... and UIX Begat ADF Faces: How Rich is ADF Faces Rich Client?

Peter Koletzke
Technical Director & Principal Instructor

From The Golden Age of Hollywood

We didn’t need dialogue. We had faces.

—Norma Desmond,
Sunset Boulevard (1950)

Survey

• Job responsibilities?
  – DBA, developer
• Frameworks?
  – MyFaces, Trinidad
  – ADF Faces, ADF Faces RC
  – ADF BC
  – Other
• Tools?
  – Eclipse
  – JDeveloper
  – Other

Agenda

• Who begat what?
• How rich?
• Atomic components
• Layout components
• Data visualization components
• Resources

The white paper contains some short hands-on practices.

See also:
2:30 (this room): Visualize Your Data, by Shay Shmeltzer
That’s Rich! Putting a smile on ADF Faces, by Lucas Jellema, ODTUG 2009

Slides and white paper will be on the conference website.
In the Beginning Was EBS

• E-Business Suite (Oracle Applications) needed a lightweight client interface
  – Oracle created UIX for this
• User Interface XML (Extensible Markup Language)
  – First supported in JDeveloper 9i
    • Became “ADF UIX” in JDeveloper 10g (10.1.2)
    • UIX support dropped out of JDev 10.1.3
  – Set of components
    • Tag language (e.g., af:inputText)
    • Used to build UIs (e.g., for web applications)
  – Concurrent with JSP in the industry
  – Still used in EBS through R12
    • 9i version of UIX, not the 10g version

And UIX Begat ADF Faces

• Application Development Framework (ADF) came upon the land
  – Formalized first in JDeveloper 10g
  – Introduced specific ADF frameworks
    • ADF Business Components (formerly BC4J)
• JSF had become a Java standard
  – Not in the EE editions, but an industry trend
• Oracle converted UIX components to the JSF standards
  – And called it “ADF Faces”
  – And it was good.

And ADF Faces Begat ADF Faces RC

• Oracle released ADF Faces (non-RC) to open source Apache Project
  – Called “Trinidad” (part of MyFaces)
  – myfaces.apache.org/trinidad
• Oracle created ADF Faces Rich Client
  – Introduced in JDeveloper 11g
  – Concurrent with introduction of ADF Controller (task flows)
• Oracle used it to build Fusion Applications
  – A very heavy investment
• And it was very good.

Agenda

• Who begat what?
  • How rich?
    • Atomic components
    • Layout components
    • Data visualization components
    • Resources
Rich == Highly-Interactive

- Components built for interactivity
  - For example, table grid component
    - Resize columns with drag and drop
    - Reorder columns with drag and drop
    - No page refresh wait when scrolling through rows
- Support for pop-ups and dialogs
- Client-side validation
  - Property driven
  - For example, formats, ranges, and required
  - Error messages appear next to the component with a problem value

Rich == Flexible

- Fully declarative AJAX support
- Built on top of JSF APIs
  - Deployable on any 1.2 implementation of JSF
- Configurable skins
  - CSS development work
  - A single property applies the skin to all components in the application
- Large range of UI items
  - 150+ components (and counting)
- Internationalization and accessibility

Rich == Easy To Use

- Property-driven components that save development effort
- Layout components to arrange other components
- Data Visualization Tools (DVT) components
  - Lots of functionality with little effort
  - Chart, Gantt, Pivot Table, Map, Hierarchy Viewer, Gauge

Deep Dive Into af:table

- Column grouping
- Row highlighting
- Column moving
- Export to Excel
- Detach option
- Column sorting & filtering
Agenda

- Who begat what?
- How rich?
- Atomic components
- Layout components
- Data visualization components
- Resources

Atomic Components

Most are on the Common Components page of the Component Palette

Some Atomic Components

- af:inputText
- af:inputDate
- af:commandButton
- af:menuItem
- af:commandImageLink
- af:selectOneChoice
- af:selectBooleanCheckbox
- af:inputListOfValues
- af:selectManyShuttle

af:selectManyShuttle

- Single component for multiple-selection list
  - Checkmark selection or
  - Double click or
  - Select and click arrow buttons

af prefix identifies the ADF Faces RC library
Some Rich Features

- Atomic components are the main user interface items
- Features:
  - Converters; e.g., number format
  - Validators; some are built in, e.g., required
  - Messages appear next to items
  - AJAX
  - Drag and drop

Using AJAX in ADF Faces RC

- Asynchronous JavaScript and XML
- Partial Page Rendering (PPR) in ADF Faces
  - “Declarative AJAX”
- Much AJAX in ADF Faces is transparent
  - Built into the components
  - Nothing special needs to be done
- You can setup non-default AJAX behavior using properties
  - partialSubmit – used by command items
  - autoSubmit – used by input items/lists, etc.
  - partialTriggers – all components, sets up the “viewer” (listener)

AJAX Interactions – Total Pay

### Total Pay

- Salary 400
- Raise 100
- Total Pay 500

#### Salary

<table>
<thead>
<tr>
<th>Id</th>
<th>Value</th>
<th>AutoSubmit</th>
</tr>
</thead>
<tbody>
<tr>
<td>salary</td>
<td>#{bindings.Salary.inputValue}</td>
<td>true</td>
</tr>
</tbody>
</table>

#### Raise

<table>
<thead>
<tr>
<th>Id</th>
<th>Value</th>
<th>AutoSubmit</th>
</tr>
</thead>
<tbody>
<tr>
<td>raise</td>
<td>#{bindings.raise.inputValue}</td>
<td>true</td>
</tr>
</tbody>
</table>

#### Partial Triggers

- Salary raise
- Recalculate

Drag and Drop

- The value from Drag Source will be copied into Drag Target
Modifying the Visual Aspects

• Skins
  – First and foremost – get this right

• Properties - secondary
  – ContentStyle
    • For data inside the component (foreground)
  – InlineStyle
    • Set from tab area below it
    • Or just type it in

Another Visual Aspects Property

• StyleClass
  – Equivalent to the HTML class property
  – Apply existing style sheet selectors
  – Can apply more than one to a component

Agenda

• Who begat what?
• How rich?
• Atomic components
• Layout components
• Data visualization components
• Resources

Container Components

• Most are on the Layout page of the Component Palette
  – Use them to hold other components
  • af:panelHeaderLayout
    – Region title
    – Stack components under it
  • af:panelGroupLayout
    – Lay out components in a row or in a column
    • horizontal
    • vertical
Some Rich Features

• Allow for sophisticated layouts
  – Nest layout components within layout components
  – Virtually limitless possibilities

• Facets (sub-components) provide special functionality
  – For example, relative placement of components – regardless of resolution or window size
  – `<f:facet>` component

• Use af:spacer to fine tune placement

• Hide or display the contents using the layout component properties

More Layout Components

• af:panelBorderLayout
  – Predefined layout areas
  – Uses facets to hold the contents of each area
    • Start, end, top, bottom (and more)

• Center area stretches its contents to fill the area

Super-Rich Layout Components

• af:panelSplitter
  – Split pane control
  – Optional: user can move the drag bar
    • Horizontal or
    • Vertical

• af:calendar
  – MS Outlook style

• af:carousel
  – Good for visual browsing

• af:panelAccordian
  – Also found in MS Outlook

Layout Tip

• Design This Container option in the pulldown menu for the layout component

  • Allows you to resize container elements visually
    – This action changes the property values

• Another tip: Assign color to the borders so you can see the containers
PanelFormLayout

- Lay out fields in rows and columns
  - Perfect for most input forms
- Right justifies prompts
- Left justifies fields
- Tab order is down the first column, then across to the second column
  - Not necessarily intuitive; workaround:
    - Multiple PFLs in a single vertical PGL
    - Set `fieldWidth` and `labelWidth` of the PFLs

Details in the white paper

Achieving the Perfect Layout

- Know your container components
  - Facets are powerful things
- More
  - `af:popup`
    - Drop in an `af:dialog` or `af:window`
    - Drop `af:showPopupBehavior` into an action item (button or menu choice)
  - `af:panelStretchLayout`
    - Expands contained components to fill width
    - Use if another container cuts a component
  - `af:menuBar`
    - Panel Menu Bar
    - Creates menu area, drop in `af:menu` then `af:menuItem`
  - `af:panelLabelAndMessage`
    - Provides a prompt for a group of objects
    - FirstName and LastName fields with a prompt of “Name”
    - Use inside `af:panelFormLayout`

More Perfect Layout Tips

- Design page fragments, not separate pages
- Use Quick Start Layouts when creating the page or template

Create Templates

- New gallery item for JSF page template
- Define facets
  - Your own layout areas
- Add attributes
  - Can transfer data from page to template
- Add container components
Using the Template

- Application’s templates appear in the JSF Page dialog
  - Use Page Template pulldown
- The layout elements are referenced from the template

Tip: Drop Into the Structure Window

- Drop on top of container into which you want the component to appear
- Much more accurate
- Other options
  - Click the component after selecting the Structure window node
  - Use the bread crumbs in the editor

Agenda

- Who begat what?
- How rich?
- Atomic components
- Layout components
- Data visualization components
- Resources

DVT Components

- Data Visualization Tools are on the ADF Data Visualizations page
- Each type has its own panel with components
  - Gantt
  - Gauge
  - Map
  - Graph
  - Hierarchy Viewer
  - Pivot Table
Some DVT Components

- Rich by nature
- Connect to data
  - Drop from the Data Control panel to the page

More Rich Features

- Specify data usage
- Control visual aspects using properties

Other Rich Features

- Drill down capability can be mostly declarative
- Some elements of the DVT components are mouse-aware
  - For example, mouse over in the bar graph
  - Displays details about the data point

Details in the white paper
Agenda

• Who begat what?
• How rich?
• Atomic components
• Layout components
• Data visualization components

• Resources

JDeveloper Help Center

• Help | Table of Contents
• Search engine
• Link to JDev forum
• Favorites tab
• Dynamic tab – Context-sensitive list based on the task at hand
• Opening a topic opens another tab

Visual Component Guide

• JDev Help Center (help system)
  – Search for “enhanced tag doc”; also one for DVT

• Other Resources

  • www.oracle.com/technology/products/adf/adffaces
    – The Web UI Developer’s Guide for Oracle ADF
    – Cheat Sheets
  • ADF EMG – enterprise methodology group
  • AMIS blog – ADF Faces white papers
  • rea.oracle.com
    – Explore Now | Sandbox | Rich Client
  • Quick Start Guide
    – Grant Ronald
  • Oracle Fusion Developer Guide
    – Nimphius and Munsinger
  • Oracle JDeveloper 11g Handbook
    – Mills and Yours Truly
Economics 101

The rich are different from you and me. They have more money.
—F. Scott Fitzgerald (1896–1940), author, and Mary McGuire Colum (1884–1957), critic

Summary

• UIX begat ADF Faces…
  – ADF Faces RC has a long heritage
• ADF Faces RC is pretty rich
  – High-interactivity, AJAX, skins, popups, dialogs, DVT, drag-and-drop
• Atomic components are mostly for interaction with users
• Layout Components provide sophisticated arrangement of components
• DVTs supply the functionality users often ask for
• There are many resources for help with learning and using ADF Faces
• And it is good.

Books co-authored with Dr. Paul Dorsey, Avrom Roy-Faderman, & Duncan Mills

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