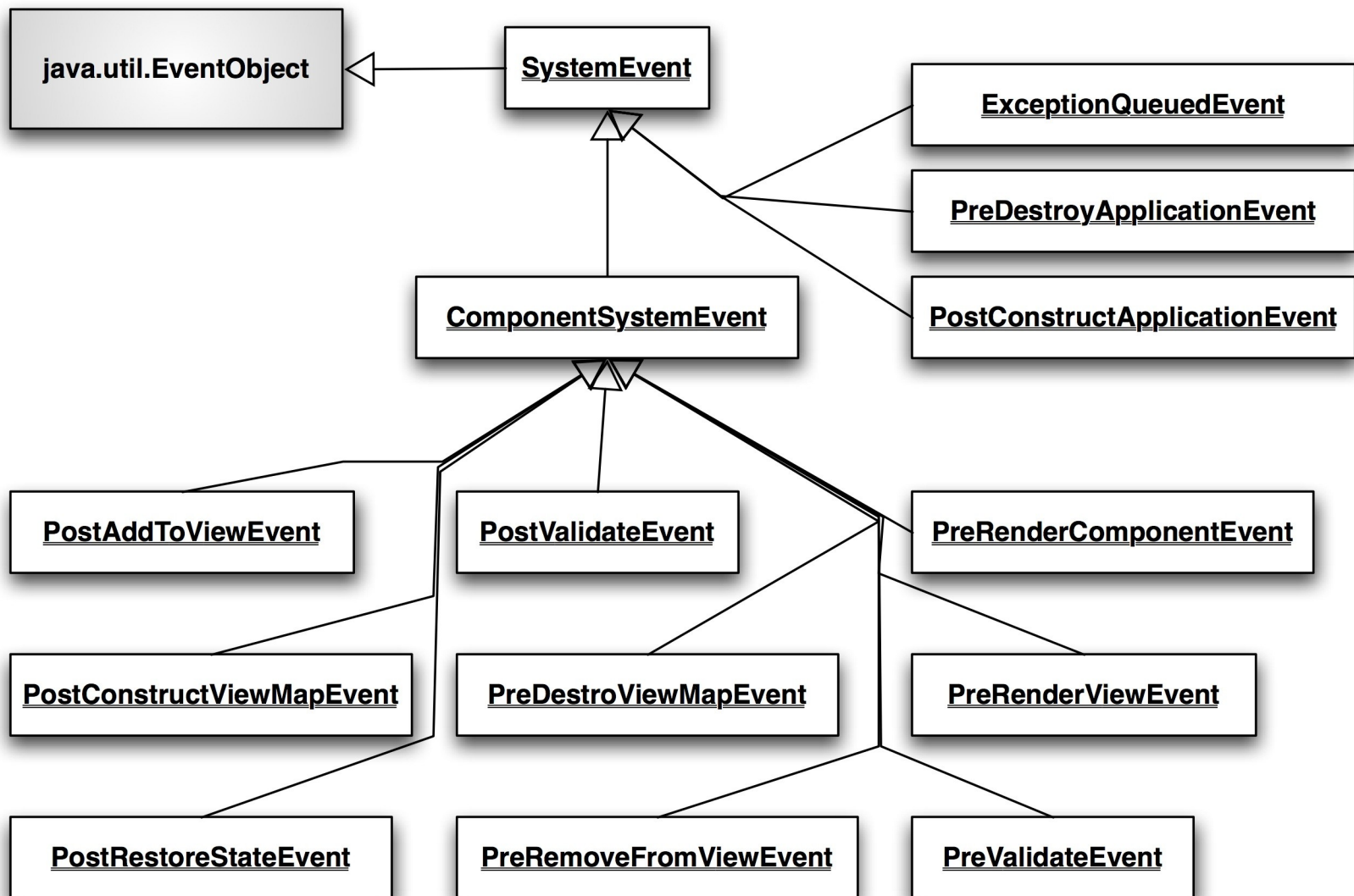
High Level Categories of Features



System Events

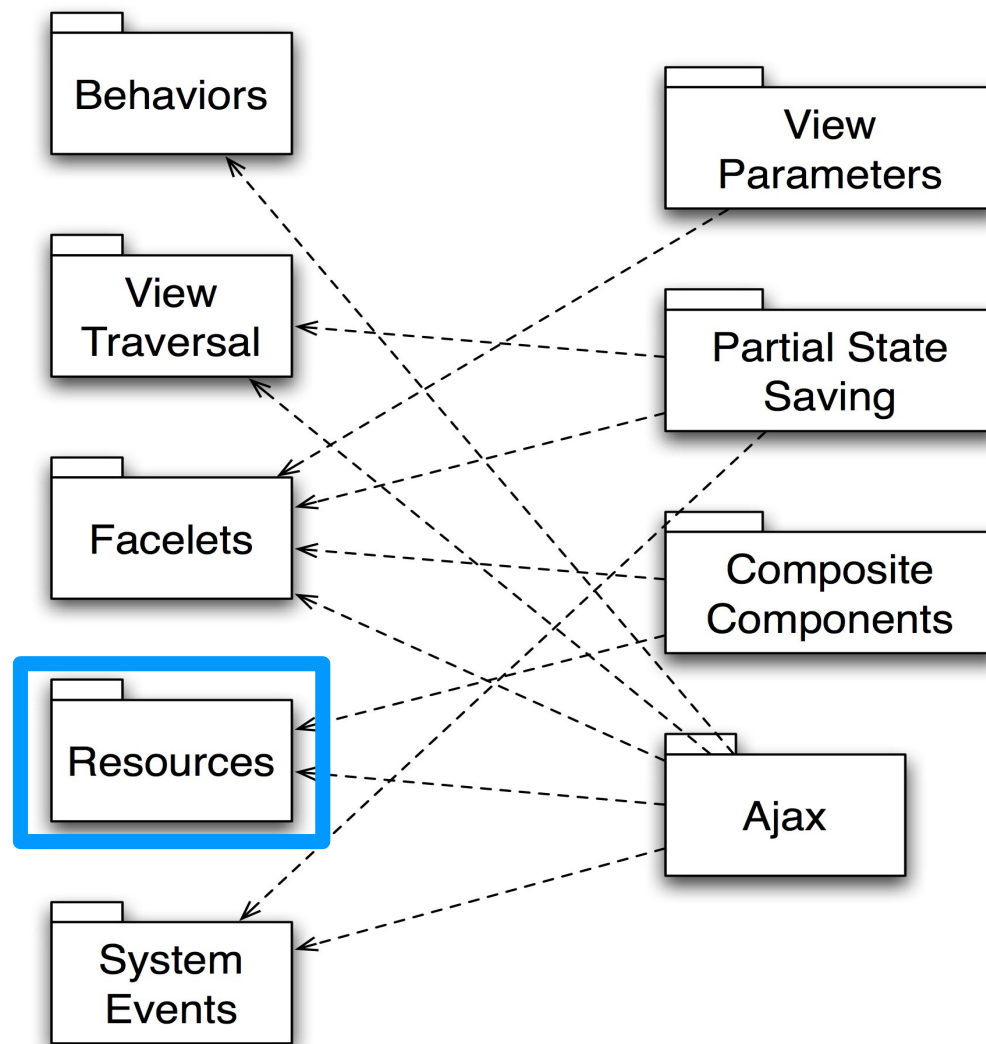
- > Influenced by JSFTemplating
- > Publish/Subscribe event bus for things that happen during the JSF Lifecycle, **not application specific**
- > Inspired by Solaris Dtrace, Linux strace, truss, etc.
- > Listeners can be registered at three scopes
 - component **UIComponent.subscribeToEvent()**
 - view **UIViewRoot.subscribeToEvent()**
 - application **Application.subscribeToEvent()**
- > Publish is always with **Application.publishEvent()**

System Events: Standard Event Types



High Level Categories of Features

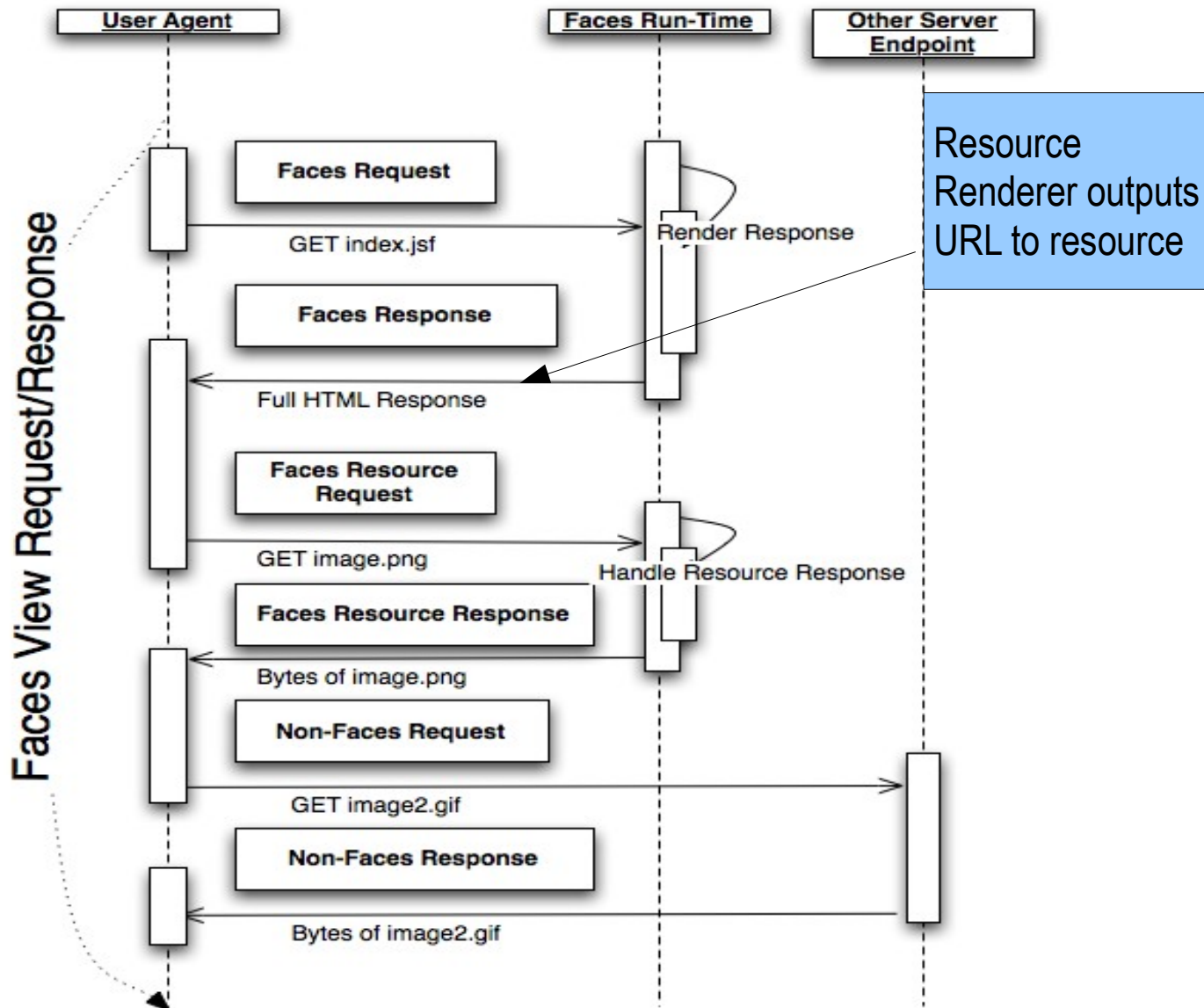
Foundational New
Features



Large New
Features

Resources

- > Avoid need for separate Filter or Servlet.
- > Resources are logically related to components, treat them that way.
- > Load from Classpath, or filesystem
- > Full support for “Library” concept, I18N, Versioning



Resources: Java API

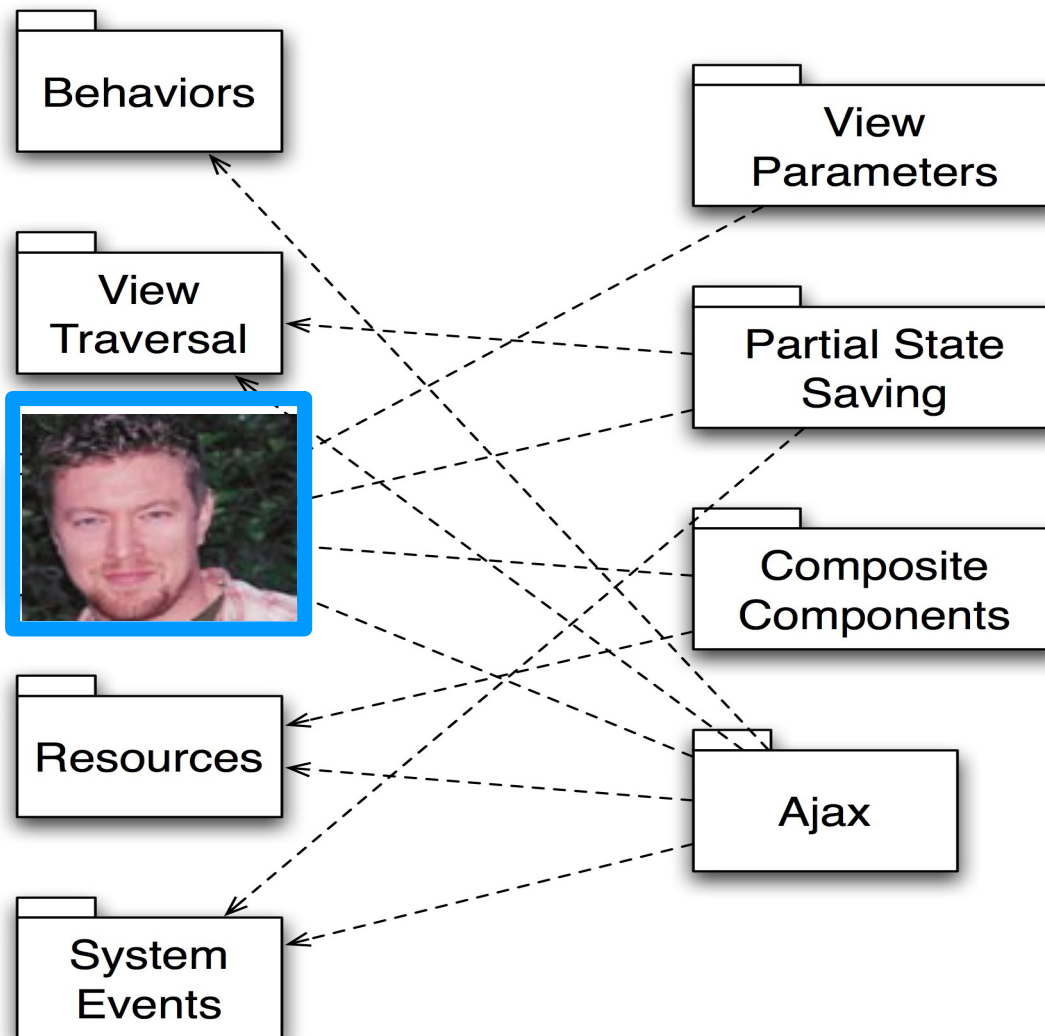
- > **@ResourceDependency** or **@ResourceDependencies** annotation on
 - **UIComponent, Renderer, Validator, Converter, ClientBehavior**
- > attributes for resource name, library name
- > attribute for target: head, body, form
- > Built in support for CSS, Image, JavaScript resources

Resources: Markup API

- > **<h:head>, <h:body>**
- > **<h:outputScript>, <h:outputStylesheet>**
- > Similar attributes for resource name, library name
- > Similar attribute for target: head, body, form

High Level Categories of Features

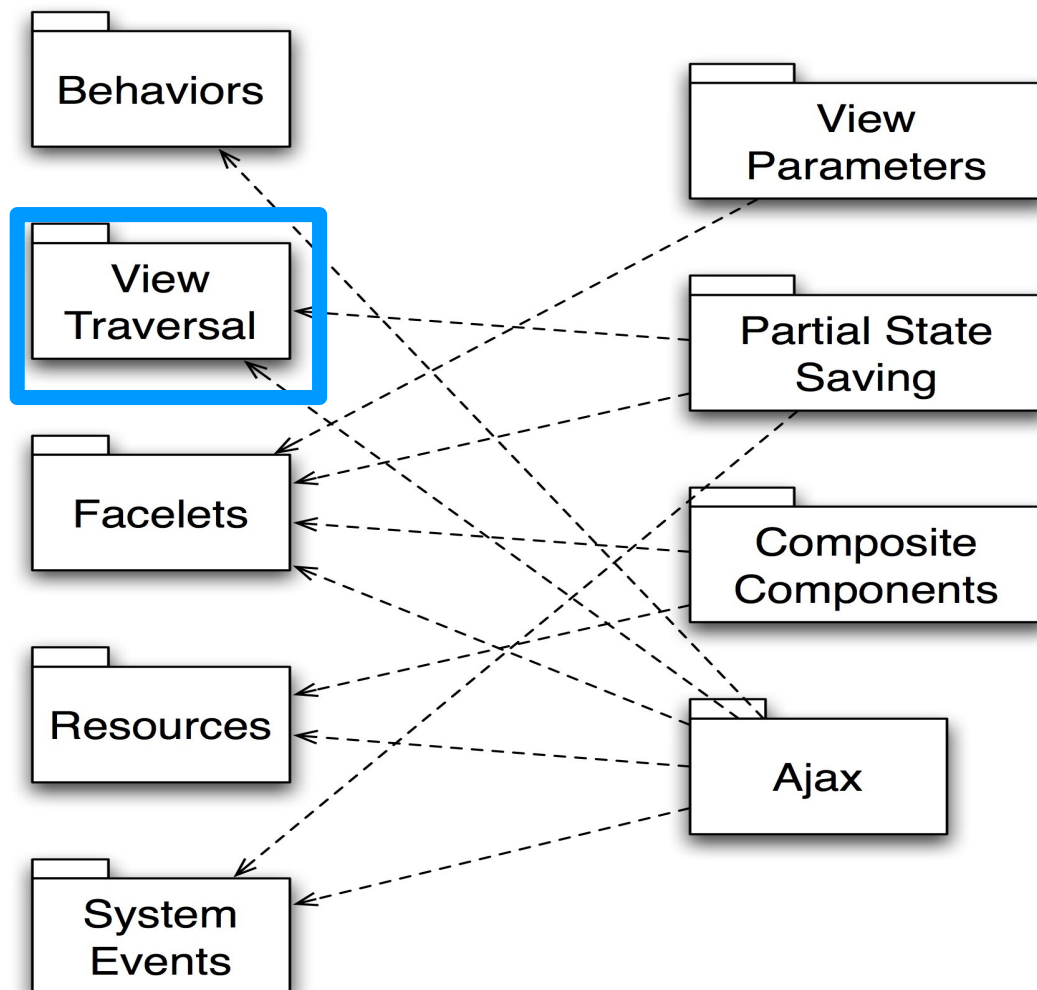
Foundational New Features



Large New Features

High Level Categories of Features

Foundational New
Features



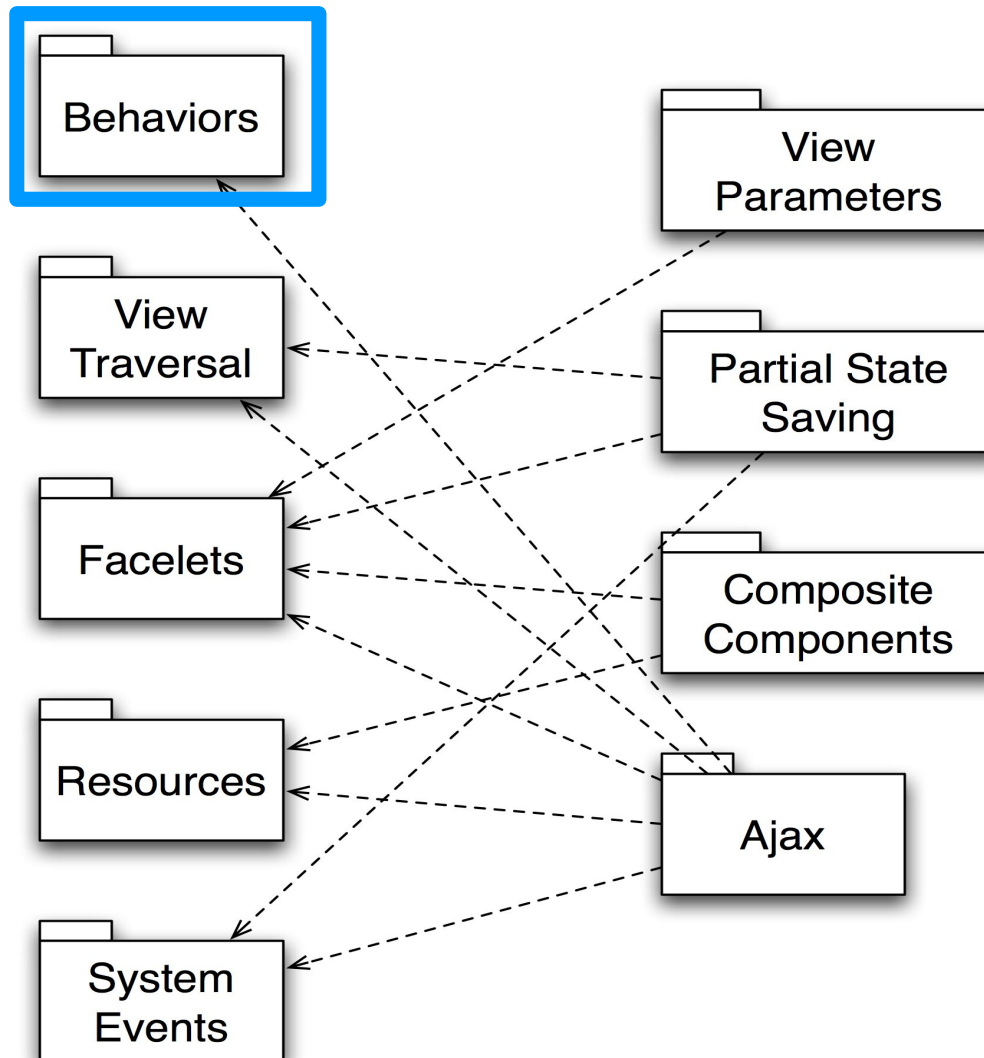
Large New
Features

View Traversal

- > **VisitContext.createVisitContext()**
- > **UICollection.visitTree(VisitContext)**
- > Used on Ajax requests for partial traversal
- > Used by new **StateManagementStrategy**

High Level Categories of Features

Foundational New
Features



Large New
Features

Behaviors

- > A **Behavior** is an interface and invents a new type of “attached object” which takes part in decode/encode.
- > **ClientBehavior** extends **Behavior**
- > Define things that happen in the view, in the browser
- > Currently only one standard **ClientBehavior** implementation: **AjaxBehavior**
- > New interface: **ClientBehaviorHolder**, implemented by all components in **javax.faces.component.html**
- > Special **ClientBehaviorRenderer**, does not extend **Renderer**.
 - **getScript()**, **decode()**

Small New Features

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

New
Scopes

Validation

EL

Exceptions

Navigation

Medium Sized New Features

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

Small New Features

Navigation

>Bookmarkable Pages

- Uses View Parameters feature to ensure request params are validated before rendering page
- New components **<h:link>**, **<h:button>**, cause the browser to submit GET request, contrast with **<h:command{Button,Link}>**

>Implicit Navigation

- If you don't want all that **<navigation-rules>** stuff, you can put the navigation logic straight in the page.

>ConfigurableNavigationHandler

- Query navigation rules at run-time

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

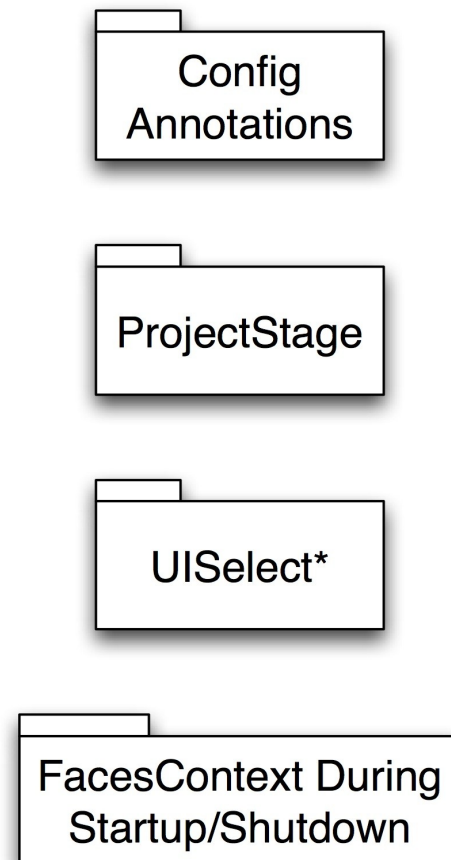
Small New Features

Exceptions

- > Exceptionhandler
 - can be decorated
 - all interesting exceptions flow through the exception handler!
 - new exception type: UpdateModelException
 - exceptions during update model don't queue faces-messages
- > **ValidatorException** contains multiple **FacesMessages**

High Level Categories of Features

Medium Sized New Features



Small New Features

EL

- > **`#{component}`**, **`#{compositeComponent}`**: get the “currently” processed component / composite component
- > **`#{component.clientId}`**
- > **`#{component.messageList}`**
- > Make `ActionEvent` param optional for action-listeners
- > For EE6, EL method invocation supports parameters
 - makes `f:setPropertyActionListener` obsolete

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

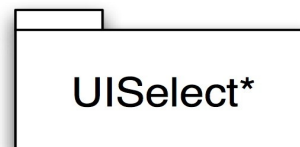
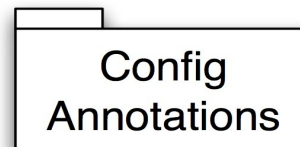
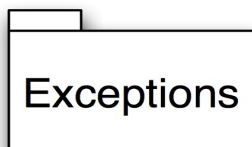
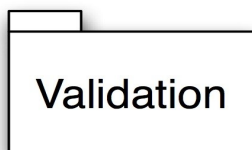
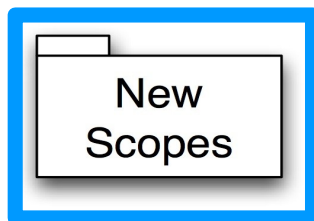
Small New Features

Validation

- > Integration with JSR 303: Bean Validation
 - **@NotEmpty private String name;**
 - default validator: javax.faces.Bean – automatically applied to all input fields
- > Default-validator: hook up a validator for all instances of EditableValueHolder
- > preValidate/postValidate system events

High Level Categories of Features

Medium Sized New Features



Small New Features

New Scopes

- > “conversation” comes from Java Contexts and Dependency Injection (JSR 299) nee WebBeans
- > Flash: inspired by Ruby on Rails
 - accessible from EL or ExternalContext
 - useful for storing data for use on the “next” request
 - preserves FacesContext-messages across redirects
- > View: persists as long as user interacts with the same view
 - managed-beans can be put in view-scope
- > `FacesContext.getAttributes()`

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

Small New Features

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

Small New Features

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

Small New Features

High Level Categories of Features

Medium Sized New Features

New
Scopes

Validation

EL

Exceptions

Navigation

Config
Annotations

ProjectStage

UISelect*

FacesContext During
Startup/Shutdown

Small New Features

Petite New Features

> ExternalContext

- **cookie, headers, realPath, contentLength, scheme, serverName, port, invalidateSession, responseReset**

> FacesContext

- **currentPhaseId, messageList, validationFailed, postback**

> RenderKit

- **componentFamilies, rendererTypes**

> Wrappers for decoratable artifacts

> DataModel.iterator

Petite New Features

- > **javax.faces.DATETIMECONVERTER_DEFAULT_TIMEZONE_IS_SYSTEM_TIMEZONE**
- > **EditableValueHolder.resetValue()**
- > **UIViewRoot.getPhaseListeners()**
- > **SEPARATOR_CHAR_PARAM_NAME**,
UINamingContainer.getSeparatorChar(),
NamingContainer.SEPARATOR_CHAR deprecated

Summary

- > Open community – many contributors
- > Big ticket features:
 - Ajax, Composite Components, Partial State Saving, View Parameters
 - Facelets replaces JSP
- > Far easier to use
 - faces-config.xml optional
 - Component creation is trivial
- > Well worth a second look!

There's More....

- > JavaServer™ Faces Platform and Ajax: State of the Union
 - BOF 4869
 - Wednesday – 6:45 PM Esplanade 307-310
- > Writing a JavaServer™ Faces 2.0 Component That Uses Ajax: It's Easy! (Really, It's Easy.)
 - BOF 4146
 - Wednesday – 7:45 PM Esplanade 307-310

There's More....

Join us at the JCP Community Party

- 19:30 - 22:00
- Hotel Intercontinental,
Pacific Terrace, 4th Floor
888 Howard Street



GlassFish Community

Open Source and Enterprise Ready



- **GlassFish v3 Preview Available now!**

- Java EE 6 reference implementation
- Modular OSGi architecture – easy to develop & deploy
- Runs in-process and easy to extend
- Support for Ruby-on-Rails, Groovy and Grails, Python and Django

- **GlassFish v2 – Production Ready**

- Best price/performance open source App server with Clustering, High Availability, Load Balancing
- Secure, Reliable, Transactional, .NET-interop Web svcs
- Support for Ajax and Comet

- **GlassFish ESB**

- SOA and Business Integration platform

- **GlassFish Communications App Server**

- SIP servlet technology for converged services

glassfish.org

- **24x7 Enterprise and Mission Critical Support**

- sun.com/appserver

- **Tools Integration**

- NetBeans and Eclipse

- **Pavilion booth numbers: 550, 566, 567**

- **Meet Java EE spec leads and experts at Ancillary Event & Booth**



JavaOneSM

Thank You

Ed Burns

<http://xri.net/=edburns>

Roger Kitain

roger.kitain@sun.com

