



# JavaOne™

[java.sun.com/javaone](http://java.sun.com/javaone)

## The Limitations of JavaServer™ Faces Technology and How To Overcome Them

Shay Shmeltzer, Group Manager, Oracle

TS-7890

**ORACLE®**  
JDEVELOPER



Learn how to be successful when using  
JavaService Faces™ technology



GOAL

# JavaServer Faces Technology In 60 Seconds

## ➤ Simplifying Web Development

- Server side UI generation
- View and controller (MVC)
- Component based not mark-up
- Automatic event and state handling
- Diverse client base not just HTML
- Designed with tooling in mind
- Applicable to wide spectrum of programmer types

## ➤ Part of Java™ Platform, Enterprise Edition 5 (Java EE 5)

# Where is JavaServer Faces Technology Today?

- JavaServer Faces 1.2 application – Part of Java EE 5
- Tools
  - Free: JDeveloper, NetBeans™ software, Eclipse
  - Commercial: BEA, IBM, Codegear, MyEclipse ....
- JavaServer Faces Components
  - Free: Apache Trinidad, ICEFaces, and others
  - Commercial: Oracle ADF Faces, Backbase, and others
- JavaServer Faces Technology Implementations
  - Sun RI
  - Apache MyFaces

# Limitations You Might Be Hitting

- Templating
- More components
- Ajax
- Simpler component development
- More scopes
- Extended jsf-config.xml
- Bookmarking
- Reusable Flows

# Templating

- Problem: JavaServer Faces application doesn't contain a templating mechanism
- Solutions:
  - Facelets
  - JSFTemplating
  - Oracle ADF Faces RC

# Facelets Sample

template.xhtml x
page.xhtml

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:ui="http://xmlns.jcp.org/jsf/facelets" >
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>template</title>
  </head>
  <body>
    <b><ui:insert name="title">Default Title</ui:insert></b>
  </body>
</html>

```

template.xhtml
page.xhtml

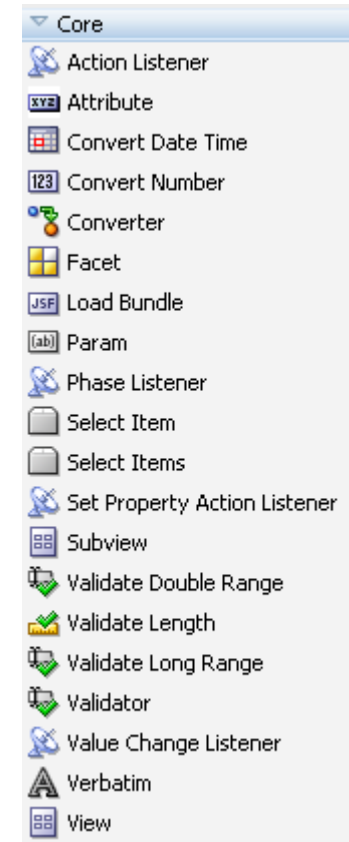
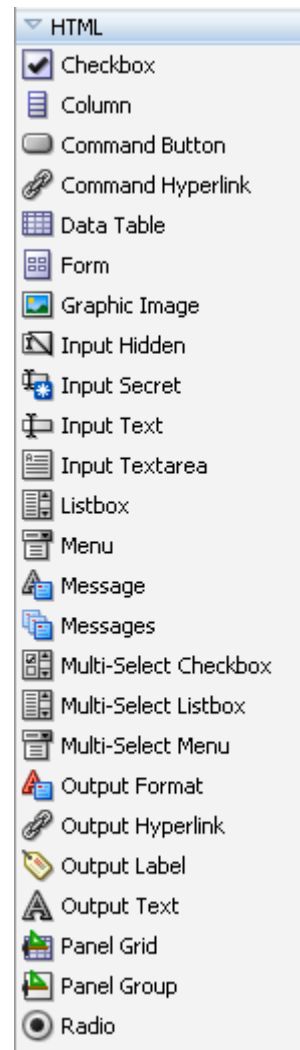
```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:ui="http://xmlns.jcp.org/jsf/facelets" >
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>page</title>
  </head>
  <body>
    <ui:composition template="/template.xhtml">
      <ui:define name="title">
        I'm thinking of a number from Can you guess it?
      </ui:define>
    </ui:composition>
  </body>
</html>

```

# More Components

- JavaServer Faces RI Components
- Is that enough?





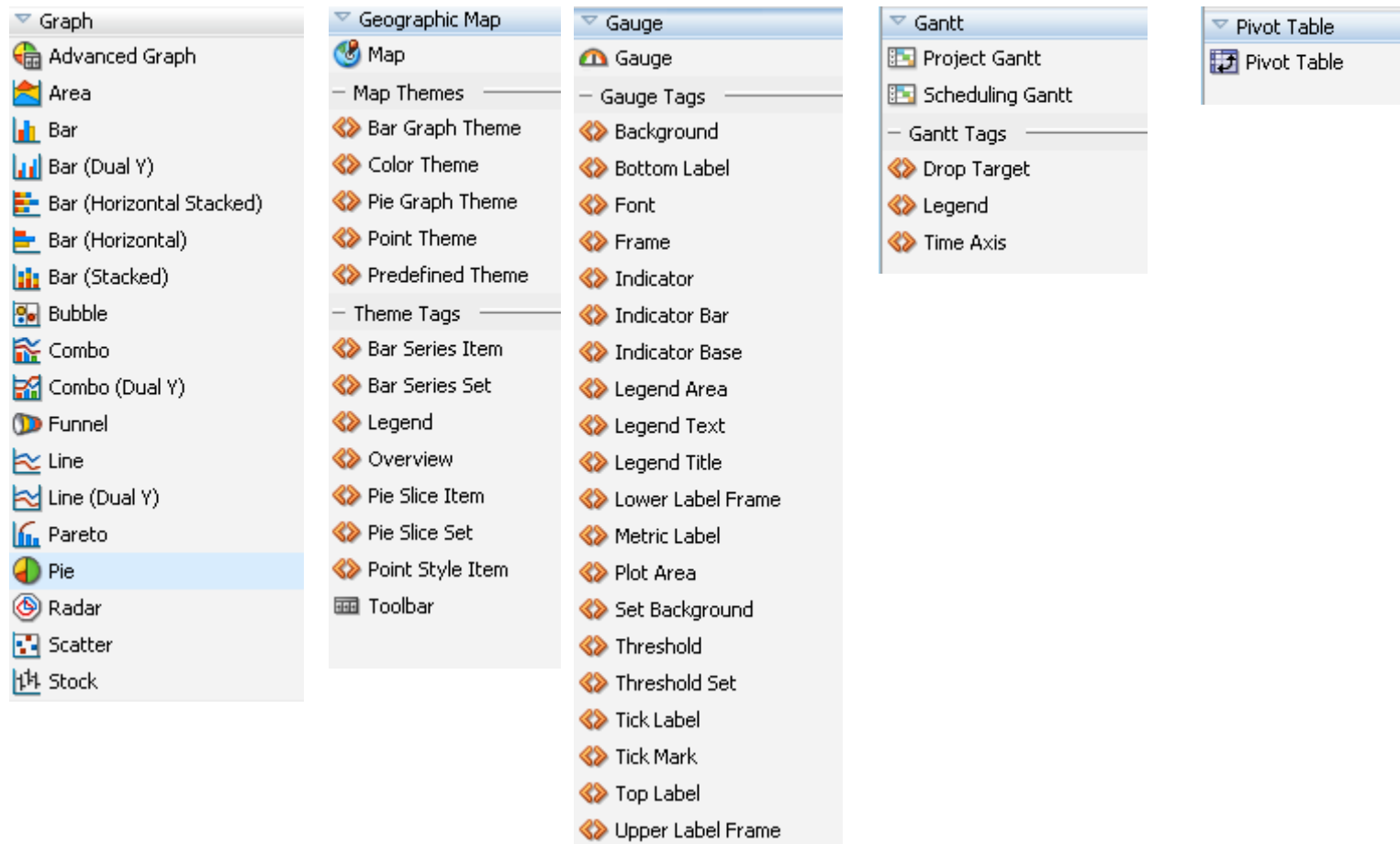
# And What About Ajax?

- JavaServer Faces RI Components are not enough
- Solutions:
- Components – Trinidad, Oracle ADF Faces RC, Backbase, ICEFaces, RichFaces, WoodStock and many more
- Watch out for interoperability between components – especially with Ajax components

# ADF Faces Components

Layout	Operations	Common Components	
Decorative Box	Attribute Drag Source	Active Image	Link
Document	Attribute Drop Target	Active Output Text	Media
Inline Frame	Client Attribute	Active Toolbar Button	Menu
Navigation Pane	Client Listener	Bread Crumbs	MenuItem
Panel Accordion	Collection Drag Source	Button	Message
Panel Border Layout	Collection Drop Target	Choose Color	Messages
Panel Box	Component Drag Source	Choose Date	Navigation Item
Panel Collection	Convert Color	column	Output Formatted
Panel Form Layout	Convert Date Time	Dialog	Output Label
Panel Group Layout	Convert Number	Facet Ref	Output Text
Panel Header	Data Flavor	Form	Panel Label and Message
Panel List	Drop Target	Go Button	Panel Menu Bar
Panel Splitter	Export Collection Action Listener	Go Link	Popup
Panel Stretch Layout	File Download Action Listener	Image	Progress Indicator
Panel Tabbed	For Each	Image Link	Query
Panel Window	Group	Input Color	Quick Query
Separator	Iterator	Input Combobox List Of Values	Reset Button
Show Detail Header	Poll	Input Date	Rich Text Editor
Spacer	Reset Action List	Input File	Select Boolean Checkbox
Toolbox	Return Action List	Input List Of Values	Select Boolean Radio
	Rich Text Editor	Input Number Slider	Select Item
	Server Listener	Input Number Spinbox	Select Many Checkbox
	Set Action List	Input Range Slider	Select Many Choice
	Show Popup Beh	Input Text	Select Many Listbox
			Select Many Shuttle
	Validate Byte Length		Select One Choice
	Validate Date Restriction		Select One Listbox
	Validate Date Time Range		Select One Radio
	Validate Double Range		Select Order Shuttle
	Validate Length		Show Detail
	Validate Long Range		Show Detail Item
	Validate Reg Exp		Status Indicator
			Subform
			Table
			Toolbar
			Toolbar Button
			Train
			Train Button Bar
			Tree
			Tree Table

# And More...



# Demo

## JavaServer Faces Technology Templates and Ajax with ADF Faces

# Looking Beyond The Components

- Evaluate component sets on more than just the components
- For example, ADF Faces RC also offers:
  - Drag-and-drop framework
  - Dialog and popup framework
  - Navigation menu framework
  - Partial page rendering
  - Active data framework
  - Advanced data streaming
  - Complete JavaScript™ technology-based API

# Simpler Components Creation

- Problem: Creating JavaServer Faces components is not trivial
- Solution
- Hopefully in JavaServer Faces 2.0 application
- ADF - Declarative JavaServer Faces Component Creation

# Missing Scope

- Problem: No scope between session and request
- Solutions:
  - ADF Faces/Trinidad – processScope
  - Shale – Dialog
  - Seam – Conversation
  - Apache MyFaces Orchestra

# Shale Dialog

Action Method:

```
FacesContext context = FacesContext.getCurrentInstance();  
DialogContextManager manager =  
    DialogHelper.getDialogContextManager(context);  
DialogContext dcontext = manager.create(context, "foo");  
dcontext.start();
```

JavaServer Faces technology page:

```
<h:inputText id="name" ... value="#{dialogScope.name}"/>
```



# ADF Faces/Trinidad ProcessScope

Action Method - setting:

```
AdfFacesContext afContext = AdfFacesContext.getCurrentInstance();  
afContext.getProcessScope().put("detailEmployee", emp);
```

Or in a JavaServer Faces page:

```
<af:setActionListener from="#{employee}"  
    to="#{processScope.employeeDetail}"/>
```

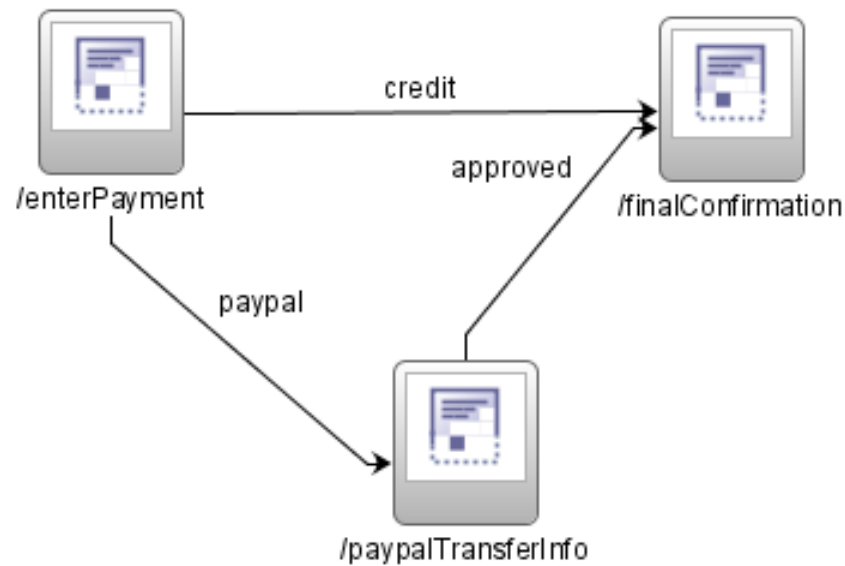
JavaServer Faces page – getting :

```
<h:outputText value="#{processScope.detailEmployee.name}"/>
```

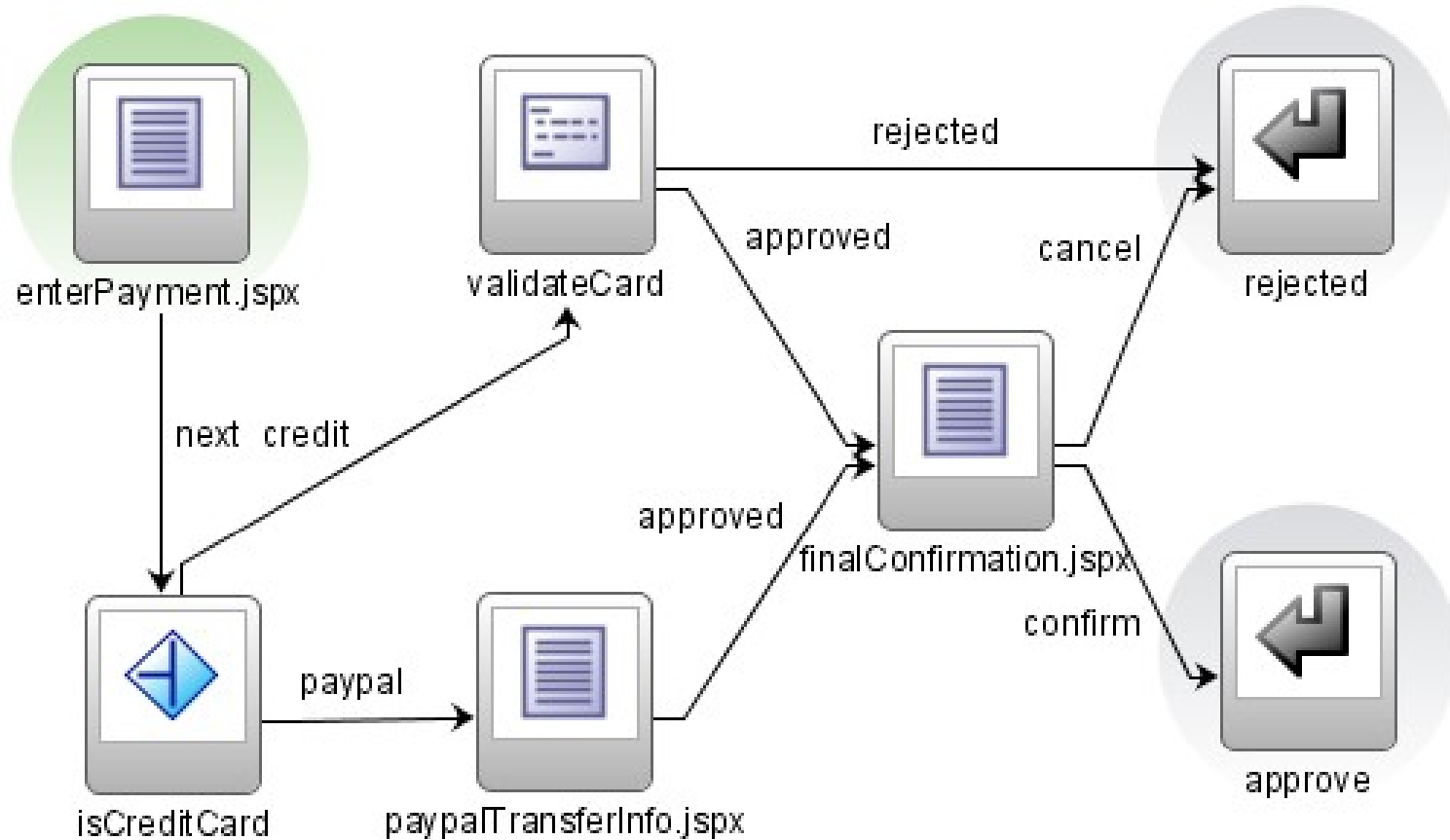
# Add Options to JavaServer Faces Technology Flow Definition

- Idea: add a place to specify code execution points in the JavaServer Faces flow
- Examples:
  - Method Execution
  - Decision Factors

# Is This Clear Enough?



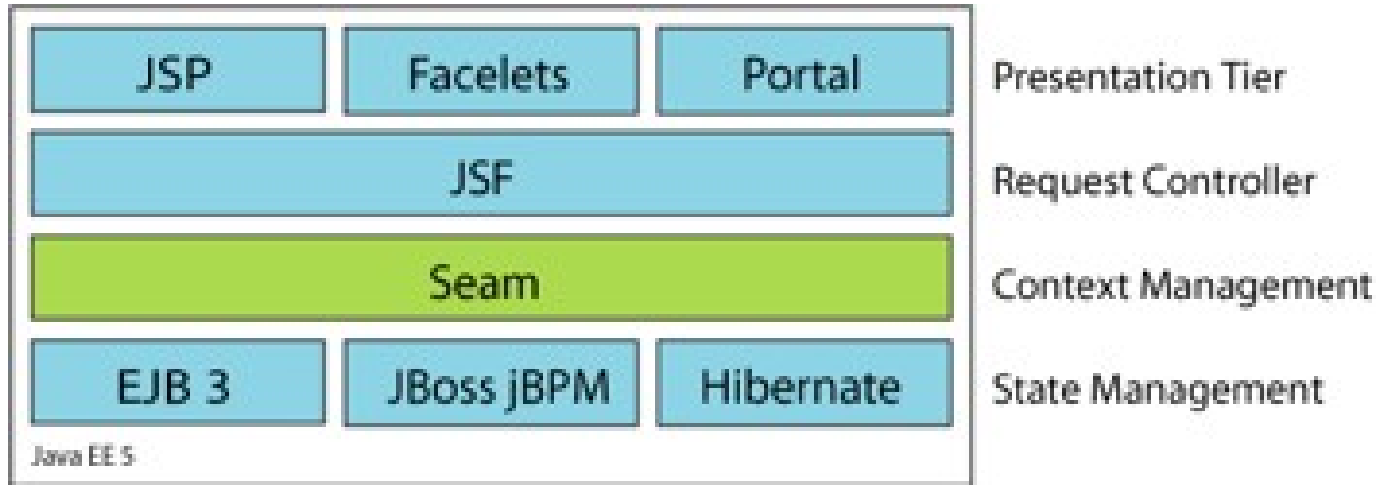
# Is This Clearer?



# Binding

- Problem : Binding UI to Business Components
- Solutions
  - JBoss Seam (WebBeans)
  - Oracle ADF (JSR-227)

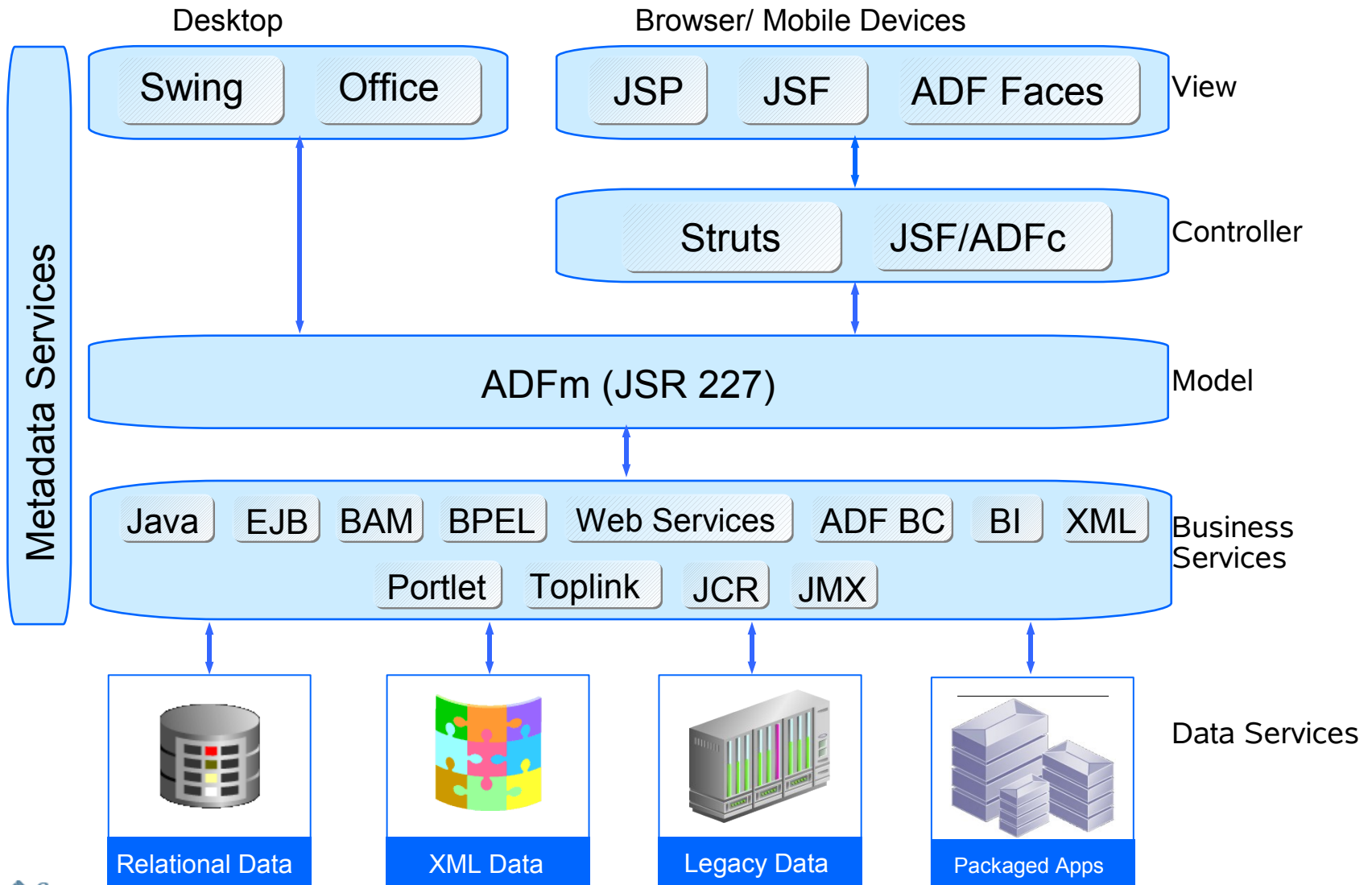
# SEAM



# Seam Binding

- Data in a JPA Bean
- Annotate it to bind to JavaServer Faces application
  - `@Name("user")`
  - `@Scope(SESSION)`
- Methods in a Session Bean
- Annotate them to bind to JavaServer Faces applications
  - `@Name("register")`
  - `@In`

# Oracle ADF





# ADF Binding

- XML Describes the data source
- XML Describes the page binding
- No code goes into your Enterprise JavaBeans™ (EJB™) architecture/POJO

# Demo

## Data Binding with Oracle ADF

# Reusability for JavaServer Faces Technology

## ➤ Task Flows

- Reusable JavaServer Faces flows

## ➤ Page Fragments

- Sub-Pages included in other pages

## ➤ Regions

- Area on a page that can include tasks or fragments

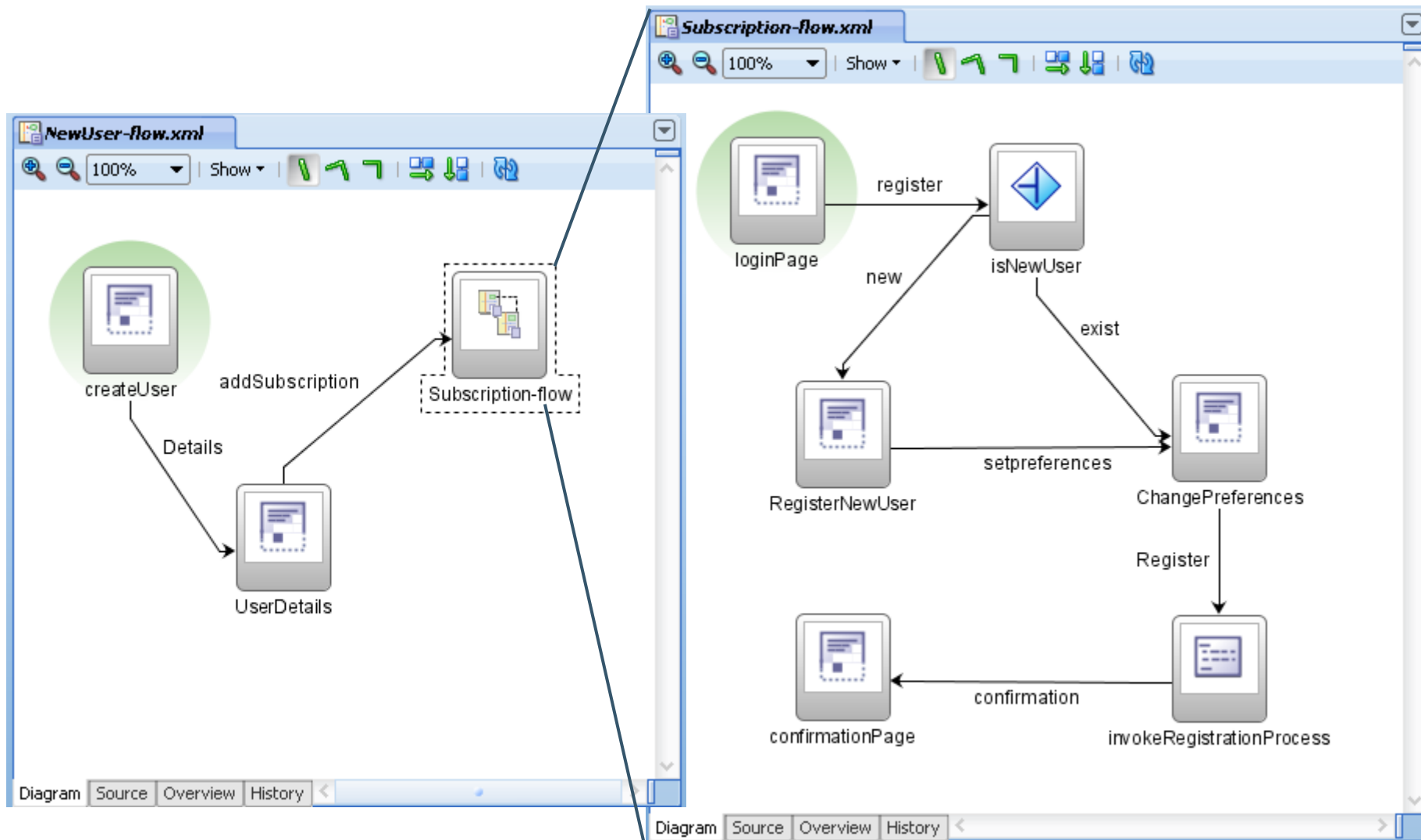
# Task Flows - Reusable Page Flows

- A Task flow represents a reusable block of functionality with a single entry point, defined exit points, and its own memory scope and transaction boundaries
- Task flows can receive parameters from, and return parameters to, a calling task flow

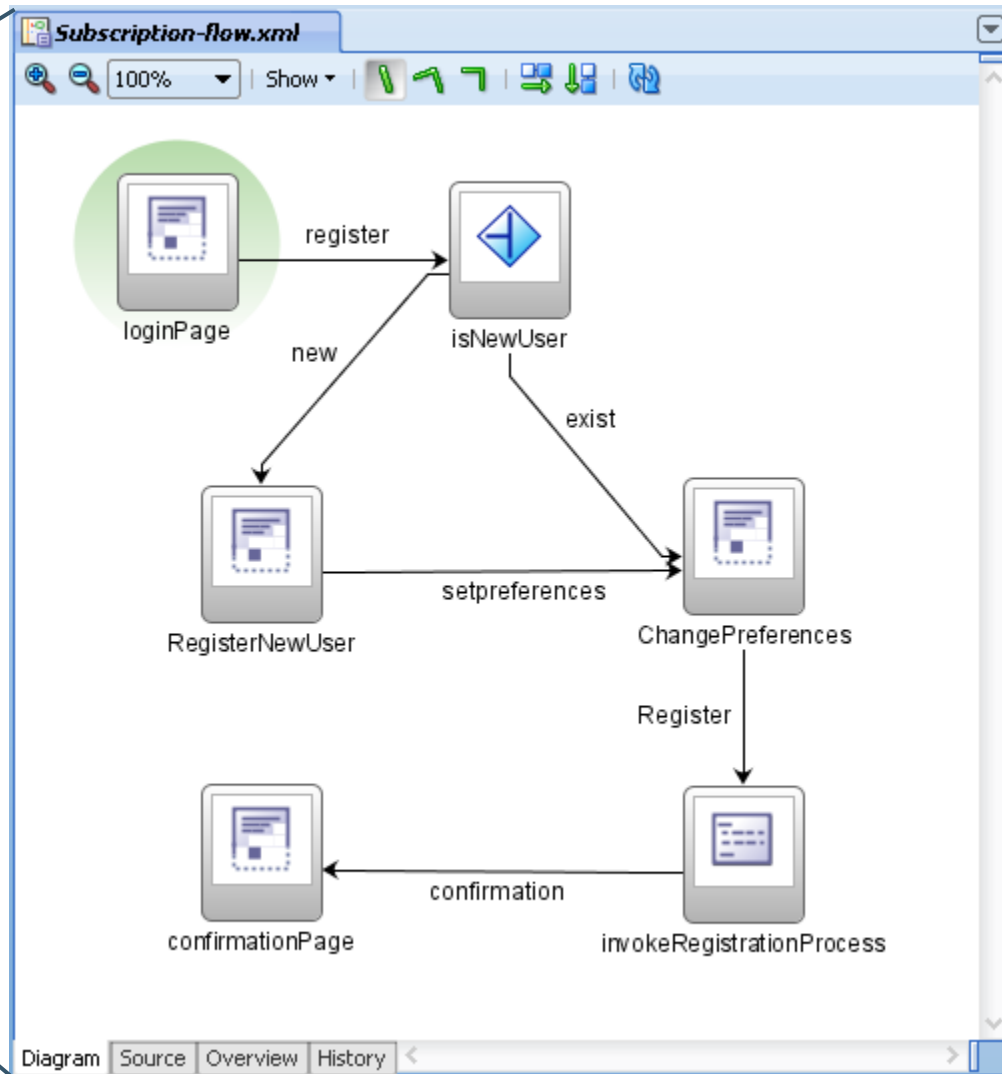
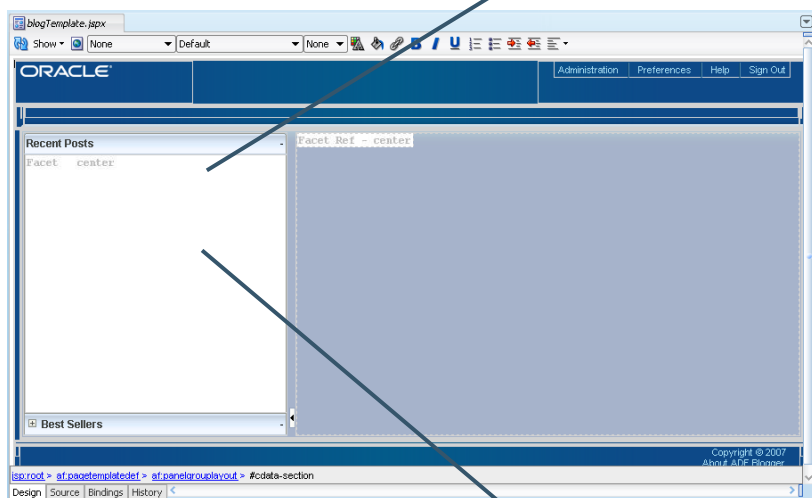
# ADF Task Flows Features

- Trains: a progression of related pages
- Save for later — resume incomplete transactions
- Declarative bookmarking
- Declarative transaction management
- Declarative support for browser back button navigation
- Declarative support for exception handlers
- Declarative security based on Java Authentication and Authorization Service (JAAS)

# Use It in Another Flow



# Use It Inside Another Page



# Demo

## Reusable JavaServer Faces Flows with JSF Page Fragments



# Conclusion

- JavaServer Faces technology is in a good state
- But to be successful...
- Consider using a JavaServer Faces technology-based framework

# For More Information

➤ <http://oracle.com/technology>

- Software
- Demos
- Tutorials
- Discussion Forum
- Blogs (mine is at [blogs.oracle.com/shay](http://blogs.oracle.com/shay))
- More...

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

# THANK YOU

Shay Shmeltzer, Group Manager, Oracle

**ORACLE®**  
**JDEVELOPER**

