JDeveloper 10*g*
Web Application
View Layer Alternatives

Peter Koletzke

Technical Director &
Principal Instructor

quovera

At Least We Have a Path

EVERY man spins a web
of light circles
And hangs this web in the sky
Or finds it hung,
already for him,
Written as a path
for him to travel.

— *Carl Sandburg (1878-1967), Webs*

Survey

- Java development
  - 1-2 years?
  - 3-9 years?
  - More than 10 years?
- JDeveloper
  - 1-2 years?
  - More than 2 years?
- JSP?
- UIX?
- JSF?

Agenda

- The View Layer and JDeveloper
- JSP Architecture and Development
- UIX Architecture and Development
- JSF Architecture and Development
- Conclusions

Rumor: There is a good book out on JDeveloper 10*g*.

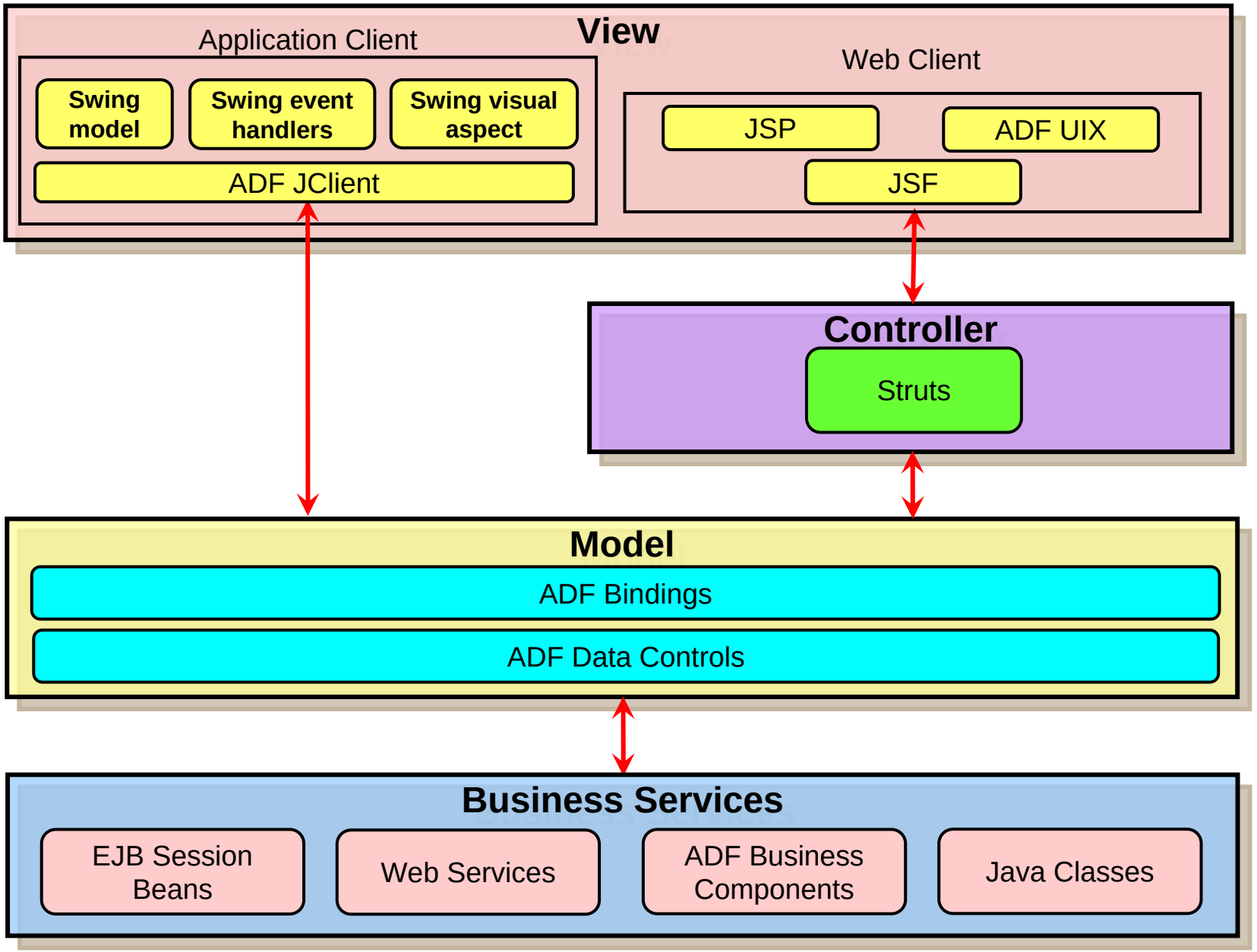
Warning: Material is from the early part of this month.

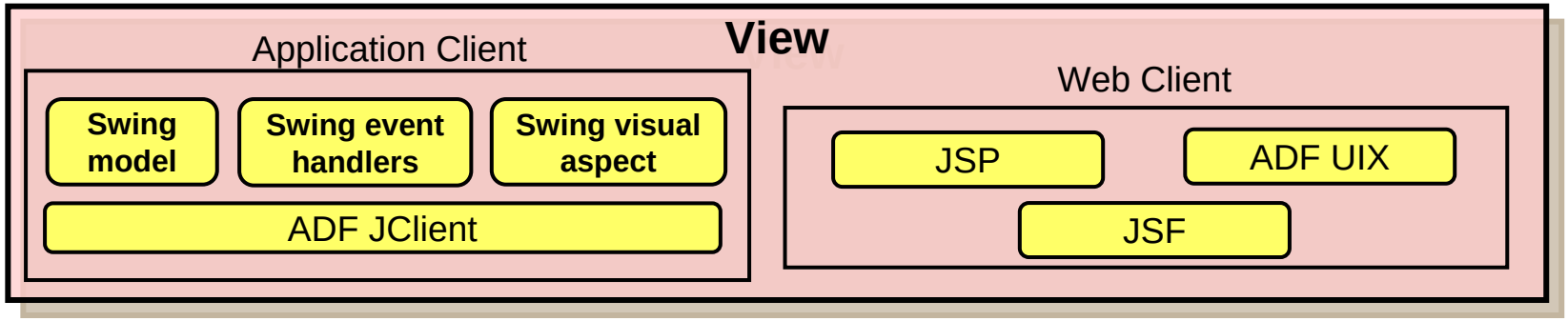
MVC and JDeveloper

- Model-View-Controller (MVC) – a J2EE design pattern
  - Separates UI (View), page flow (Controller), and data access (Model) code so layers can be swapped in and out
- Oracle Application Development Framework (ADF) architecture in JDeveloper built around MVC
- Different areas of JDeveloper support different layers
  - Model – Business Services modelers and editors
  - View – Visual editors, Property Inspector
  - Controller – Struts Page Flow Diagram

MVC	ADF	Description
Model	Model	Automatic data binding to a business service (data) source; data controls offer components; common to all business services
	Business Services	Code to access database sources; business logic; persistence; O.R. mapping
View	View	JSP and UIX fit here
Controller	Controller	Currently, the integrated technology is Struts.

ADF Architecture

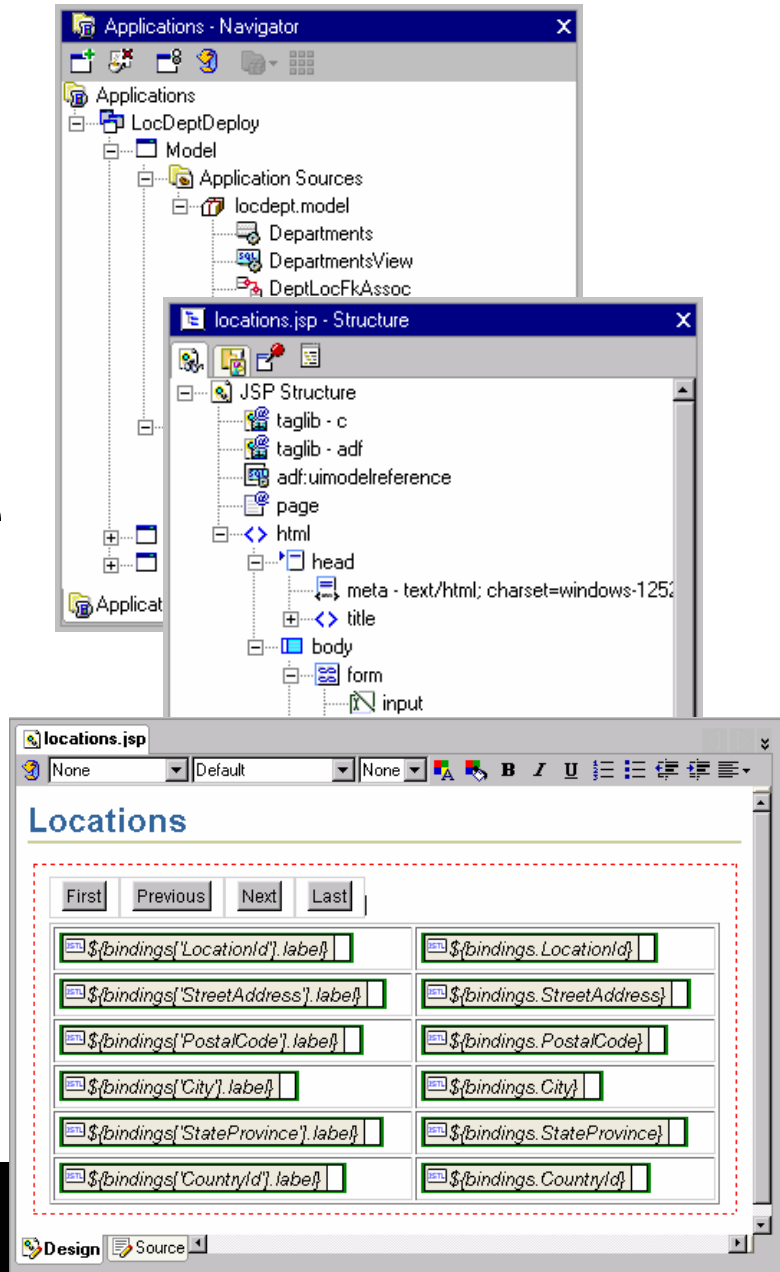




- User interface technologies
  - Application client
    - Java runtime on the client
    - Part of J2SE (standard edition)
    - Uses JClient framework to communicate with model layer
    - Swing contains its own MVC components
  - Web client
    - JavaServer Pages (JSP) technology
      - J2EE standard, light-client, tag-based interface
    - ADF UIX
      - Oracle-specific, XML-based interface used by E-Business Suite applications

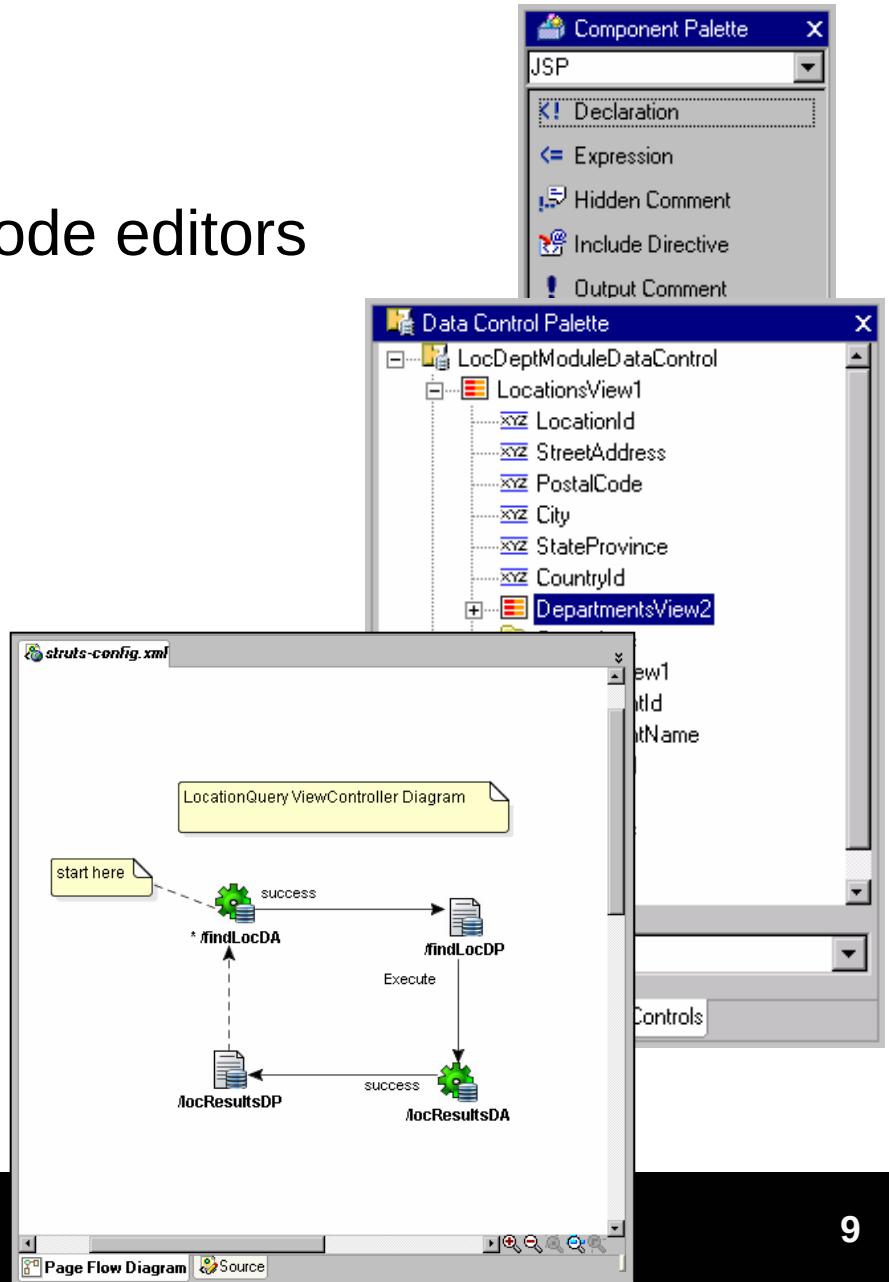
JDeveloper Work Areas

- Navigator
  - Workspaces and projects
  - Files
- Structure Window
  - Shows details of selected file
  - For UIX and JSP code, shows the object hierarchy
- Code Editor
  - Standard, full-featured editor
- Visual Editor
  - Modify layout



More JDeveloper Work Areas

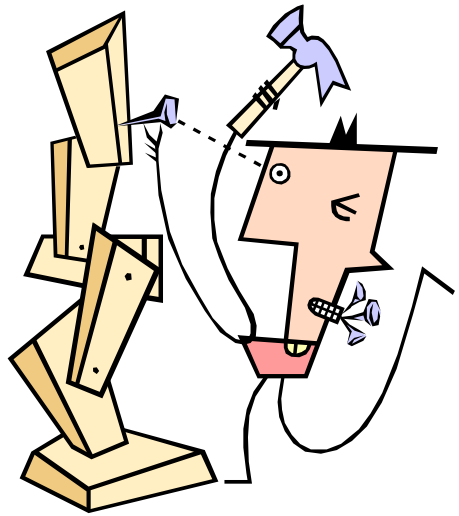
- Component Palette
  - Drag into the visual or code editors
- Data Control Palette
  - Drag into the editors
  - Automatic data binding
- Property Inspector
  - The usual
- Struts Page Flow Diagram
  - Define and manage Struts components



JDeveloper ADF Development Process

1. Create application workspace
2. Create Business Services and Model layers
3. Create View and Controller layers
4. Test and debug

- Use the same tools for development regardless of technology choices



Agenda

- The View Layer and JDeveloper
- JSP Architecture and Development
- UIX Architecture and Development
- JSF Architecture and Development
- Conclusions

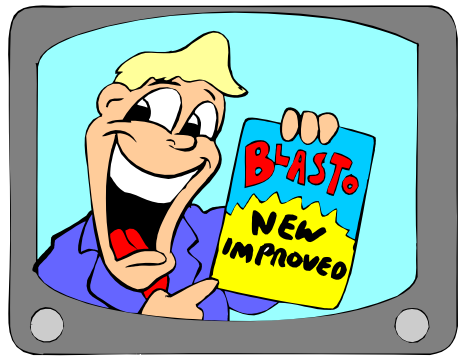
What is JSP?

It's an adjective

- JavaServer Pages technology
- Mature evolution of Java servlet
  - Pure Java code running on app server
- Defined by Java 2 Platform, Enterprise Edition (J2EE) specs
  - Lots of Java community interest and support
  - Lots of prebuilt code libraries
    - For example, one component draws an HTML table with data

JSP Features

- Runs in a Java Virtual Machine (container) process on a web application server
- Coded in Java (servlet) tags and HTML tags
  - JSP-specific tags
  - Tag library tags
- Client display is limited to HTML
  - Can extend functionality with JavaScript
  - Can extend look and feel with Cascading Style Sheets

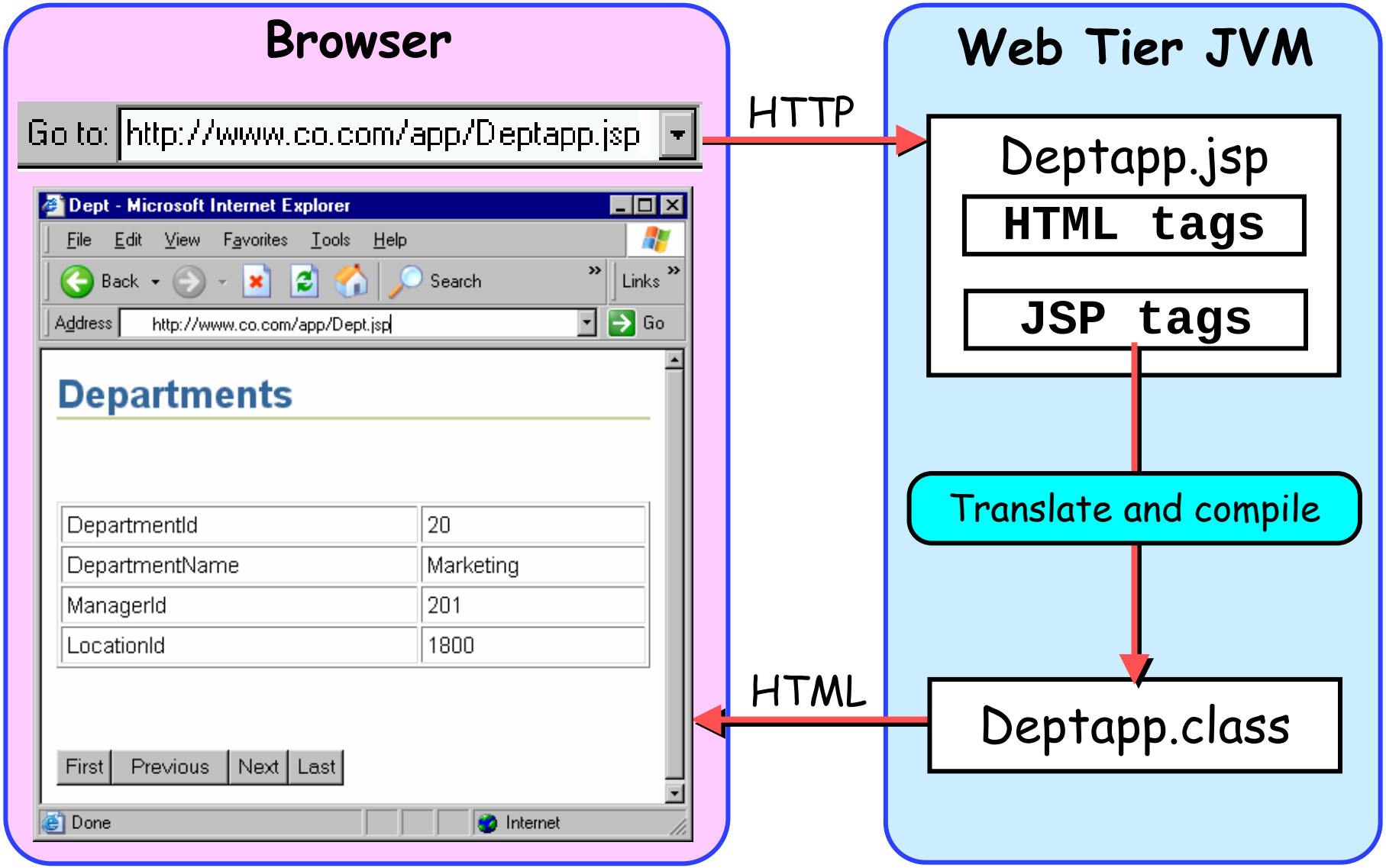


JSP Runtime Architecture

- Code runs in a JVM (Java Virtual Machine) on app server – called the Web Tier container
  - Use Struts or other Controller code

1. The client issues a URL request for a JSP file
2. The web server sends the request to the Web Tier (JVM) container
3. Container translates the file into Java, compiles the Java file (one time only
4. Container runs the file
5. The Java file creates HTML that is sent back to the browser

JSP Calling Sequence

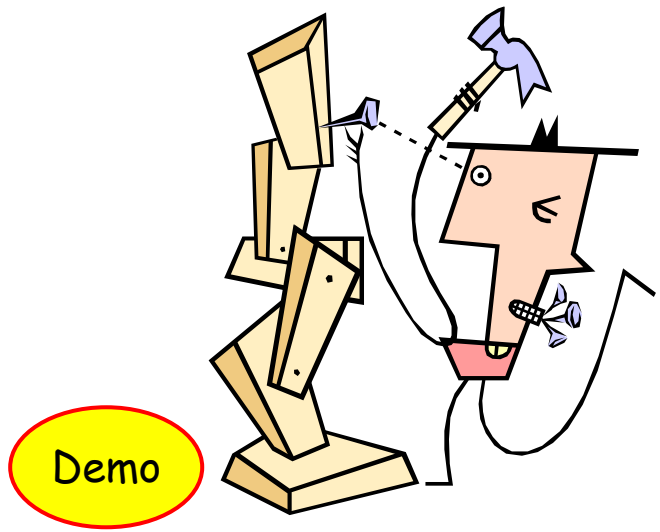


JSP Code Snippet

```
<table border="1" cellpadding="3" cellspacing="2" width="100%">
   <tr>
     <td>
       <c:out value="${bindings['DepartmentId'].label}"/>
     </td>
     <td>
       <c:out value="${bindings['DepartmentId']}"/> 
     </td>
   </tr>
   <tr>
     <td>
       <c:out value="${bindings['DepartmentName'].label}"/>
     </td>
     <td>
       <c:out value="${bindings['DepartmentName']}"/> 
     </td>
   </tr>
<!-- more table rows with fields -->
</table>
```

JSPs in JDeveloper 10*g*

- No wizard support
- Various code-generating tools shown earlier
  - Structure Window
  - Data Control Palette
  - Component Palette
  - Property Inspector
  - Struts Page Flow Diagram
  - Visual Editor
  - Code Editor

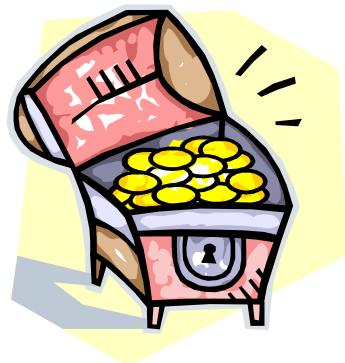


Agenda

- The View Layer and JDeveloper
- JSP Architecture and Development
- UIX Architecture and Development
- JSF Architecture and Development
- Conclusions

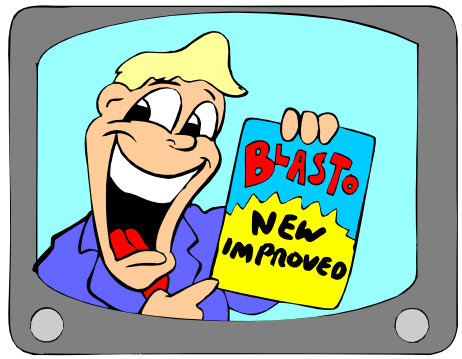
What is UIX?

- User Interface XML (9*i*); ADF UIX (10*g*)
  - Code file is written in XML
- Oracle framework for light client applications
  - Code libraries
  - Documented development method
  - Support in JDeveloper
- The main view technology for E-Business Suite applications
  - Developed and used by Oracle Apps developers for over 5 years
  - You can extend apps using UIX
  - You can also use it for any application



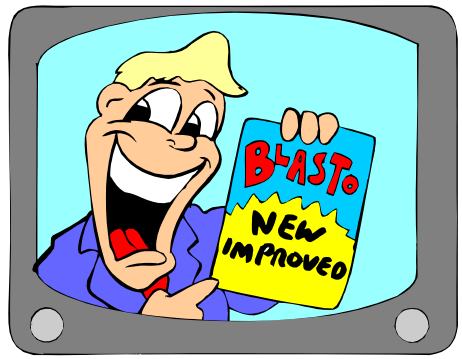
UIX Features

- J2EE compliance
  - Shares design principles with JavaServer Faces (a new addition to JSPs)
- Standardization
  - Templates are core design elements
  - Look-and-feel (fonts and colors) or “skins”
    - Can be changed with one config property
- Solid development support in JDeveloper 10*g*
  - Limited visual editing; Property Inspector
- Dynamic images
  - Tab and button images are generated at runtime
    - Text on image is base on UIX properties
    - No maintenance of image files



More UIX Features

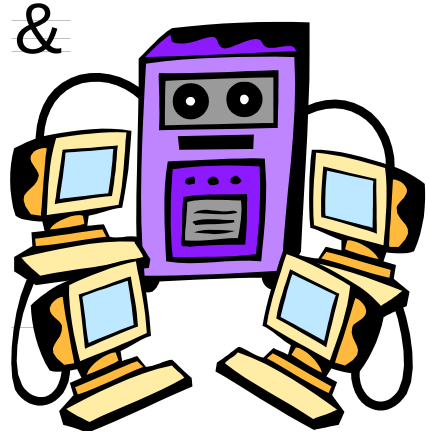
- Partial Page Rendering
  - Only part of page updates when you resubmit
  - Available on selected controls
- Message handling
  - Standard message area under tabs
  - Error for a field contains link to problem field
- Rich component set
  - Date field with calendar LOV button
  - Search component that contains OR capability
  - Tree, Master-Detail (various styles)
  - Shuttle control, containers



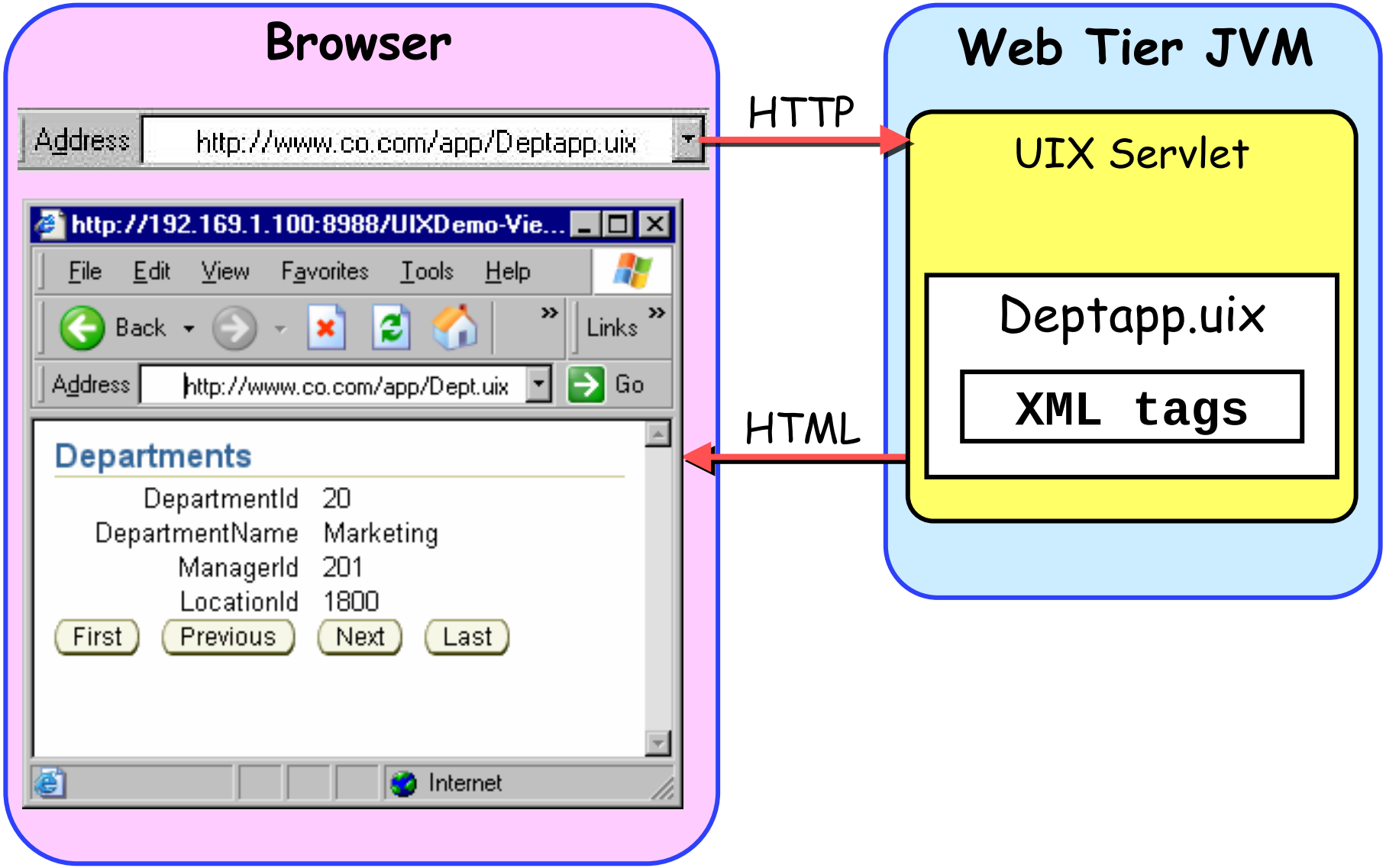
UIX Runtime Architecture

- Code runs on a web application server (like JSP code)
  - A special process (UIX servlet) runs in an application server JVM
  - Uses Struts or other controller

1. The client issues a URL request for a UIX file
2. The web server sends the request to the UIX servlet running in the Web Tier container
3. The UIX servlet interprets the XML tags, & assembles data base for the UI controls
4. Servlet constructs HTML and sends it to the browser



UIX Calling Sequence



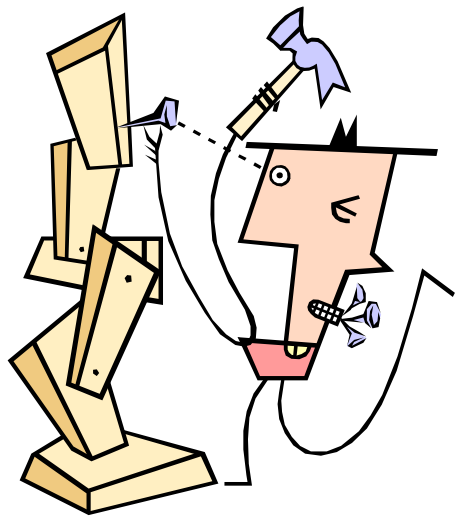
UIX Code Snippet

```
<labeledFieldLayout>
  <contents>
    <messageTextInput model="${bindings.DepartmentId}" columns="10"
       readOnly="true">
      <onSubmitValidater>
        <decimal/>
      </onSubmitValidater>
    </messageTextInput>
    <messageTextInput model="${bindings.DepartmentName}" columns="10"
         readOnly="true"/>
    <messageTextInput model="${bindings.ManagerId}" columns="10"
         readOnly="true">
      <onSubmitValidater>
        <decimal/>
      </onSubmitValidater>
    </messageTextInput>
    <messageTextInput model="${bindings.LocationId}" columns="10"
        readOnly="true">
      <onSubmitValidater>
        <decimal/>
      </onSubmitValidater>
    </messageTextInput>
  </contents>
</labeledFieldLayout>
```

DepartmentId	20
DepartmentName	Marketing
ManagerId	201
LocationId	1800

UIX in JDeveloper 10*g*

- Wizards
  - Start with template
  - Start without template
  - Roll your own template
- Same tools as JSP except
  - UIX Visual Editor
  - UIX Preview
  - XML Editor



Demo

Agenda

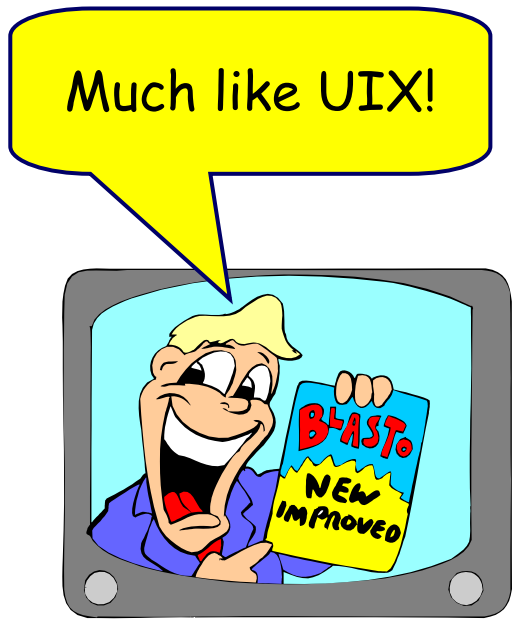
- The View Layer and JDeveloper
- JSP Architecture and Development
- UIX Architecture and Development
- JSF Architecture and Development
- Conclusions

What is JSF?

- JavaServer Faces
- “New” technology (ratified JCP in 5/2004)
  - Not part of J2EE yet
  - Offers reference implementation
- Effort to simplify JSP development
  - Component-ize it
    - High-level components provide much functionality
  - Integrate the controller
    - No Struts needed
  - Write less HTML
    - Component handles HTML writing
- Development friendlier to Forms developers
  - Declarative programming

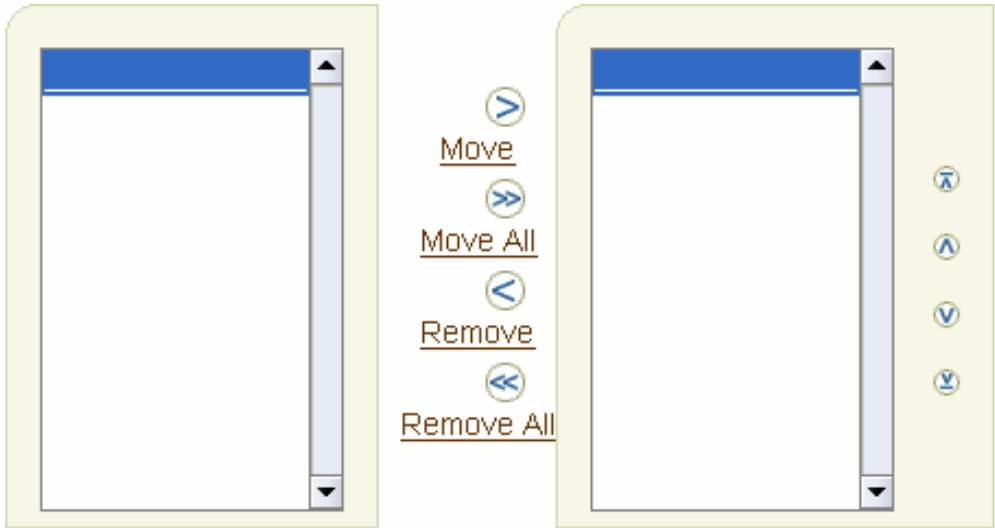
JSF Features

- Rich component set
  - Core library – for application tasks
  - HTML library – for HTML tags, forms
  - JSP tag library included
    - Can be implemented in other languages
  - Include data binding properties
- Event-driven
  - Events on the component level
  - Think Forms triggers
- Flexible output
  - HTML, WML, telnet (char mode)



ADF Faces

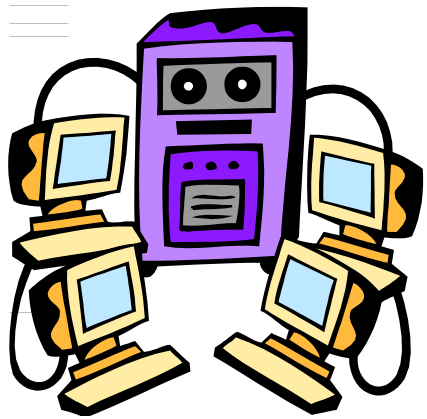
- Oracle tag library
  - Available on OTN as preview
  - Will be production with JDeveloper 10.1.3
- Implements components available in UIX
  - Uses JSF mechanisms instead of UIX mechanisms
  - Adds even more functionality to JSF
  - For example, selectOrderShuttle:



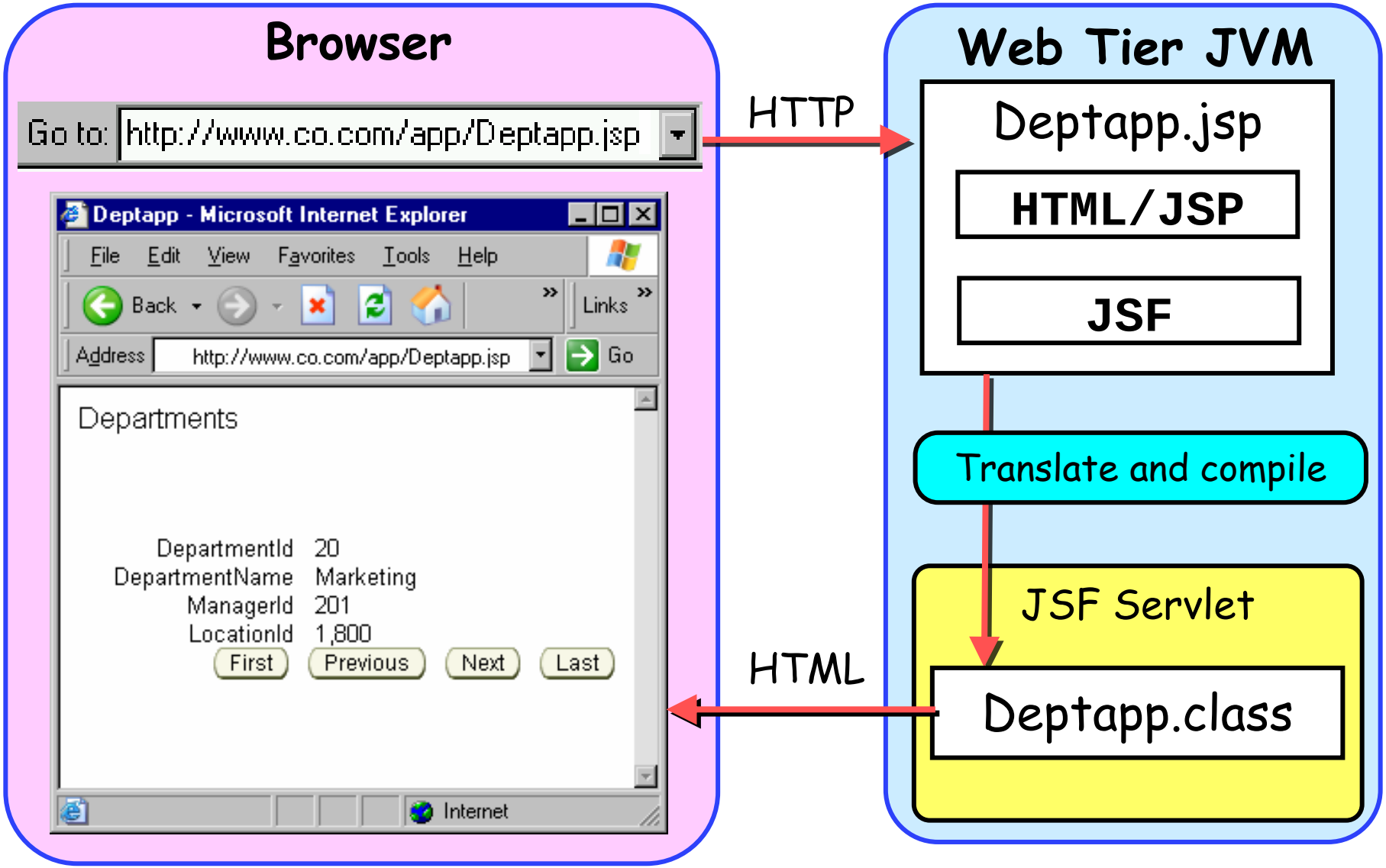
JSF Runtime Architecture

- Code runs on a web application server
  - Like JSP and UIX code
  - A special process (FacesServlet) runs in an application server JVM – similar to UIX
  - Uses JSF Controller

1. The client issues a URL request for a JSP file
2. The web server sends the request to the Web Tier container
3. Container translates the file into Java, compiles the Java file (one time only); passes it to Faces servlet
4. Faces servlet runs the file and interprets the JSF code
5. The servlet constructs HTML and sends it to the browser



JSF Calling Sequence

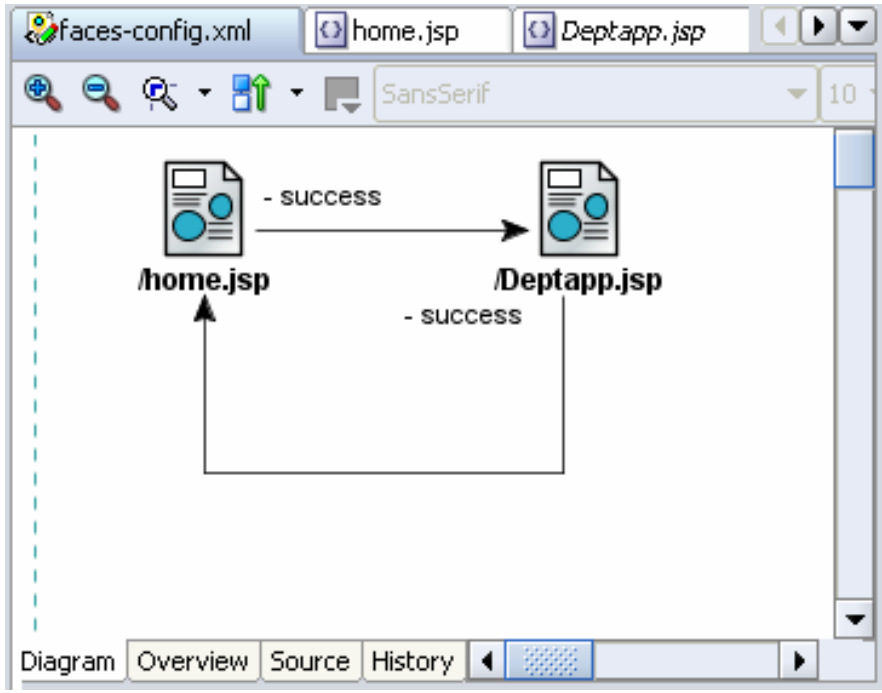


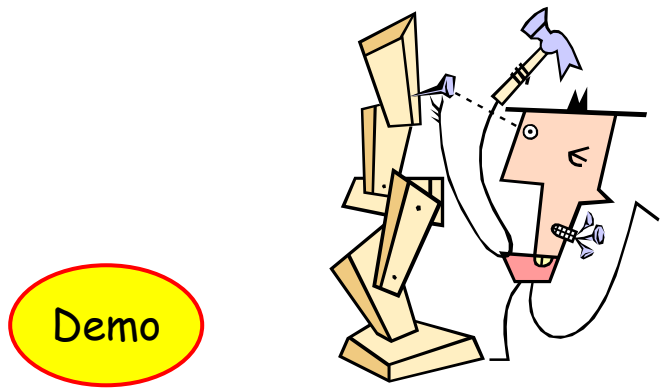
JSF Code Snippet

```
<f:facet name="footer">
  <af:panelButtonBar
      binding="#{backing_Deptapp.panelButtonBar1}">
    <af:commandButton
        actionListener="#{bindings.First.invoke}"
        action="First" text="First"
        disabled="#{!bindings.First.enabled}"
        binding="#{backing_Deptapp.commandButton1}"/>
    <af:commandButton
        actionListener="#{bindings.Previous.invoke}"
        action="Previous" text="Previous"
        disabled="#{!bindings.Previous.enabled}"
        binding="#{backing_Deptapp.commandButton2}"/>
  </af:panelButtonBar>
</f:facet>
```

JSF in JDeveloper 10*g* (10.1.3)

- Similar to JSP and UIX
- Instead of Struts Page Flow Diagram, use JSF Navigation Diagram
- Different Component Palettes, of course





Demo

Agenda

- The View Layer and JDeveloper
- JSP Architecture and Development
- UIX Architecture and Development
- JSF Architecture and Development
- Conclusions

When to Use JSP Technology

- You have existing applications using JSP and do not want to learn a new technology
- You have pre-existing JSP templates or a JSP look and feel
  - Or, you do not mind developing a look and feel
- You need industry-wide support and assistance from a large user community
- You have sufficient in-house Java expertise
  - JSP pages require more Java coding to customize controls and behavior
- You need to be 100% J2EE now

When to Use UIX Technology

- You need to extend Oracle Apps
  - Use JDev 9*i*'s Oracle Application Framework (OAF) to assist – 11.5.10
- You have a shop of "traditional" Oracle developers
  - UIX has slicker controls that require less Java coding
- You want a pre-built look and feel
- You can live with Oracle-centric support and user communities
- You do not need to be entirely J2EE
  - 10.1.3 will be 100% J2EE – extensions to JSF

When to Use JSF Technology

- You want to work with components
  - Less code, event-driven
- You have a Java shop
  - Use standard JSF components
  - Plug into JDeveloper 10.1.2
  - When ADF Faces releases, add that to your toolbox
- You are not a Java shop
  - Taste it with OTN preview
  - Wait until JDev 10.1.3 production
  - The normal caution about new technologies
    - Develop a small “unimportant” app first

Summary

- JSP pages are coded in HTML and JSP tags
  - J2EE standard, not Oracle-specific
  - Extensive support in JDeveloper 10.1.2
    - Tag libraries and easy data binding
- UIX is XML code
  - Oracle-specific, E-Business Suite
  - Rich component set
- JSF is an add-on to JSP pages
  - Main tool will be JDev 10.1.3
  - JSF Faces (Oracle) will offer UIX capabilities to JSF code
  - Preview on OTN

Also co-authored with Avrom Roy-Faderman

- Please fill out the evals
- Books co-authored with Dr. Paul Dorsey
- Personal web site:
  `http://ourworld.compuserve.com/homepages/Peter_Koletzke`

quovera

http://www.quovera.com

- Founded in 1995 as Millennia Vision Corp.
- Profitable for 7+ years without outside funding
- Consultants each have 10+ years industry experience
- Strong High-Tech industry background
- 200+ clients/300+ projects
- JDeveloper Partner
- More technical white papers and presentations on the web site