

Introduction to ADF Core Technologies

Peter Koletzke

Technical Director & Principal Instructor



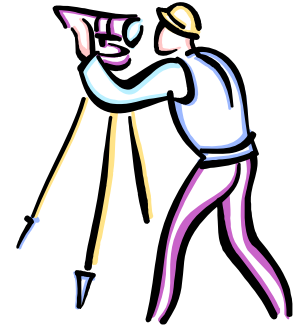
ORACLE
ACE Director



quovera

Survey

- Job responsibilities?
 - DBA, developer
- Languages?
 - PL/SQL, Java
 - Other
- Tools?
 - APEX
 - Developer Forms/Reports
 - JDeveloper
 - Eclipse, NetBeans
 - Other



quovera

2

Agenda

- What is ADF?
- ADF Business Services
- ADF Model, View, and Controller



Slides and white paper are available on the ODTUG and Quovera websites.

Screenshots are 11g unless marked as 12c



quovera

3

On the Positive Side...

If we do not find anything pleasant, at least we shall find something new.

Si nous ne trouvons pas des choses agréables, nous trouverons du moins des choses nouvelles.

—Voltaire (1694-1778), *Candide*

quovera

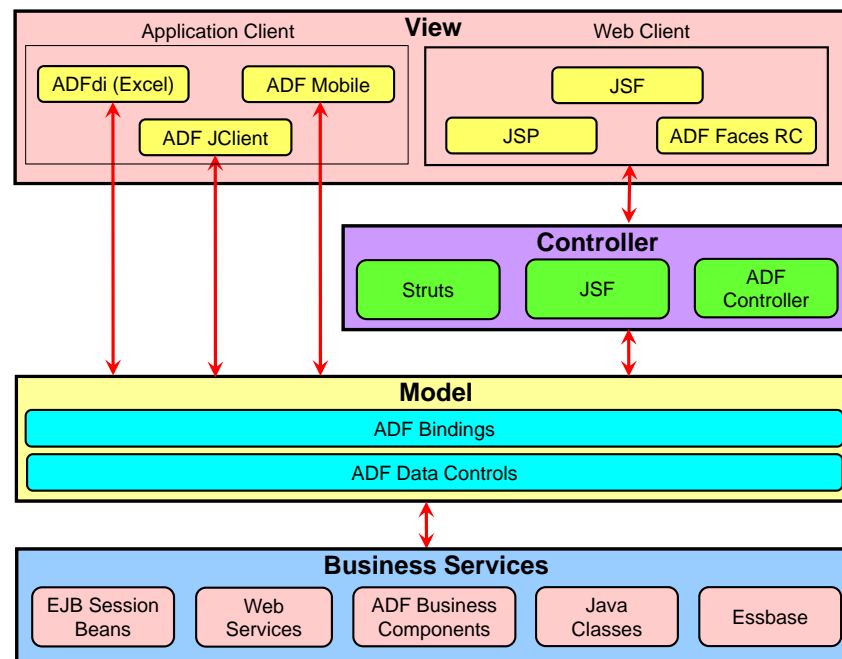
4

Oracle Application Development Framework (ADF)

- A *framework* is a prebuilt service for solving a particular problem – like access to the database
 - Code libraries and standards support the framework
 - Implements code reuse and best practices
 - An architecture with code libraries
- ADF is a *meta-framework*
 - A wrapper for other frameworks, JDev 10g and later
 - Provides a consistent developer experience
- Based on Model-View-Controller design pattern (SmallTalk, Java EE)



ADF Architecture



Which ADF Technologies to Use?

- An example: Cloud Applications (Oracle applications suite; f.k.a. Fusion Applications)
- *Core technology stack* used for Cloud Apps is:
 - ADF Business Components
 - ADF Faces Rich Client
 - ADF Model
 - ADF Controller
- Other *high-level technologies* or strategies also used
 - SOA, ESB, Business Rules, WebCenter, BPM, BPA, BAM
 - Mobile-enablement

Out of scope



ADF Essentials

- No-license-fee version of ADF
 - Runs on the public domain app server, Glassfish, not WebLogic Server
- Works in JDeveloper
- Works in Eclipse
 - Through Oracle Enterprise Pack for Eclipse
- More information on OTN
 - www.oracle.com/technetwork/developer-tools/adf/overview/adfessentialsfaq-1837249.pdf

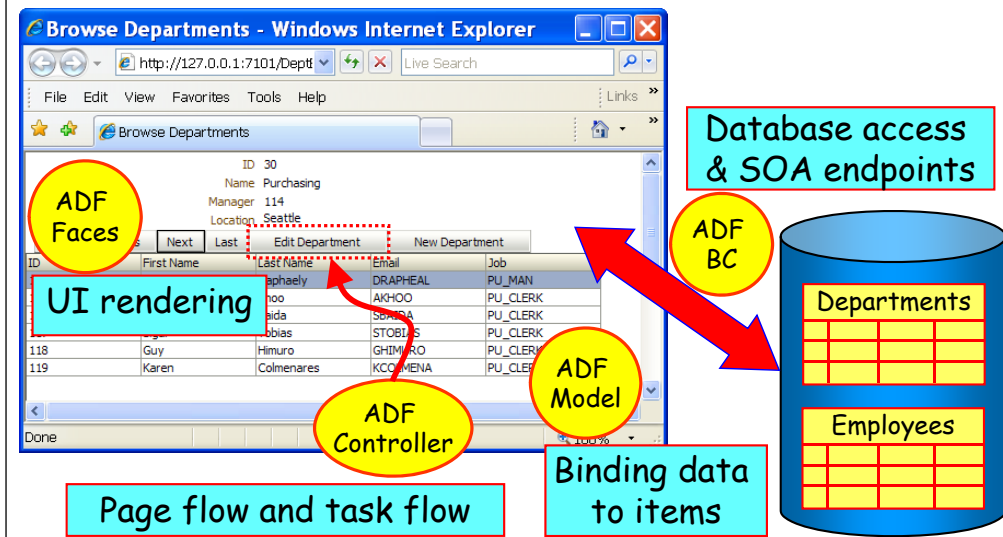


Mobile Application Framework

- Similar ADF principles and development methods
 - Business services
 - Pages (features) with components
 - Task flows
- Supports iOS and Android
- Use local services like camera & contacts
- OTN demo video and ADF Academy video
www.oracle.com/technetwork/developer-tools/maf



Where Do The Core Technologies End Up?



Which Languages Do You Use?

1. Java
 - All important programmatic code
 - Think “trigger code” as in Oracle Forms
2. XML
 - The components rely on XML
 - Property editors create it for you
3. JavaScript and Cascading Style Sheets
 - Add functionality to HTML pages
 - Usually the components do this work for you
4. Expression Language
 - Used in JSF binding properties
5. Groovy
 - ADF BC scripting



Agenda

- What is ADF?
- ADF Business Services
- ADF Model, View, and Controller

Sample app tutorial at quovera.com



The World View

In this best of all possible worlds ...
everything is for the best.

Dans ce meilleur des mondes possibles ...
tout est au mieux.

—Voltaire (1694-1778), *Candide*

ADF Business Components

- ADF BC: an option in the Business Services layer of ADF
- Persistence: storing data in a database
- O/R mapping: Translates relational database things to object-oriented (Java) whatsits
- Handles JDBC mechanics
 - Creates SQL and handles results
- Primarily declarative
 - XML source code to define the use of framework classes



More About ADF BC

- Various component types, e.g.,
 - View objects: define queries
 - Entity objects: define insert-update-delete (“DML”)
 - View links: view object relationships
 - Associations: entity object links
 - Application modules: Define the data models and the database transaction
- It does not create user interfaces



Sample ADF BC Development

A screenshot of the ADF BC development environment. The main window shows the 'Employees.xml' file with a table of attributes. The 'EmployeeId' attribute is selected. A 'Properties' dialog box is open for 'EmployeeId', showing fields for Name, Display Name, Type (Integer), Type Mapping, Property Set, and Date Effectivity. The 'Entity object editors' label is highlighted in a yellow box at the bottom of the screenshot.

Name	Type
CommissionPct	BigDecimal
CreatedBy	String
CreatedDate	Timestamp
DepartmentId	Integer
Email	String
EmployeeId	Integer
FirstName	String
HireDate	Timestamp
JobId	String
LastName	String



View Object Code

View Object = SELECT statement

```
<ViewObject
xmlns="http://xmlns.oracle.com/bc4j"
Name="AllEmployees"
Version="11.1.1.53.41"
SelectList="Employees.EMPLOYEE_ID,
Employees.FIRST_NAME,
Employees.LAST_NAME,
Employees.JOB_ID,
Employees.EMAIL,
Employees.HIRE_DATE,
Departments.DEPARTMENT_NAME,
Departments.DEPARTMENT_ID,
Departments.LOCATION_ID"
FromList="DEPARTMENTS Departments,
EMPLOYEES Employees"
Where="Departments.MANAGER_ID =
Employees.EMPLOYEE_ID"
BindingStyle="OracleName"
CustomQuery="false"
PageIterMode="Full"
UseGlueCode="false">
...
```

View Attribute = Column in query

```
<Attribute
Name="EmployeeId"
IsNotNull="true"
Precision="6"
Scale="0"
ColumnName="EMPLOYEE_ID"
SQLType="NUMERIC"
Type="oracle.jbo.domain.Number"
ColumnType="NUMBER"
TableName="EMPLOYEES"
PrimaryKey="true">
<DesignTime>
<Attr Name="_DisplaySize"
Value="22"/>
</DesignTime>
</Attribute>
<Attribute
Name="FirstName"
Precision="20"
ColumnName="FIRST_NAME"
...
```

quovera

17

List-of-Values (LOV)

- Defined for a view object attribute
- Associate a query (view object) with the attribute
- Set up is declarative
 - Start in the view object editor
- UI code will display this attribute specially
 - Pulldown list item loaded with values
 - Popup list-of-values dialog



quovera

18

VO Editor – LOV Subtab

Name	List Data Source	List Attribute
LOV_JobId	JobLOVAccessor	JobId

quovera

19

View Criteria

- Used for queries
- View Object Editor
 - 11g Query page
 - 12c VC page
- View Criteria dialog
 - Defines the WHERE clause declaratively
 - Bind variables are optional

Criteria Name: BasicEmployeeSearchCriteria Query Execution Mode: Database

View Criteria:

- BasicEmployeeSearchCriteria
 - Group
 - FirstName CONTAINS :queryFirstName
 - AND LastName CONTAINS :queryLastName
 - AND Email CONTAINS :queryEmail

View Object Where Clause:

```
((Employees.DEPARTMENT_ID = Departments.DEPARTMENT_ID) AND (Employees.JOB_ID = Jobs.JOB_ID) AND (((UPPER(Employees.FIRST_NAME) LIKE UPPER('% || :queryFirstName || %') ) OR (:queryFirstName IS NULL ) AND (UPPER(Employees.LAST_NAME) LIKE UPPER('% || :queryLastName || %') ) OR (:queryLastName IS NULL ) ) AND (UPPER(Employees.EMAIL) LIKE UPPER('% || :queryEmail || %') ) OR (:queryEmail IS NULL ))))
```

Criteria Item:

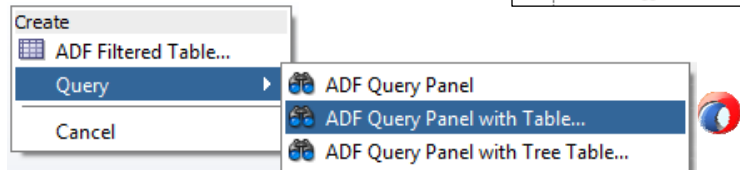
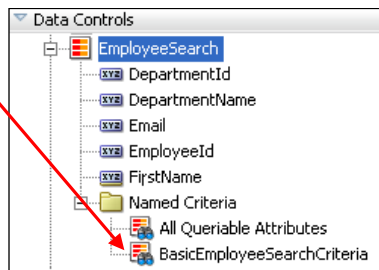
Conjunction: AND Attribute: Email Operator: Contains Operand: Bind Variable Parameter: queryEmail

quovera

20

View Criteria in the DCP

- Data Controls panel lists the defined criteria
- You can drag and drop a view criteria onto a JSF page



- Components are added for each attribute in the view criteria

Demo

Agenda

- What is ADF?
- ADF Business Services
- ADF **View**, Model, and Controller



ADF Faces Rich Client Overview

- Built on top of JSF APIs
- Deployable on any 1.2 implementation of JSF
- Support for pop-ups and dialogs
- Skins, skin editor
- Templates
- AJAX-like interactivity
- Over 150 components
- Data Visualization Tools (DVT) components
 - Charts, Gantt, Pivot, Maps, Hierarchy

Really rich!



Some Components

Sample ADF Faces Development

ADF Faces JSF Snippet

```
<jsp:root xmlns:jsp="http://java.sun.com/JSP/Page" version="2.0"
  xmlns:f="http://java.sun.com/jsf/core"
  xmlns:af="http://xmlns.oracle.com/adf/faces/rich">
  ...
  <af:panelStretchLayout styleClass="AFVisualRoot" topHeight="105px"
    bottomHeight="20px">

    <f:facet name="top">
      <af:panelBorderLayout>
        <f:facet name="start">
          <af:image source="/images/tuhra.gif" shortDesc="TUHRA Logo"/>
        </f:facet>
        <f:facet name="end">
          <af:panelGroupLayout layout="horizontal" halign="right"
            valign="bottom">
            <af:commandImageLink text="Logon" shortDesc="Logout from TUHRA"
              depressedIcon="/images/groupdisconnect_dwn.png"
              disabledIcon="/images/groupdisconnect_dis.png"
              hoverIcon="/images/groupdisconnect_ovr.png"
              icon="/images/groupdisconnect_ena.png"
              disabled="true"
              rendered="#{attrs.anonymous}"/>
            <af:commandImageLink text="Logoff" shortDesc="Logout from TUHRA"
              depressedIcon="/images/groupdisconnect_dwn.png"
              disabledIcon="/images/groupdisconnect_dis.png"
              hoverIcon="/images/groupdisconnect_ovr.png"
              icon="/images/groupdisconnect_ena.png"
              disabled="true"
              rendered="#{!attrs.anonymous}"/>
          </af:panelGroupLayout>
        </f:facet>
      </af:panelBorderLayout>
    </f:facet>
  </af:panelStretchLayout>
</jsp:root>
```

Agenda

- What is ADF?
- ADF Business Services
- ADF View, **Model**, and Controller



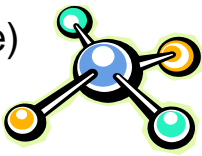
ADF Model

- ADF Data Controls
 - Provides list of components or groups of components for a node in the data model
 - “Drop as” options
- ADF Bindings
 - Prebuilt connection from the ADF BC to the UI
 - Drag and drop action above does the work

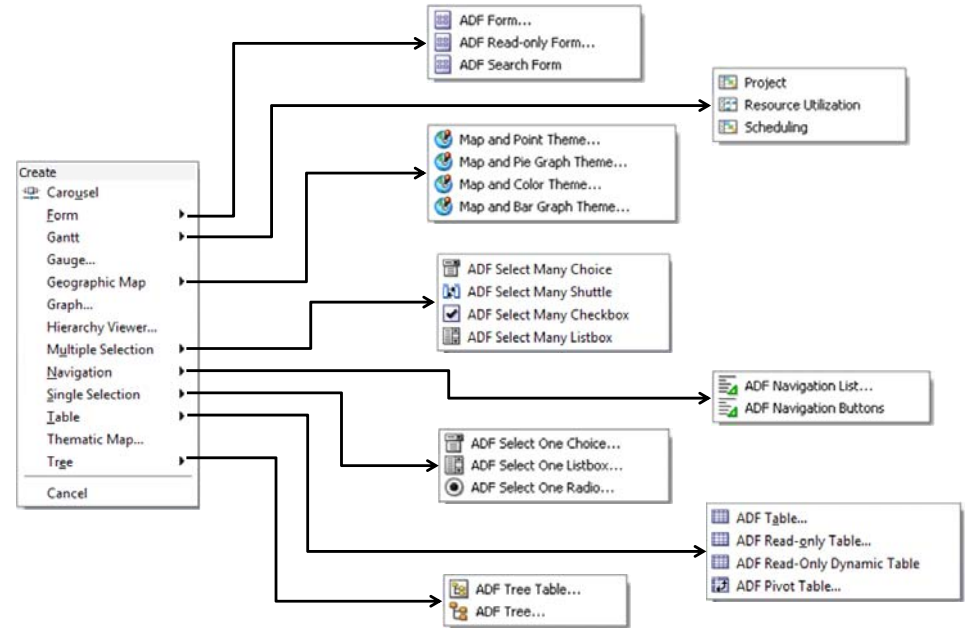
ADF Data Controls

- Business Services abstraction
 - Makes Model components available to ViewController
 - Automatically created with ADF BC
 - Can be created for other business services
 - For non-ADF BC, defined in DataControls.cpx
- Provide list of “Drop as” options that create pre-bound components
 - Collection level (view object instance)
 - Attribute level (view attribute)

The Good News:
You don't normally write data controls

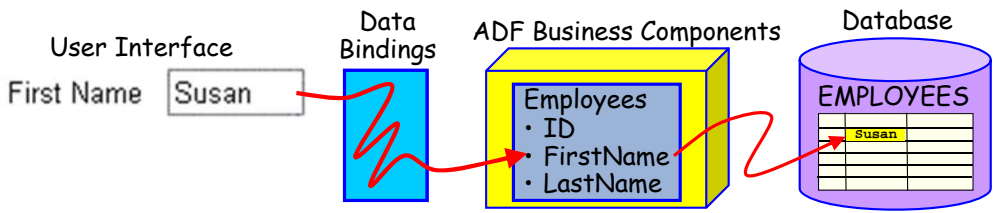


Collection “Drop As” Options



ADF Bindings

- Association of a business service data element or action with a UI component
 - Relatively automatic in Oracle Forms
 - Definitely not automatic in native Java EE
- Binding normally takes a lot of coding
 - One-off solution is not the answer
 - Need a framework to assist



Drop As Examples: Form and Table

deptBrowse.jspx

#{...DepartmentId...label} #{...DepartmentId.inputValue}

#{...DepartmentName...label} #{...DepartmentName.inputValue}

#{...ManagerId...label} #{...ManagerId.inputValue}

#{...LocationId1...label} #{...LocationId1.inputValue}

First	Previous	Next	Last	Edit Department	New Department
#{...EmployeeId...label}	#{...FirstName.label}	#{...LastName.label}	#{...Email.label}	#{...JobId.label}	
#{...EmployeeId}	#{...FirstName}	#{...LastName}	#{...Email}	#{...JobId}	
#{...EmployeeId}	#{...FirstName}	#{...LastName}	#{...Email}	#{...JobId}	

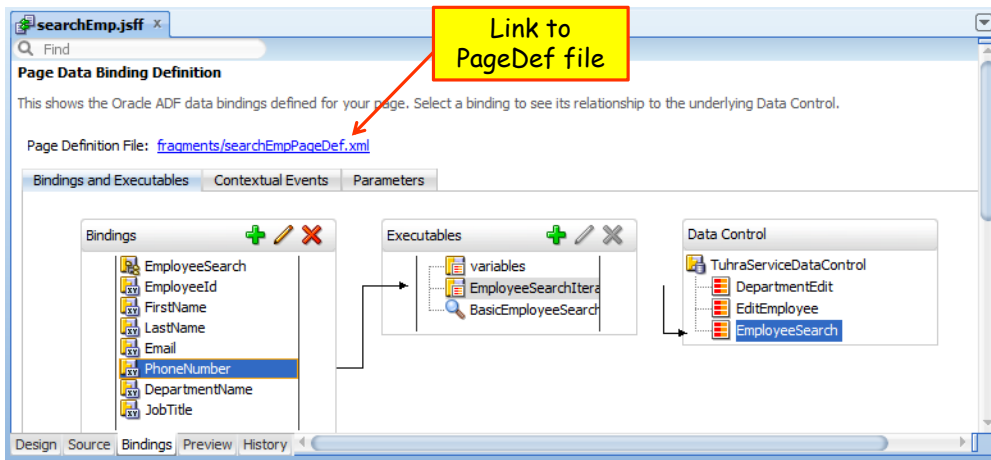
contextMenu

jsp:root ▶ f:view ▶

Design Source Bindings Preview History

Note the binding expressions

Editing Bindings



Binding Code

JSF Page Snippet

```
<af:inputText value="#{bindings.DepartmentId.inputValue}"
  label="#{bindings.DepartmentId.hints.label}"
  required="#{bindings.DepartmentId.hints.mandatory}"
  columns="#{bindings.DepartmentId.hints.displayWidth}"
  maxLength="#{bindings.DepartmentId.hints.precision}"
  shortDesc="#{bindings.DepartmentId.hints.tooltip}"
  id="it1">
</af:inputText>
```

Bindings (PageDef) Snippet

```
<bindings>
  <attributeValues IterBinding="DepartmentsView1Iterator"
    id="DepartmentId">
    <AttrNames>
      <Item Value="DepartmentId"/>
    </AttrNames>
  </attributeValues>
```

Agenda

- What is ADF?
- ADF Business Services
- ADF View, Model, and **Controller**

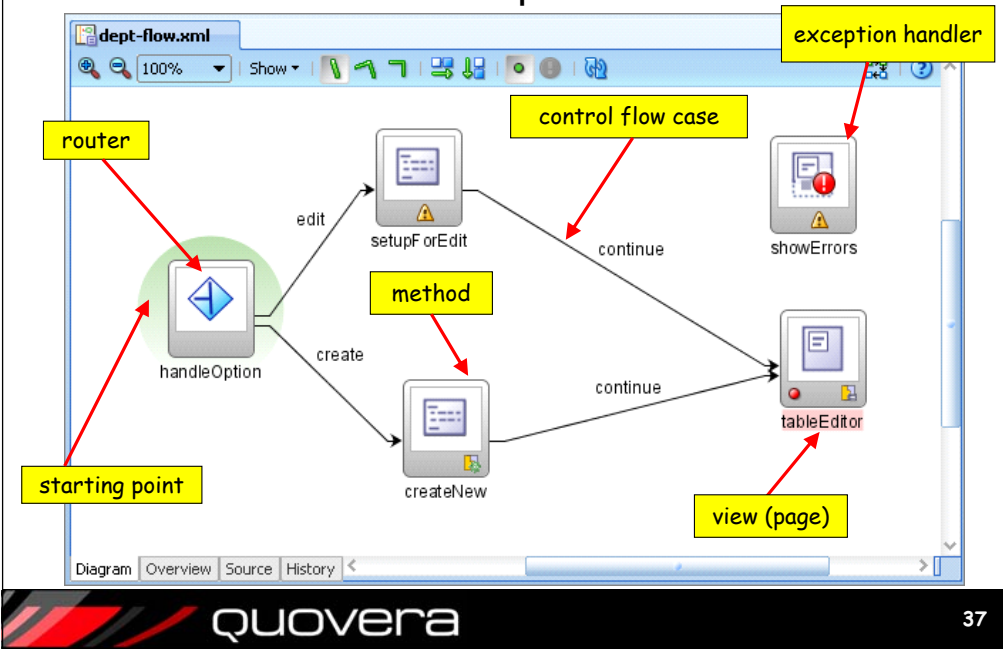


ADF Controller (ADFc)

- Extension to standard JSF Controller functionality
- Defines *task flows*
 - Logic and page fragment components
 - Embedded on the page in a region component
- Benefits
 - Page fragment re-use
 - Executing code in a logic-defined flow
 - “Task flow” not “page flow”
 - Security
 - Exception handling and transaction management
- Defined in a diagram
 - Like JSF but more components available



Sample ADF Controller Development



Sample ADF Controller Code

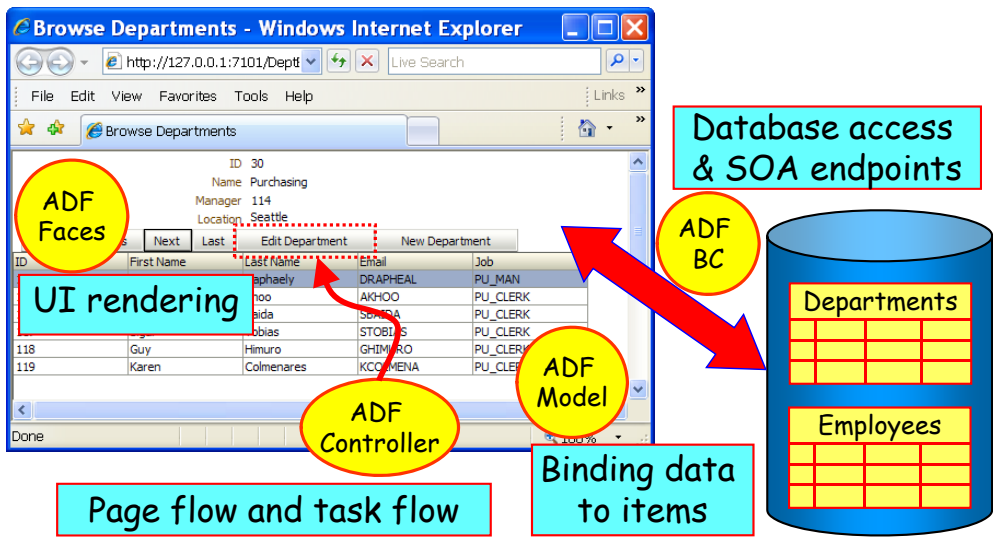
```

<task-flow-definition id="dept-flow">
<default-activity>deptBrowse</default-activity>
<view id="deptBrowse">
  <page>/deptBrowse.jspx</page>
</view>
<view id="deptEdit">
  <page>/deptEdit.jspx</page>
</view>
<control-flow-rule>
  <from-activity-id>deptBrowse</from-activity-id>
  <control-flow-case>
    <from-outcome>toEdit</from-outcome>
    <to-activity-id>deptEdit</to-activity-id>
  </control-flow-case>
</control-flow-rule>
<router id="checkForExplicitID">
  <case id="_6">
    <expression>#{!empty pageFlowScope.employeeId}
    </expression>
    <outcome>byId</outcome>
  </case>
  <default-outcome>currentUser</default-outcome>
</router>
<method-call id="queryEmployeeById">
  <method>#{bindings.queryEmployeeById.execute}</method>
  <outcome>
    <fixed-outcome>queryEmployeeById</fixed-outcome>
  </outcome>
</method-call>

```

Demo

Summary: ADF Core Technologies



Summary

- Oracle builds Cloud (Fusion) Applications with an ADF core technology stack
- ADF offers a consistent developer experience regardless of the technologies
- **ADF Business Components** provide access to the database and other data sources
- **ADF Model** connects ADF BC to ADF Faces
- **ADF Faces** provide 150+, feature-rich item and container components for JSF pages
- **ADF Controller** manages page flow and task flow



Final Voltaire Wisdom

The secret of being a bore
is to tell everything.

Le secret d'ennuyer
est celui de tout dire.

—Voltaire (1694-1778), *Sept
Discours en Vers sur l'Homme*



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



www.quovera.com

