Agenda

- Building the User Interface
  - UI Development
  - Page Flow
- A Focus on Faces
  - Introducing Java Server Faces
  - What is ADF Faces?
  - Building a JSF User Interface
- Building a User Interface: Comparing Forms and ADF

UI Development

Rich Clients          Web and Wireless
                       Controller
                       Model
                       Business Services
UI Development

- Visual editors
  - JavaServer Faces
  - JSP and HTML
  - Swing (ADF Swing)
- Component palette
- Common binding
- Property inspector
- Structure window

Design and Code Sync

- Design View or Code View
  - Available for all components
  - Full synchronization with the code
ADF Architecture Overview

- Rich Clients
- Web and Wireless
  - Controller
  - Model
  - Business Services

Developing the Page Flow

- Controller handles Page Flow
  - Promotes page reuse through abstraction
  - Increase flexibility and manageability
- Visual development with the Page Flow modeler
  - Draw the page flow
  - Drill down to code & pages
A Focus on Faces

What is JSF?

- Nickname: “Faces”
- Developed through Java Community Process (JCP) as JSR-127
- Started: May 2001
- First release: March 2004
- Contributes include: Sun, Oracle, IBM
- Should be part of JEE 5.0
Advantage of Using JSF

- Component centric approach
  - Build from libraries of components
    - ADF Faces, MyFaces
  - Huge range of components
    - E.g. outputLabel, dataTable
- Supports multiple clients
  - Browser, mobile/smart phone
- Empowers diverse types of developers
- Designed to be leveraged by tools

Advantage of Using JSF

- Abstracts away from HTTP
  - Developers don’t need to understand request-response cycle
  - Developers don’t need to decode requests
- Abstracts away from HTML
  - Developers work with components
  - Layout handled by components
Evolution of J2EE web tier

- Each is built on the previous layer
- Progressive levels of abstraction
- Higher level programming

The Forms Architecture
What is ADF Faces?

- Evolution of ADF UIX
- Built on top of JSF APIs
- Deployable on any compliant implementation of JSF
- Provides much more that JSF reference implementation
  - Much larger component set
  - Far more advanced and interesting components
  - Client side behaviour
  - ADF model support out-of-the-box
Choice of UI

Building a User Interface: Comparing Forms and ADF
Building a JSF User Interface

- Visual and Declarative editing
- Drag and drop
- Wizards and dialogs for data binding
- Page flow diagram
- Advanced structure editing with visual synchronization

Visual & Declarative Editing in Forms
Visual & Declarative Editing with ADF

Drag and Drop in Forms
Drag and Drop With ADF Faces

Wizards and Dialogs In Forms
Wizards and Dialogs in ADF

Controlling Page Flow in Forms

If process_submit then
Call_Form ();
Else
Open Form ();
End if;
Page Flow with ADF Faces

Editing Structures & Visual Synchronization in Forms
Editing Structures & Visual Synchronization with ADF

Global Look and Feel

- Similar to Visual Attributes
- Pre-Defined CSS
- Centralized file
- Easy to customize
- Template and Skin mechanism
Summary

- Creating a User Interface has many familiar features to Forms developers
- Using ADF Faces brings additional functionality, bring more power to visual web application development
Workshop Summary

- J2EE development is on ever increasing productivity path
- ADF is the best gateway into J2EE development for Forms and Designer developers
- Learn More:
  - Oracle Developer Days Workshops
  - OTN: http://www.oracle.com/technology/formsdesignerj2ee
Learn Oracle From Oracle

- Instructor led training
- Self-Study
- Online learning
- Oracle Certification
- Oracle iLearning
- Oracle Tutor

oracle.com/education

QUESTIONS & ANSWERS
Oracle
Developer Day
Sponsored by: intel
Oracle Developer Day ORACLE