# Oracle's Application Development Framework (ADF) at a Glance


**Presented by: PITSS and King Training Resources**

**Presenter: John King -** john@kingtraining.com

Copyright @ 2011, John Jay King

# Objectives

- Be aware of Oracle's Application Development Framework (ADF) standards-based applications
- Understand ADF development using advanced graphical editing  and declarative techniques
- Know ADF’s Java and XML underpinnings
- See how JDeveloper creates ADF applications
- Grasp ADF BC’s reusable data source support
- Learn how to build UIs graphically with drag and drop components, and declarative properties

 Copyright @ 2011, John Jay King

# Who Am I?

- John King – Partner, King Training Resources
- Providing training to Oracle and IT community for over 20 years – http://www.kingtraining.com
- "Techie" who knows Oracle, SQL, Java, and PL/SQL pretty well (along with many other topics)
- Leader in Service Oriented Architecture (SOA) design and implementation
- Member of ODTUG (Oracle Development Tools User Group) Board of Directors
- Active member of Rocky Mountain Oracle Users Group (RMOUG)

 Copyright @ 2011, John Jay King

# Who Are You?

- Forms Developer
- 4GL Developer
- Java Developer
- All of the above
- None of the above

 Copyright @ 2011, John Jay King

# Oracle Fusion Architecture

- So what is Oracle's “Fusion” and where does it fit?
- Oracle uses the title “Fusion” to unify its SOA-directed offerings and highlight the integration features incorporated in their products
- Two major legs of Oracle Fusion Architecture are:
  - Oracle Fusion Middleware
  - Oracle Fusion Applications

 Copyright @ 2011, John Jay King

# Oracle Fusion Middleware

- Oracle Fusion Middleware (FMW) includes several key application development components including:
  - Oracle WebLogic Server (application server)
  - Oracle JDeveloper (multi-purpose IDE)
  - Application Development Framework (ADF)
  - Other Oracle Application Development tools:
    - TopLink (object-relational mapping)
    - JRocket (high-speed JVM)
    - Tuxedo (transaction-monitor)
    - Coherence (in-memory data grid; svc sharing/caching)
    - More…

 Copyright @ 2011, John Jay King

# Oracle Application Development Framework (ADF), page 1

- ADF is a “meta-Framework” interacting with a variety of underlying software components (including Frameworks) to provide:
  - Database connectivity and transfer
  - Mapping of application views to data sources
  - Database interaction: constraints, keys, data types, master/detail, null handling
  - Data caching via entity objects

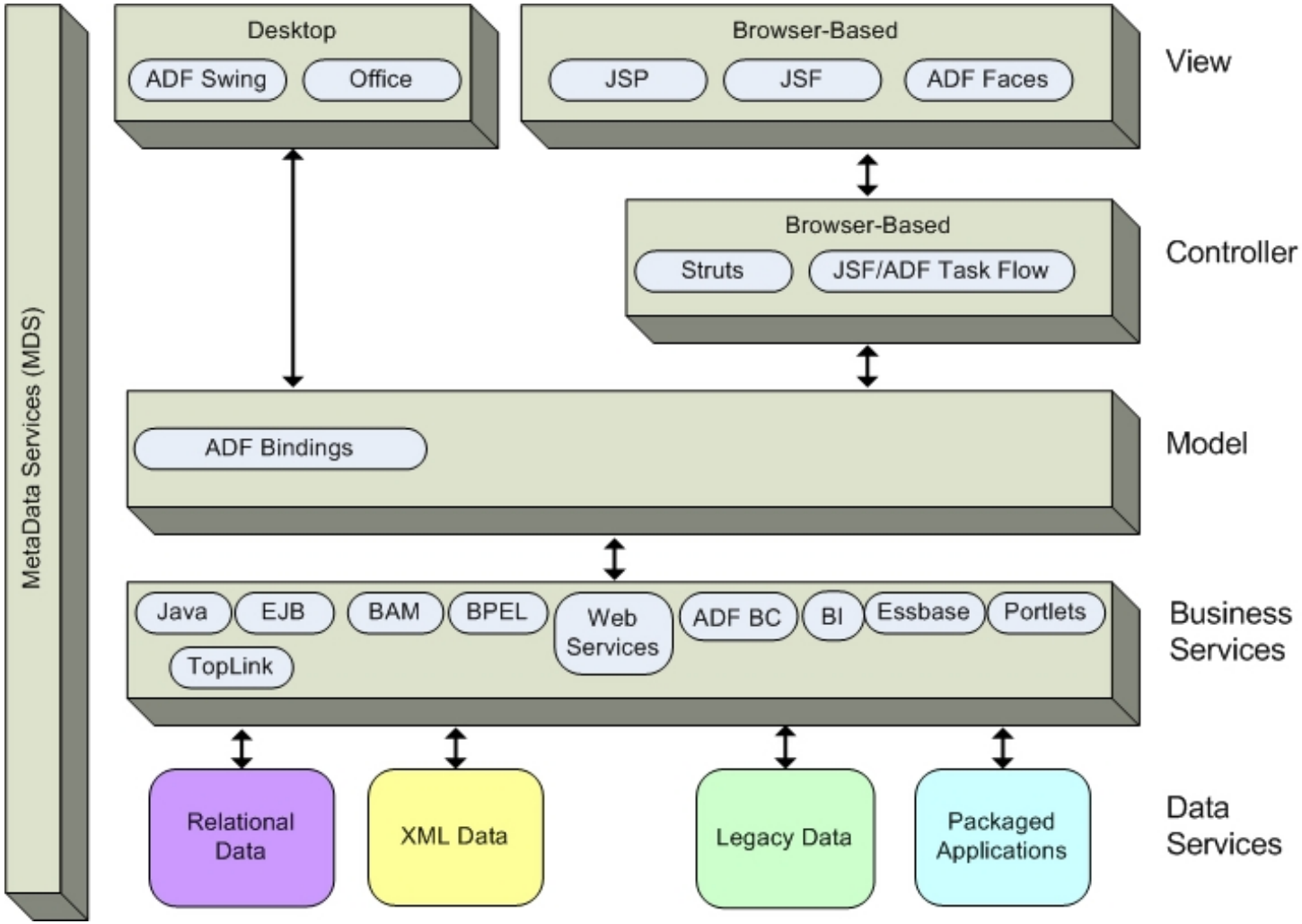
 Copyright @ 2011, John Jay King

# Oracle Application Development Framework (ADF), page 2

- ADF features (continued)
  - Transaction management (locks, commit, rollback, etc...)
  - Declarative validation
  - Business logic and event handling
  - User Interface (UI) logic, flow, look & feel
  - Data-bound UI Components
  - UI properties including: formatting, colors, defaults, visual components, LOVs, etc...

 Copyright @ 2011, John Jay King

# ADF Technology "Stack"




 Copyright @ 2011, John Jay King

# Why Oracle ADF?

- Oracle Application Development Framework (ADF) is a Java-based development tool (much like Oracle Forms is a PL/SQL-based tool) designed to take full advantage of Java Enterprise Edition or Java EE
- Java EE is one of the most widespread application environments today
- ADF's 4GL features make application development much easier than normal Java "coding"
- Oracle is rewriting their ERP stack as "Fusion Applications" using ADF; the already rich toolset gets richer every day

 Copyright @ 2011, John Jay King

# Do I Need to Know Java ?

- Probably not well
  - Someone with very basic Java and Web Skills can easily create applications with ADF (much the same as someone with basic PL/SQL could create very basic Oracle Forms)
  - Someone on your team needs to know Java very well
- Someone on your team needs to understand ADF and its available components very well

 Copyright @ 2011, John Jay King

# Is Forms Going Away?

- NO, NO, NO, NO, NO
- Oracle is committed to supporting Oracle Forms for many years to come
- A new version of Oracle Forms (12g) is on the way!

 Copyright @ 2011, John Jay King

# ADF: Two Major Pieces

- ADF has many parts but two are central to creating applications
  - ADF BC Business Components (data)
  - ADF Faces Graphical User Interface

 Copyright @ 2011, John Jay King

# ADF Business Components (ADF BC)

- ADF Business Components is a framework that simplifies developing Java EE business services
- ADF BC is part of the ADF Business Services layer and is used to provide:
  - Persistence and data retrieval with SQL using data views
  - Object-Relational Mapping (ORM) between Java classes and database data
  - Simplified data access, validation, and business logic
  - Transactional infrastructure
  - Implementation of best practices

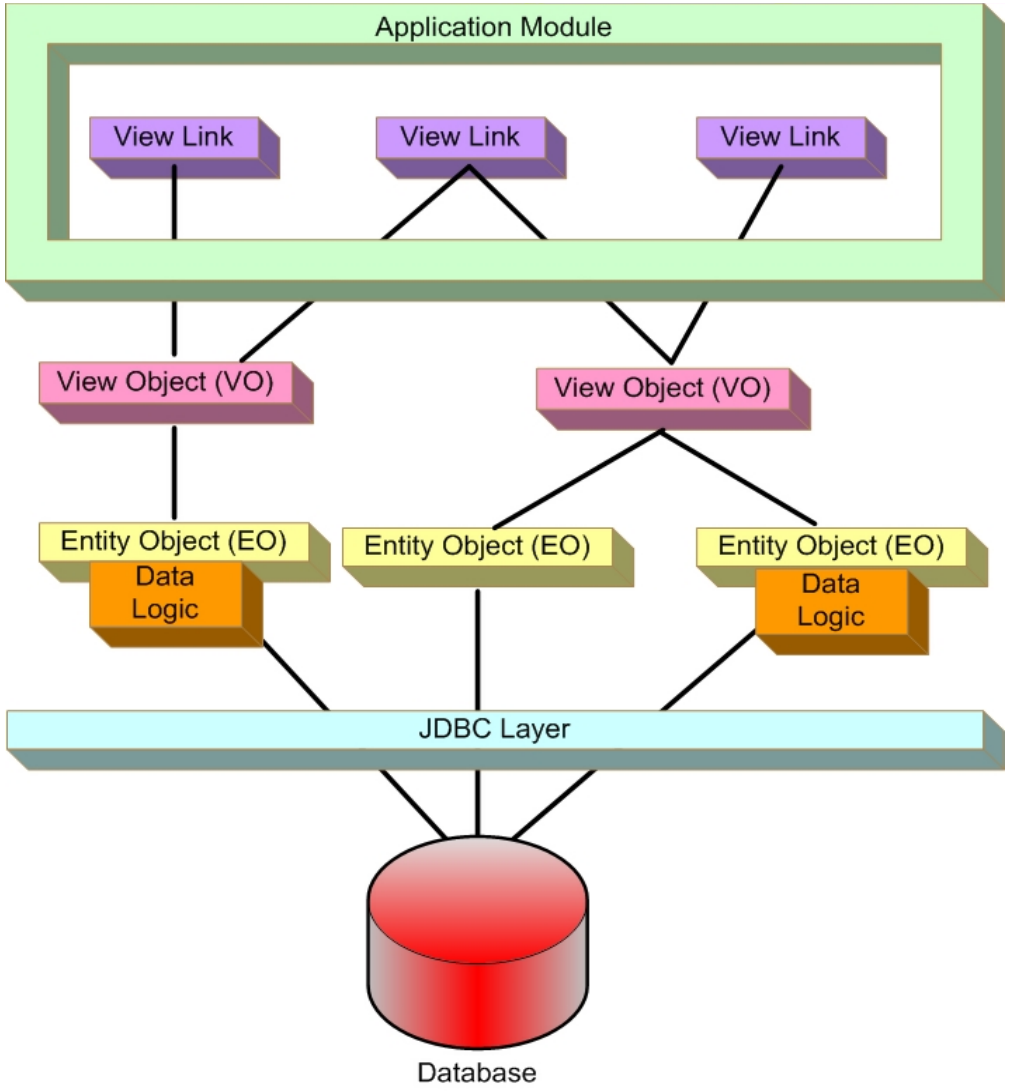
 Copyright @ 2011, John Jay King

# ADF BC Objects

- ADF BC is implemented using a variety of objects to:
  - Define Insert-Update-Delete views to perform queries and data manipulation
  - Define query views (read-only)
  - Define links between queries

 Copyright @ 2011, John Jay King

# ADF BC - Component Structure




 Copyright @ 2011, John Jay King

# ADF BC Components

- ADF BC uses a variety of object types to represent data:
  - Database tables and views — Application Base Data
  - Entity Objects — Business rules, validations, defaults for a table (or view)
  - View Objects — SQL output to query, filter, join,modify, or sequence data
  - Application Modules — Use View Objects to access/modify data acting as a back-end data service
  - Appl. Module Data Model — Describes actual View Object uses
- Objects may be reused in multiple Application Modules

 Copyright @ 2011, John Jay King

# ADF Data Binding

- After identifying Entity Objects and View Objects two additional ADF Data Model components are used
  - Data Controls — Java objects used to abstract View Object Business Services
  - Binding Containers — Java object; provides data access to a single ADF application page, fragment, or activity

 Copyright @ 2011, John Jay King

# Java Server Faces (JSF)

- Java Server Faces (JSF) is a Web-tier framework of JSP technology and JSP Tag libraries to create and use User Interface components
- JSF is extended by components of Oracle ADF Faces
- JSF includes:
  - Runtime architecture
  - Library of JSF components
  - JSF “Life Cycle”
  - Many JSF-Oriented Files

 Copyright @ 2011, John Jay King

# ADF Faces

- Even though JSF sought to simplify user interface; it is often felt to be too complex
- Oracle has extended JSF as “ADF Faces” providing a set of libraries and tags that include enhanced UI components and easier use
- Oracle has presented ADF Faces to the Open Source community where it is part of the Apache Foundation Trinidad MyFaces project

**http://myfaces.apache.org/trinidad/index.html**

 Copyright @ 2011, John Jay King

# Using ADF

- Using ADF Faces is simple using JDeveloper:
  - Add Application layout containers to describe user interface
  - Add ADF Faces components to layout containers
  - All UI is done with ADF Faces; no HTML coding
- Features added by ADF Faces:
  - Pop-ups and Dialog boxes
  - Data Visualization Tools: Charts, graphics, etc...
  - Declarative AJAX support
  - More…

 Copyright @ 2011, John Jay King

# ADF Controller

- The ADF Controller extends the JSF controller and controls ADF's MVC (Model-View-Controller) in ADF
- ADF Controller features include:
  - Sequence of page displays (may be conditional)
  - Allows partial-page processing in the same way as full page processing; only the necessary part of a page is rendered, the rest is unchanged
  - Allows reuse of page parts
  - Provides conditional control of page flow

 Copyright @ 2011, John Jay King

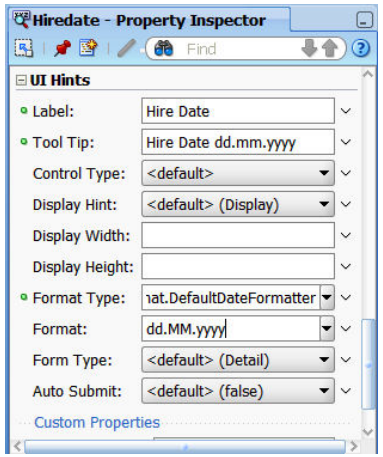
# ADF Faces “Rich-Client” Features

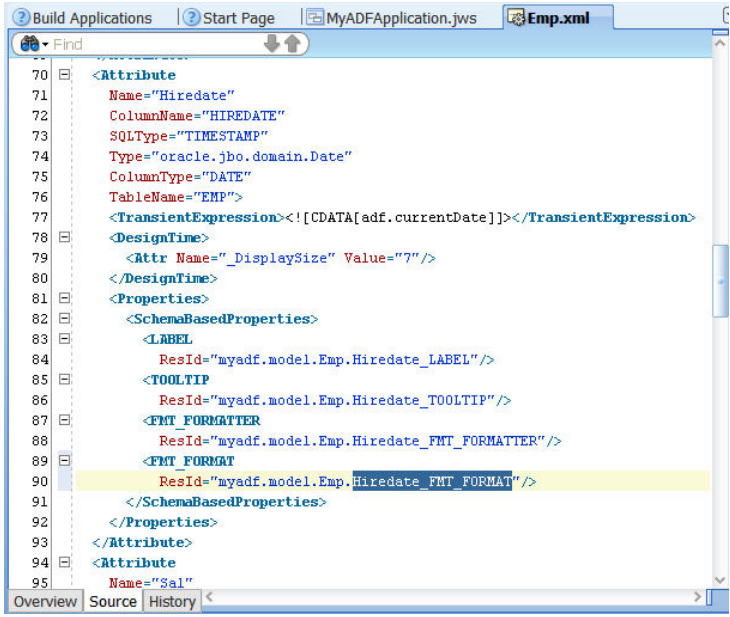
- ADF Faces is designed to create “rich-client” (RC) interfaces; full-featured and declarative including:
  - Complete JDeveloper support graphic development (screen-painter) and property palettes
  - Visual Editor
  - Property Inspector
  - Changeable “skins” to easily alter look-and-feel
  - Modifiable look-and-feel properties (declarative)
  - Layout control

 Copyright @ 2011, John Jay King

# Declarative Features of ADF and JDeveloper

- One of the advantages of ADF and ADF BC is the declarative nature of the framework
  - Rather than writing code to perform typical data definition tasks; developers use JDeveloper Property Palettes to set (or check) values
  - ADF uses XML files to store declared definitions





Copyright @ 2011, John Jay King

# Oracle JDeveloper

- JDeveloper provides a world-class, easy to use IDE
- Oracle has extended JDeveloper beyond Java to include:
  - Oracle ADF modeling, business services, and GUI design
  - XML edit including Syntax Checking & Schema Validation
  - SQL development including debugging of stored PL/SQL
  - UML Modeling and MDA (Model Driven Architecture)
  - Web Services development
  - ESB design
  - BPEL design
  - Portlets

 Copyright @ 2011, John Jay King

# Downloading JDeveloper

- JDeveloper is Free!
- To learn more about JDeveloper, see Oracle's website:

  **http://www.oracle.com/technetwork/developer-tools/jdev/overview/index.html**

 Copyright @ 2011, John Jay King

# Oracle WebLogic Server

- Oracle WebLogic Server is Oracle's preferred platform to provide both a standard Java EE environment and an environment specifically tailored to Oracle Fusion Middleware; providing:
  - Complete Java EE 5 compatibility
  - Complete Java SE 6 compatibility
  - Web Services support
  - Integration with Oracle's Fusion Middleware tools

 Copyright @ 2011, John Jay King

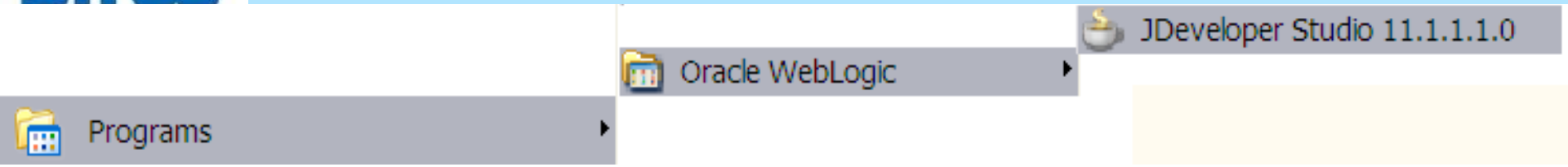
# Oracle AS and OC4J?

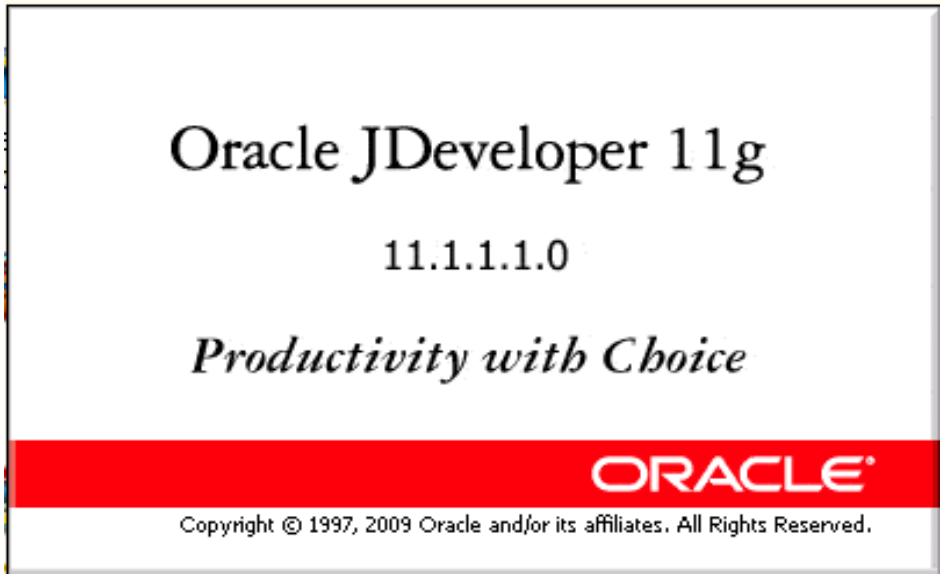
- Oracle WebLogic Server is the replacement for Oracle Application Server (OAS) and OC4J
- OAS and OC4J are still supported and may be used instead of WebLogic if desired but ADF requires Java 1.5 / Java 5 (needed for ADF)
- To learn more about Oracle WebLogic Server see Oracle's website:

  **http://www.oracle.com/us/products/middleware/application-server/index.htm**

 Copyright @ 2011, John Jay King

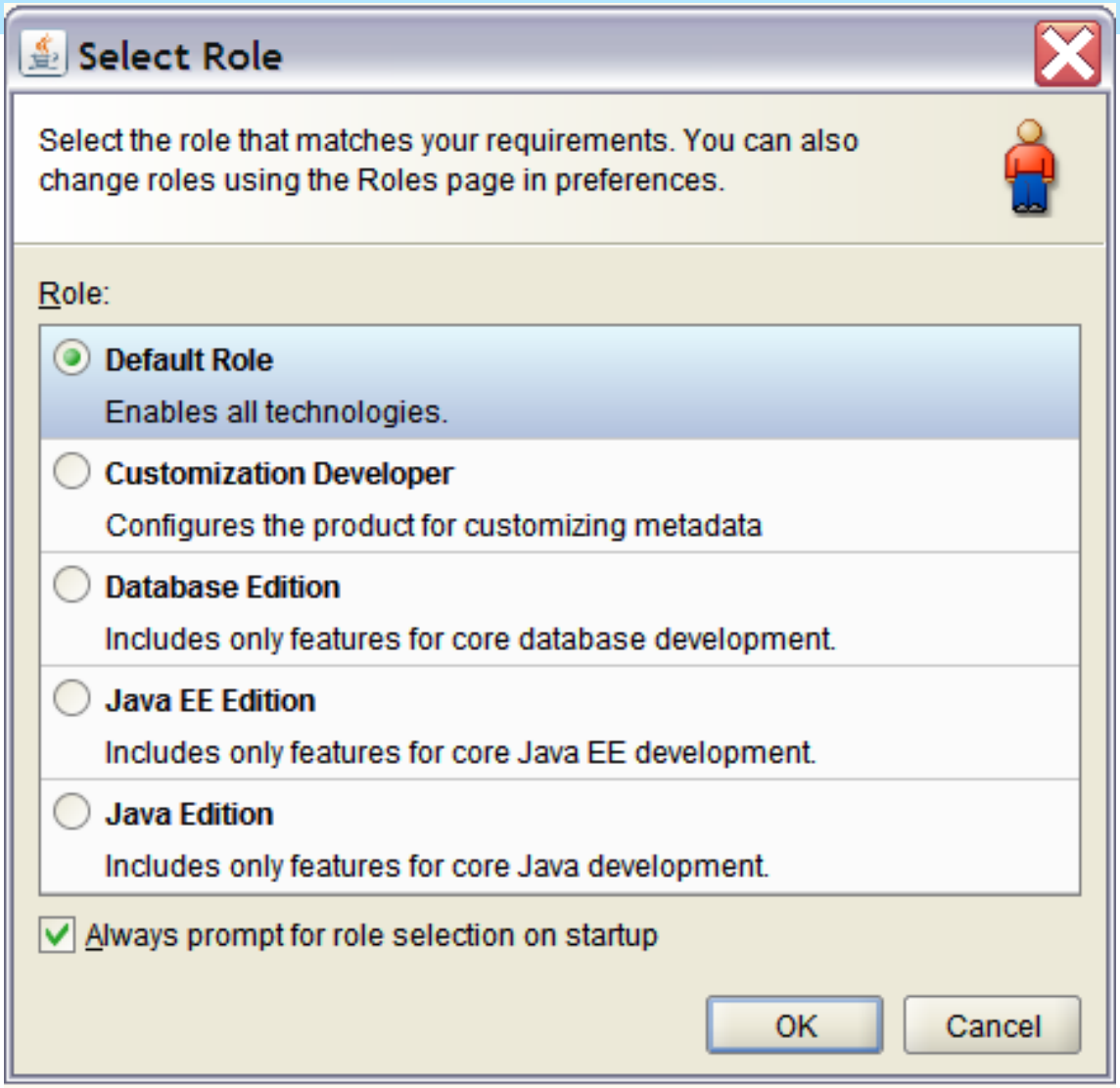
# Starting JDeveloper



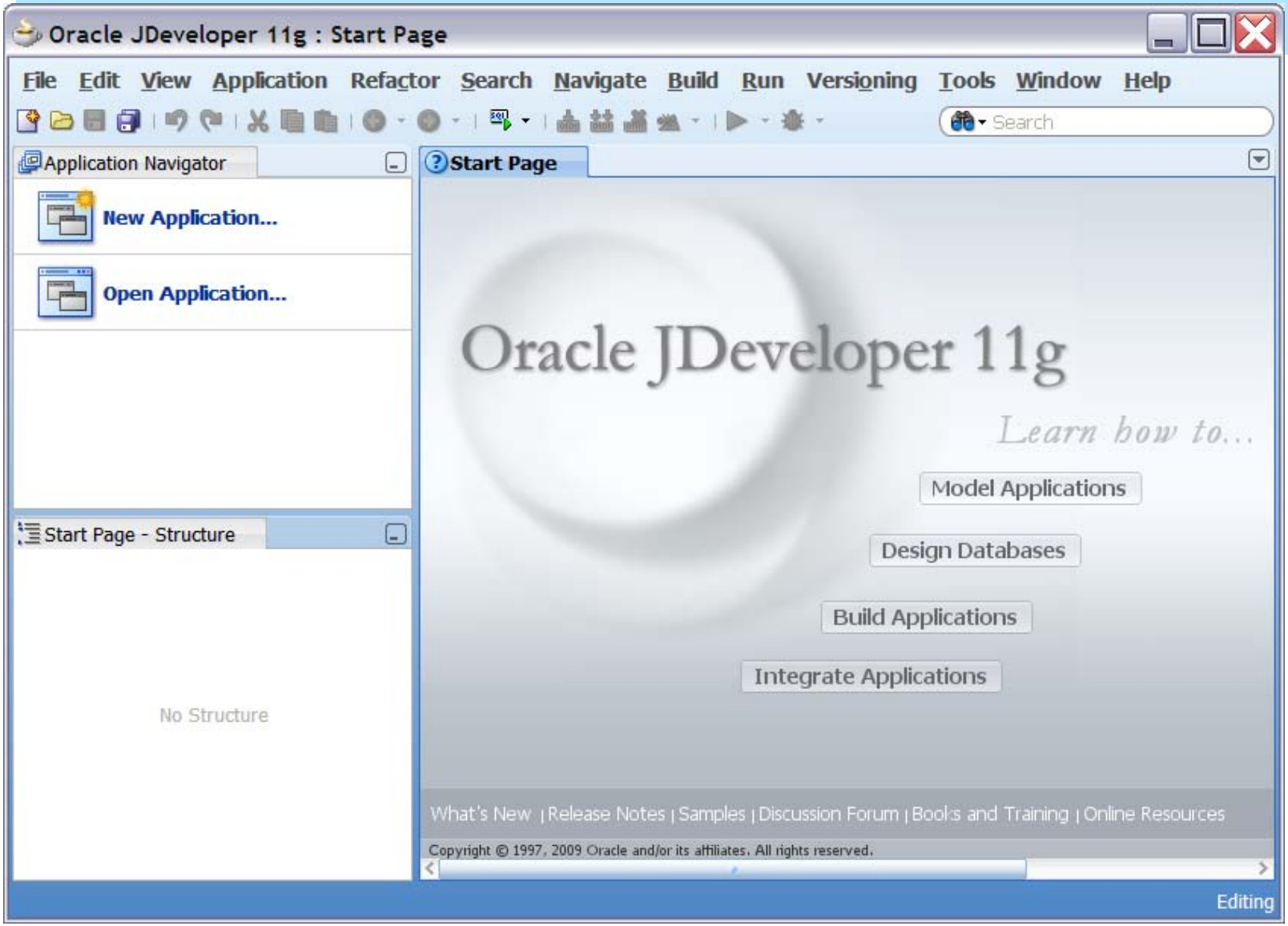

 Copyright @ 2011, John Jay King

# JDeveloper - Select Role



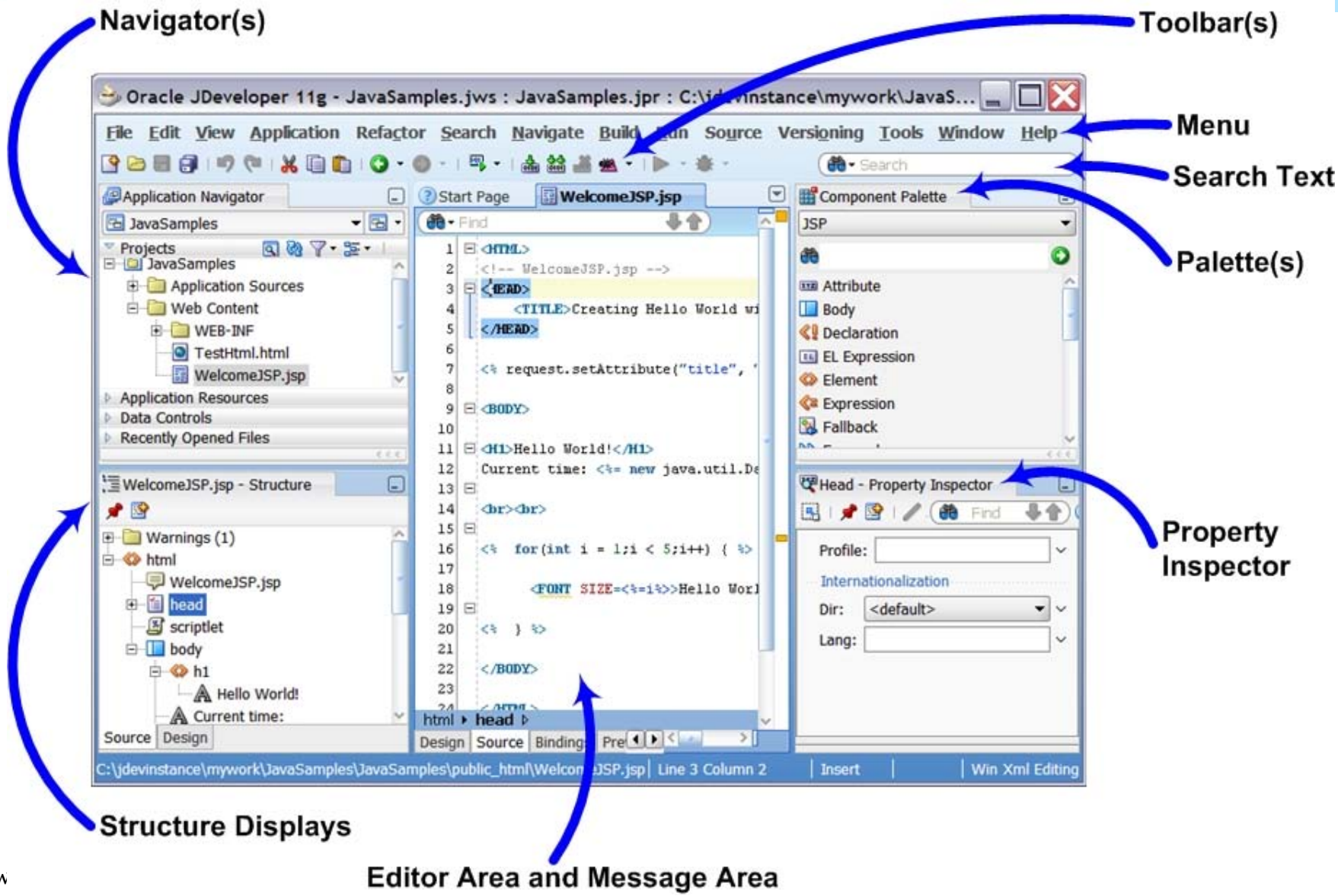

 Copyright @ 2011, John Jay King

# JDeveloper - Start Page




 Copyright @ 2011, John Jay King

# Exploring JDeveloper

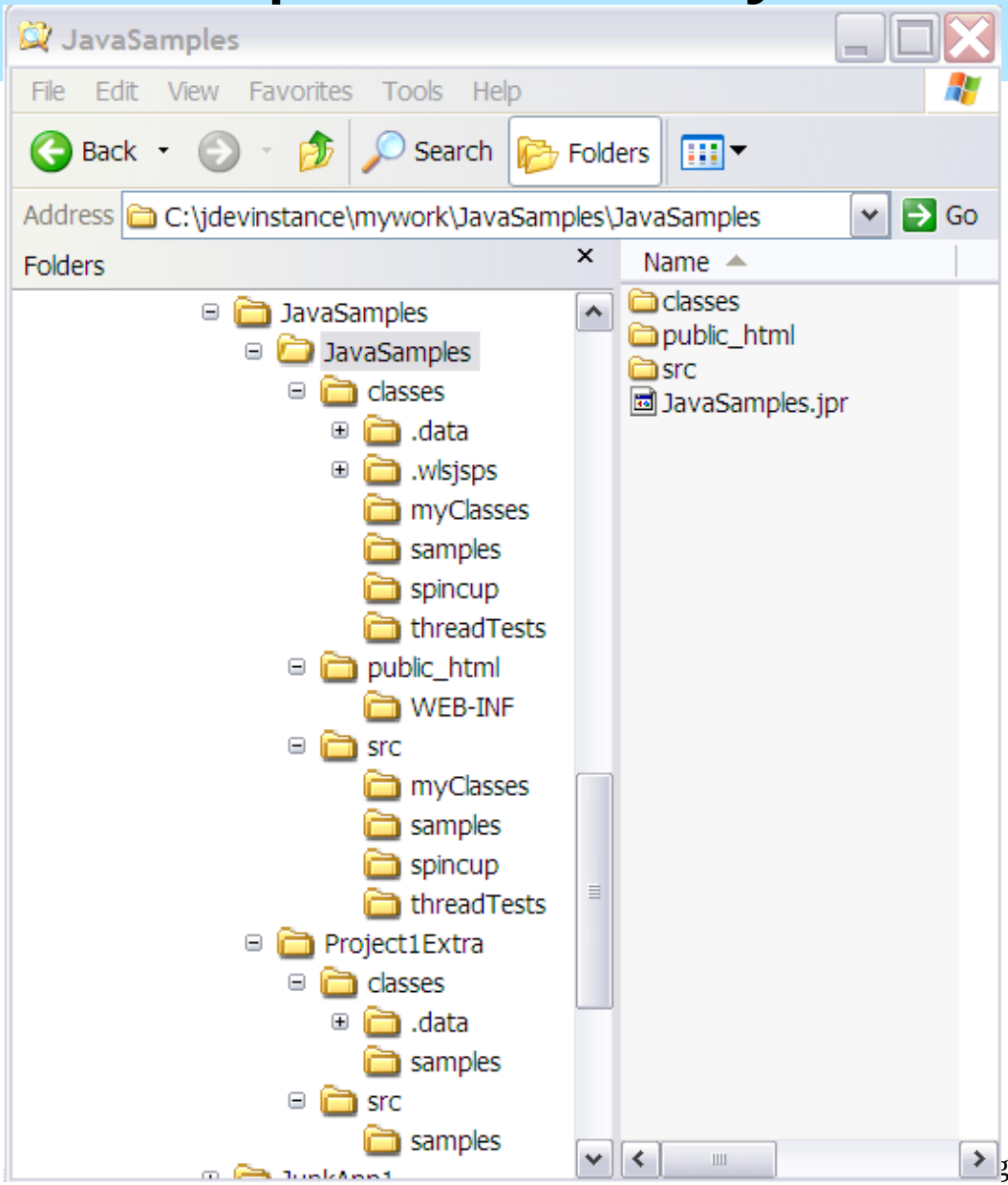


# Applications and Projects

- JDeveloper uses a non-standard, Oracle-specific “Application” to group a collection of “Projects” (similar to how it is done by other IDEs)
- All files representing an “Application” share a common root directory (folder) on a disk
- Many Applications may be open at once in JDeveloper; but only one at a time will be visible in the Application Navigator

 Copyright @ 2011, John Jay King

# JDeveloper Directory Structure

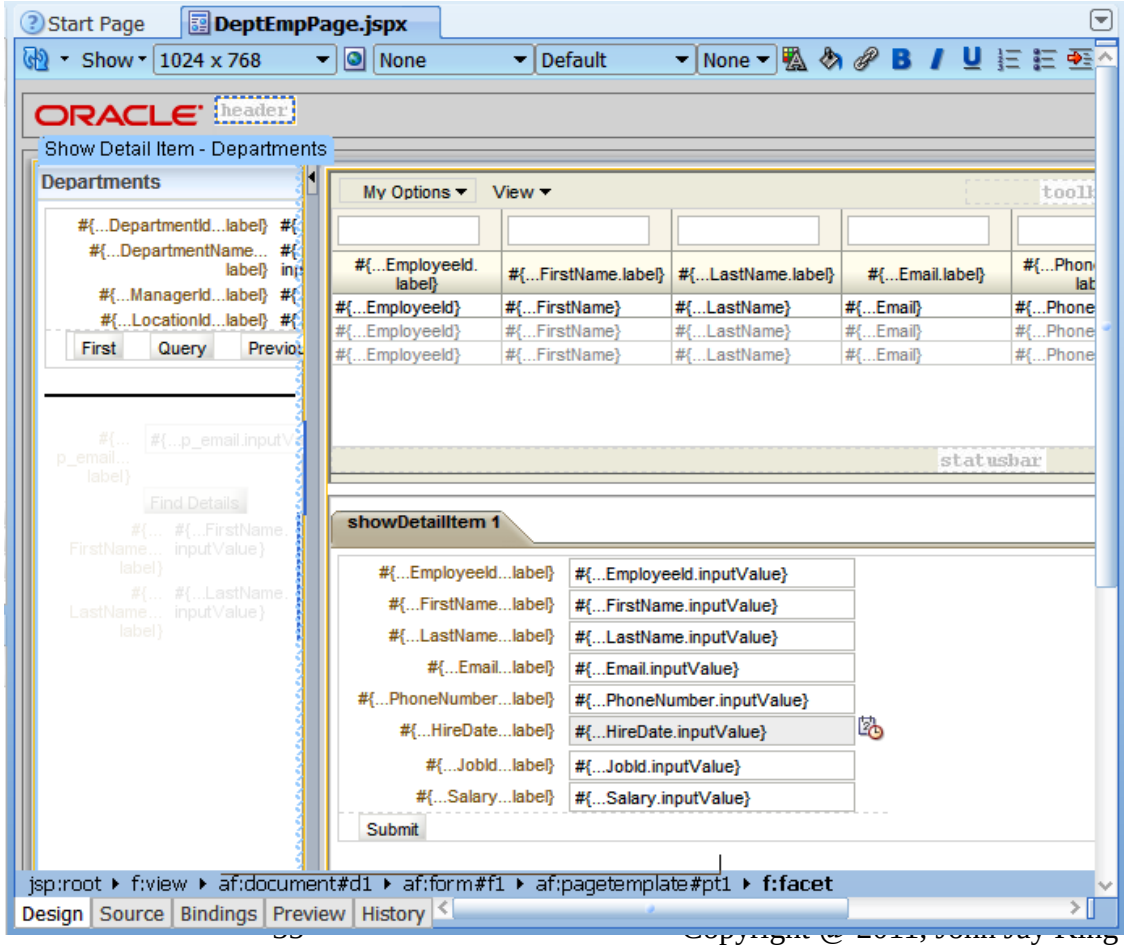




ght @ 2011, John Jay King

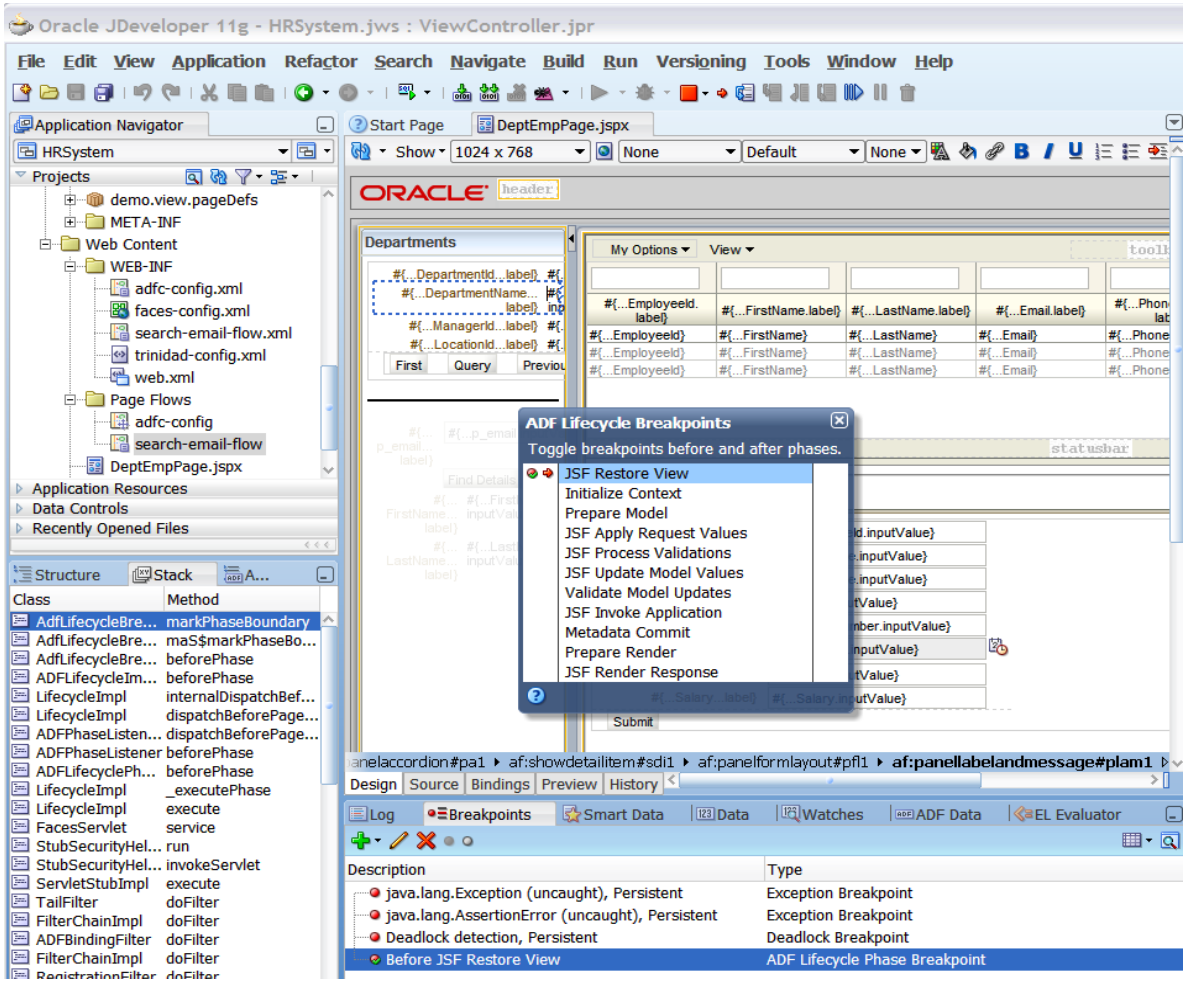
# JDeveloper Editing

- JDeveloper has many Code Editors & Visual Editors including: Java, XML, HTML, JSP, JSF/ADF Faces, BPEL, & more



# JDeveloper Debugging

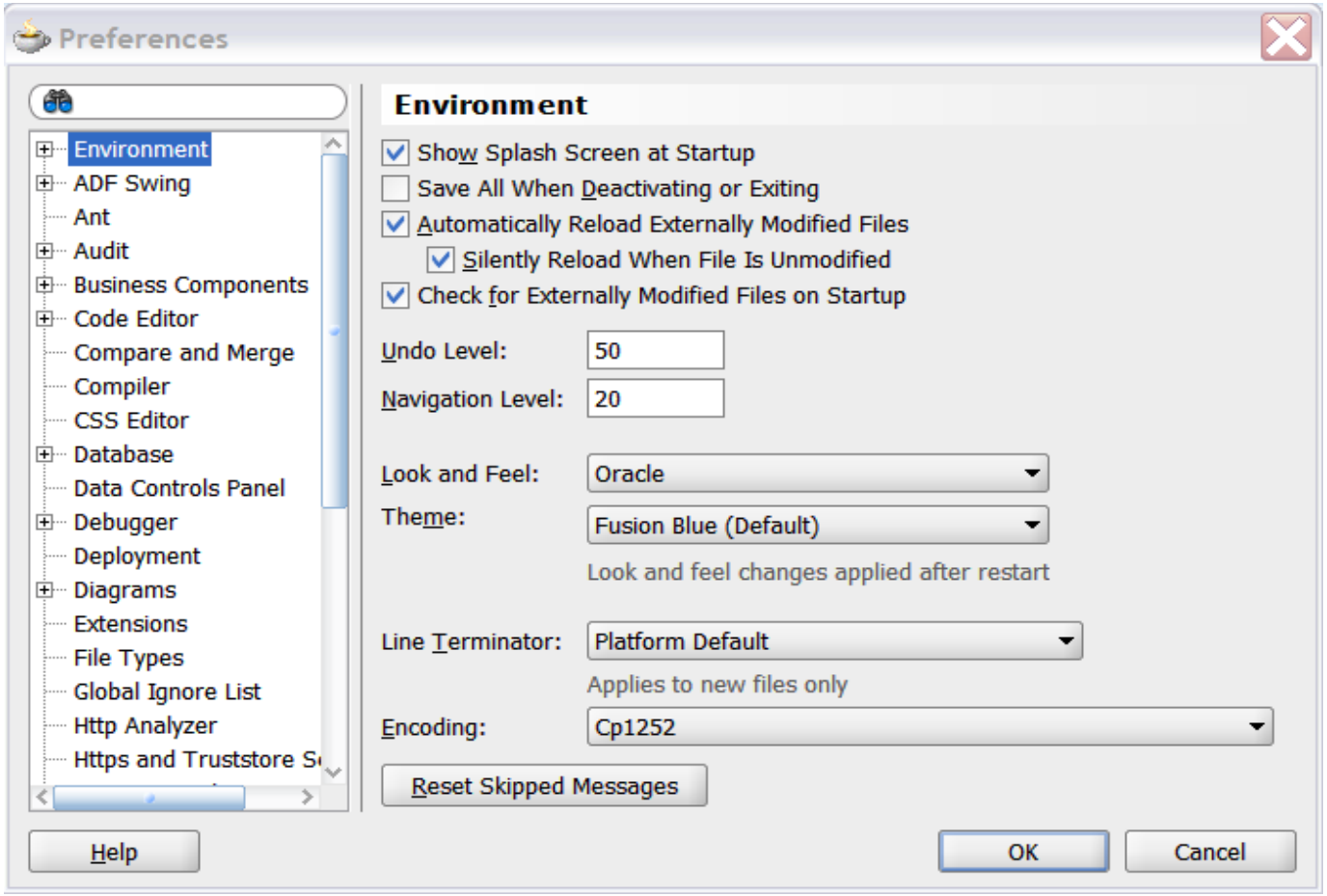
- JDeveloper allows local and remote debugging




 Copyright @ 2011, John Jay King

# JDeveloper Preferences

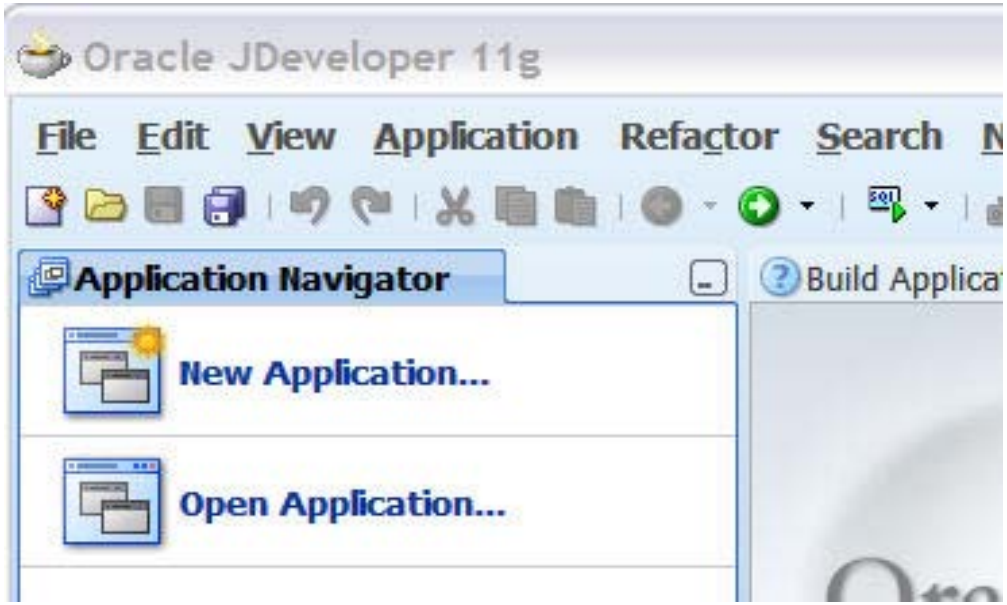
- JDeveloper is customizable; preferences may be viewed/modified using Tools->Preferences




 Copyright @ 2011, John Jay King

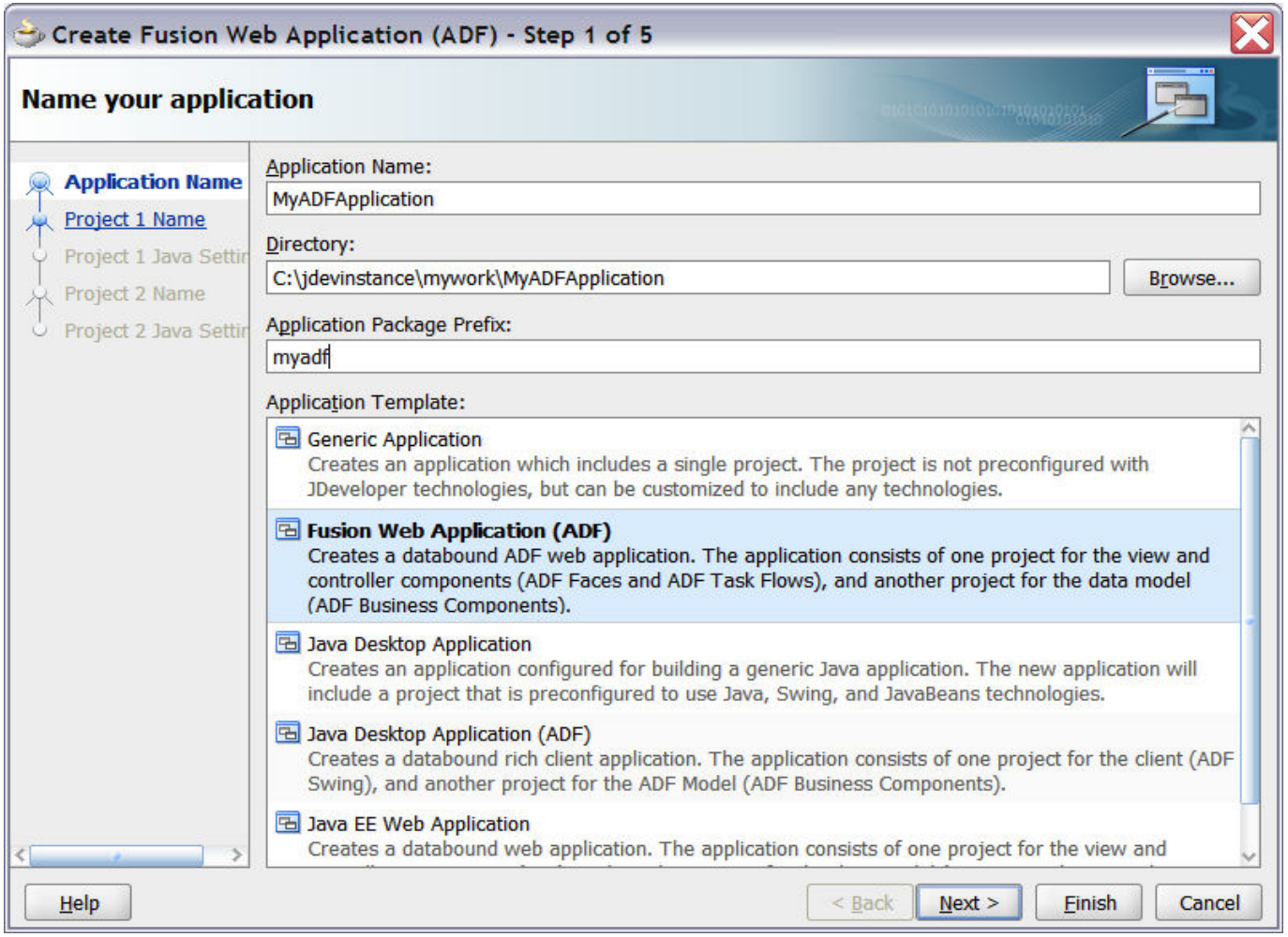
# New Application

- To create a new application use the JDeveloper menu's File->New->General->Applications option




 Copyright @ 2011, John Jay King

# New Gallery




 Copyright @ 2011, John Jay King

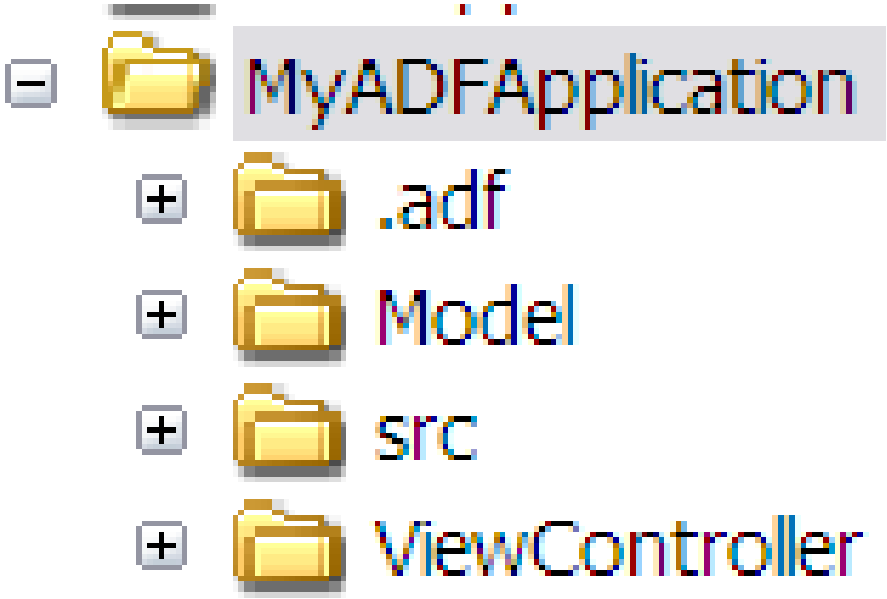
# Application Structure

- When a JDeveloper ADF Web Application is created ADF uses the MVC (Model-View Controller) pattern
- JDeveloper creates two subordinate projects
  - Model Data and Business Rules
  - ViewController User Interface
  - ADF provides the “Controller”

 Copyright @ 2011, John Jay King

# File Structure

- Review the directory structure created to support the application and the associated projects




 Copyright @ 2011, John Jay King

# How It Looks In JDeveloper

Oracle JDeveloper 11g - MyADFApplication.jws : ViewController.jpr

File Edit View Application Refactor Search Navigate Build Run Versioning Tools Window Help

Application Navigator

MyADFApplication

Projects

- Model
- ViewController
  - Web Content
    - WEB-INF
      - adfc-config.xml
      - faces-config.xml
      - trinidad-config.xml
      - web.xml
    - Page Flows

Application Resources

Data Controls

Recently Opened Files

ViewController.jpr - Structure

No Structure

Build Applications | Start Page | MyADFApplication.jws

Show: All Projects

File Summary: Total: 4 Error: 0 Warning: 0 Incomplete: 0 Advisory: 0 Ok: 4

Java Files — Getting Started — New

Overview: The Java Files category contains java classes and interfaces

Java Class

Java Interface

Cue Cards | Tutorials | Detailed Help

XML Files — Getting St

Overview: The XML Files category contai

XML File

Cue Cards | Tutorials | Detailed Help

Page Flows — Getting Started — New

Overview: Page Flows define an application's web pages and navigation.

JSF Page Flow

ADF Task Flow

ADF Task Flow Template

Struts Page Flow

Web Pages — Getting St

Overview: The Web Pages cate used to create the sc applications that use

JSF Page

JSP Page

HTML Page

JavaScript File

Style Sheet (CSS)

Enterprise JavaBeans 3.0 | ADF Binding Files | Web Services | Offline Databases

Overview

Feedback - Log

Messages | Feedback

Resource Palette

Name

My Catalogs

IDE Connections

- Application Server
- Database

C:\jdevinstance\mywork\MyADFApplication\ViewController\ViewController.jpr

Editing


Copyright @ 2011, John Jay King

# Create ADF BC Objects

- The following pages show how to create ADF BC objects using the Wizards provided by JDeveloper
- Each object created may be created individually using JDeveloper's features or by coding them manually rather than using the Wizards
- JDeveloper's database modeling capabilities are shown to good effect by the use of Database Connections and Wizards

 Copyright @ 2011, John Jay King

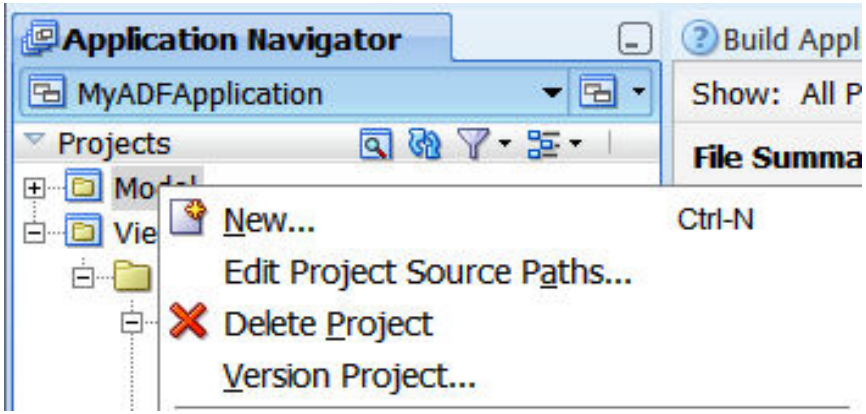
# Wizard-Based Development

- The “Create Business Objects from Tables” Wizard follows a few simple steps:
  - Create Business Component, select type of Business Component to be built
  - Select Database Connection to be used (may create Database Connection via Wizard)
  - Build Entity Objects using database Tables/Views
  - Build Updateable View Objects (if desired)
  - Build Read-Only View Objects (if desired)
  - Save Application Module

 Copyright @ 2011, John Jay King

# New ADF BC Object: 1

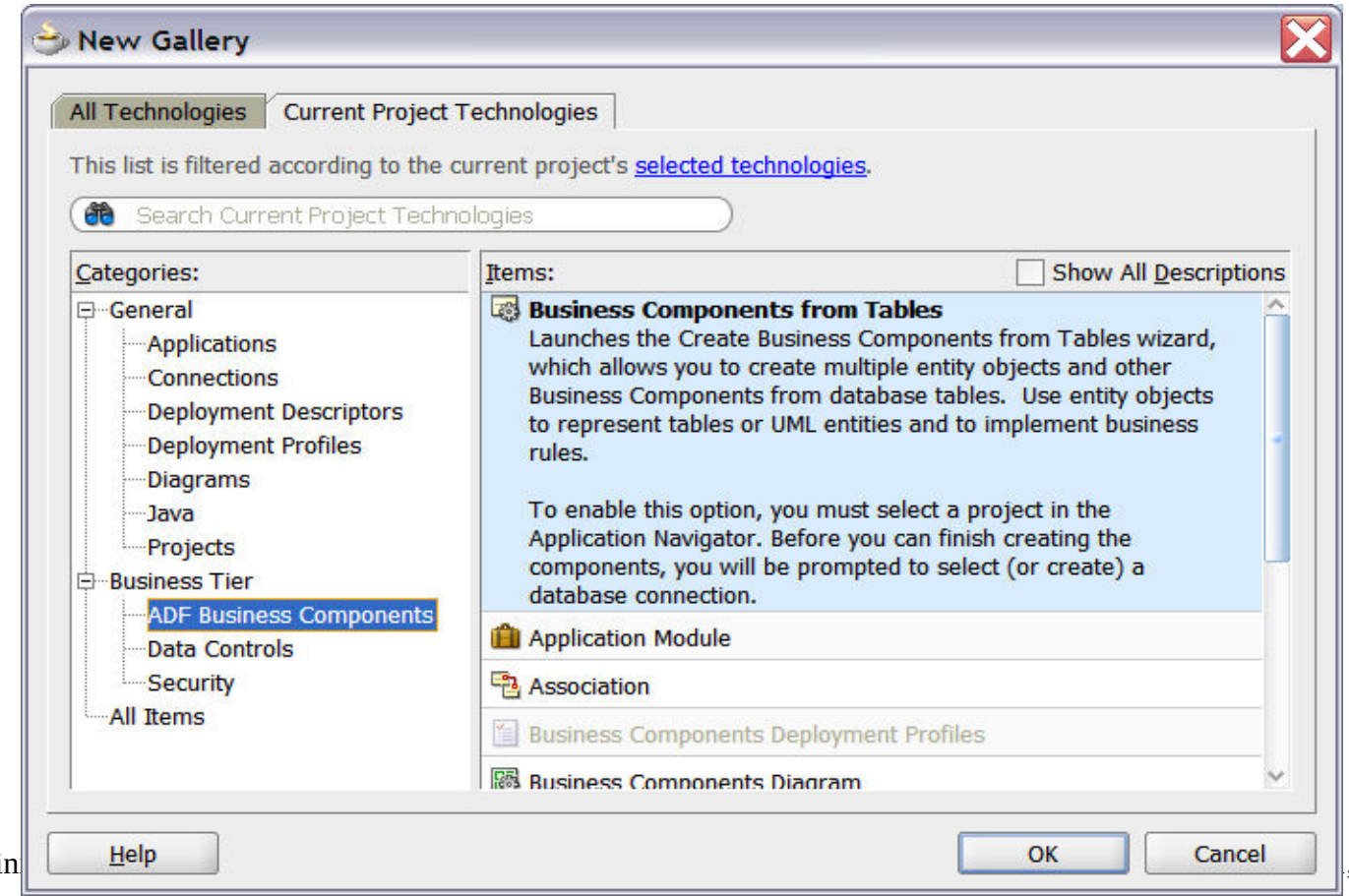
- Start building new components as follows:
- Right-click on the application's “Model” project and choose “New”




 Copyright @ 2011, John Jay King

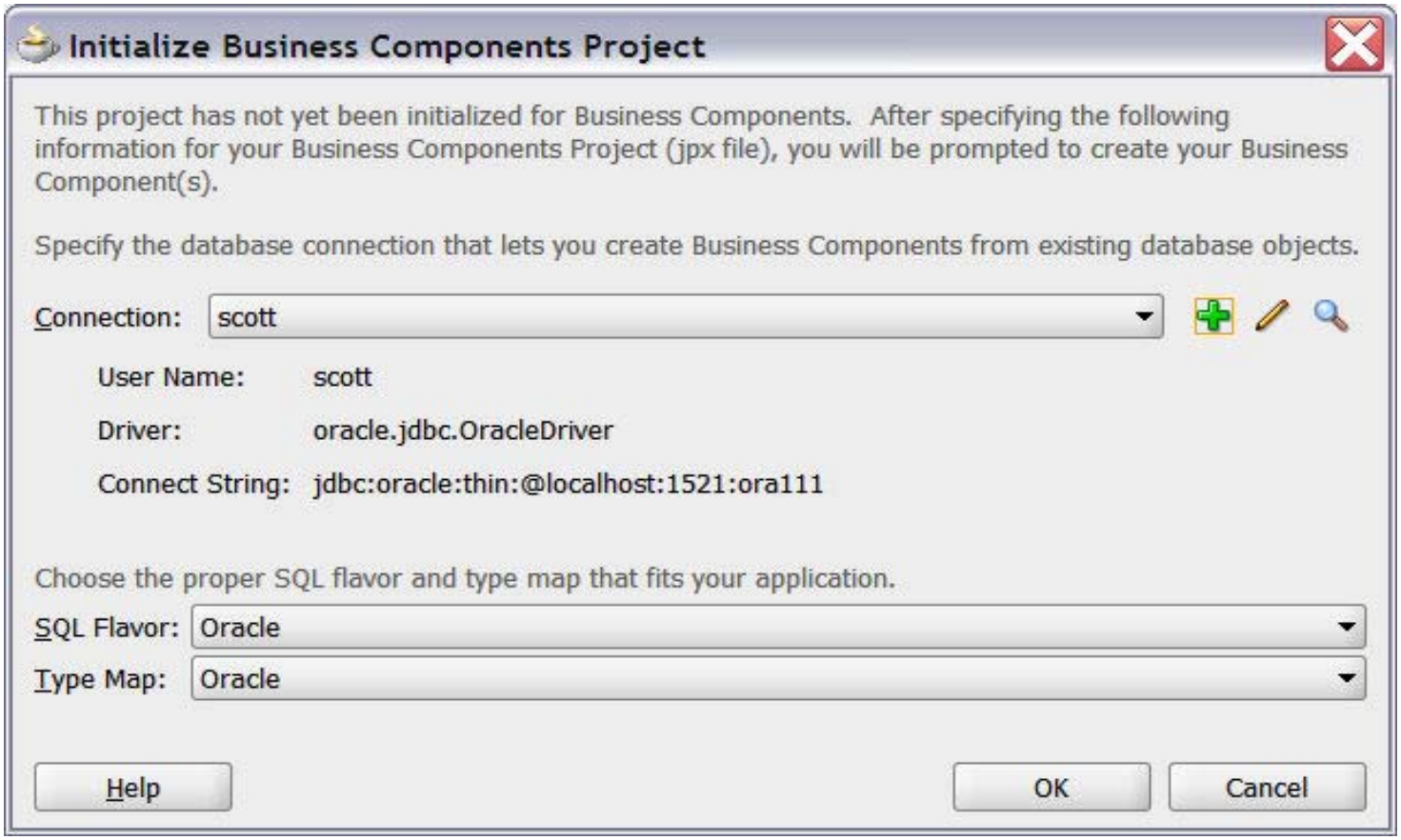
# New ADF BC Object: 2

- Choose **Business Tier -> ADF Business Components -> Business Components from Tables** from the “New Gallery”



# Choosing Database Connection

- Choose an existing Database Connection from the drop-down list or build a new one by clicking the green plus sign (Oracle client and tnsnames not required!)




 Copyright @ 2011, John Jay King

# Create Database Connection

**Create Database Connection**

Configure a new database connection and add it to the current application (MyADFApplication).

Create Connection In: (•) Application Resources ( ) IDE Connections

Connection Name: scott

Connection Type: Oracle (JDBC)

Username: scott

Role:

Password: •••••

[x] Save Password

- Oracle (JDBC) Settings

[ ] Enter Custom JDBC URL

Driver: thin

Host Name: localhost

JDBC Port: 1521

(•) SID: ora111

( ) Service Name: XE

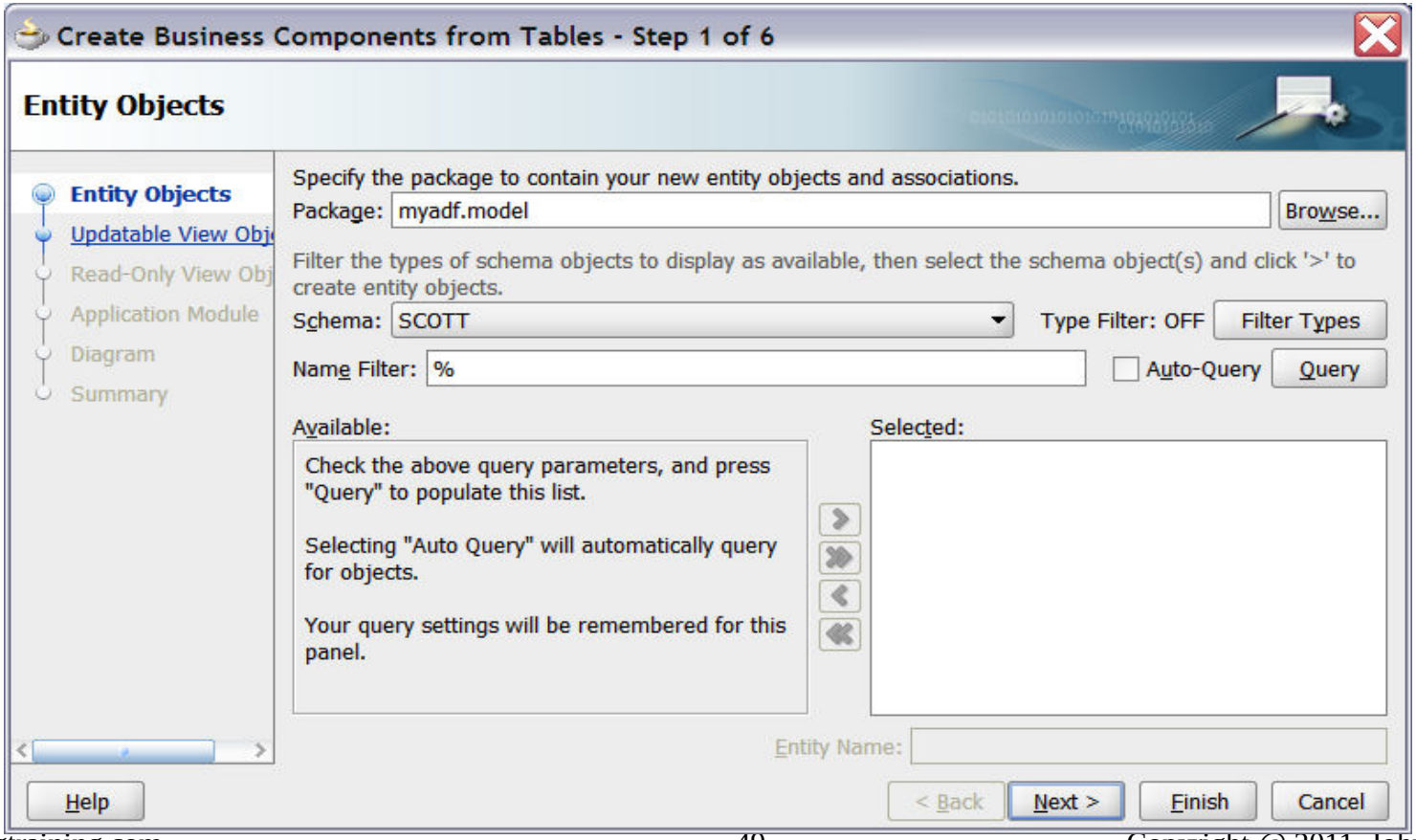
Test Connection

Help | OK | Cancel


Copyright @ 2011, John Jay King

# Create BC from Tables, 1

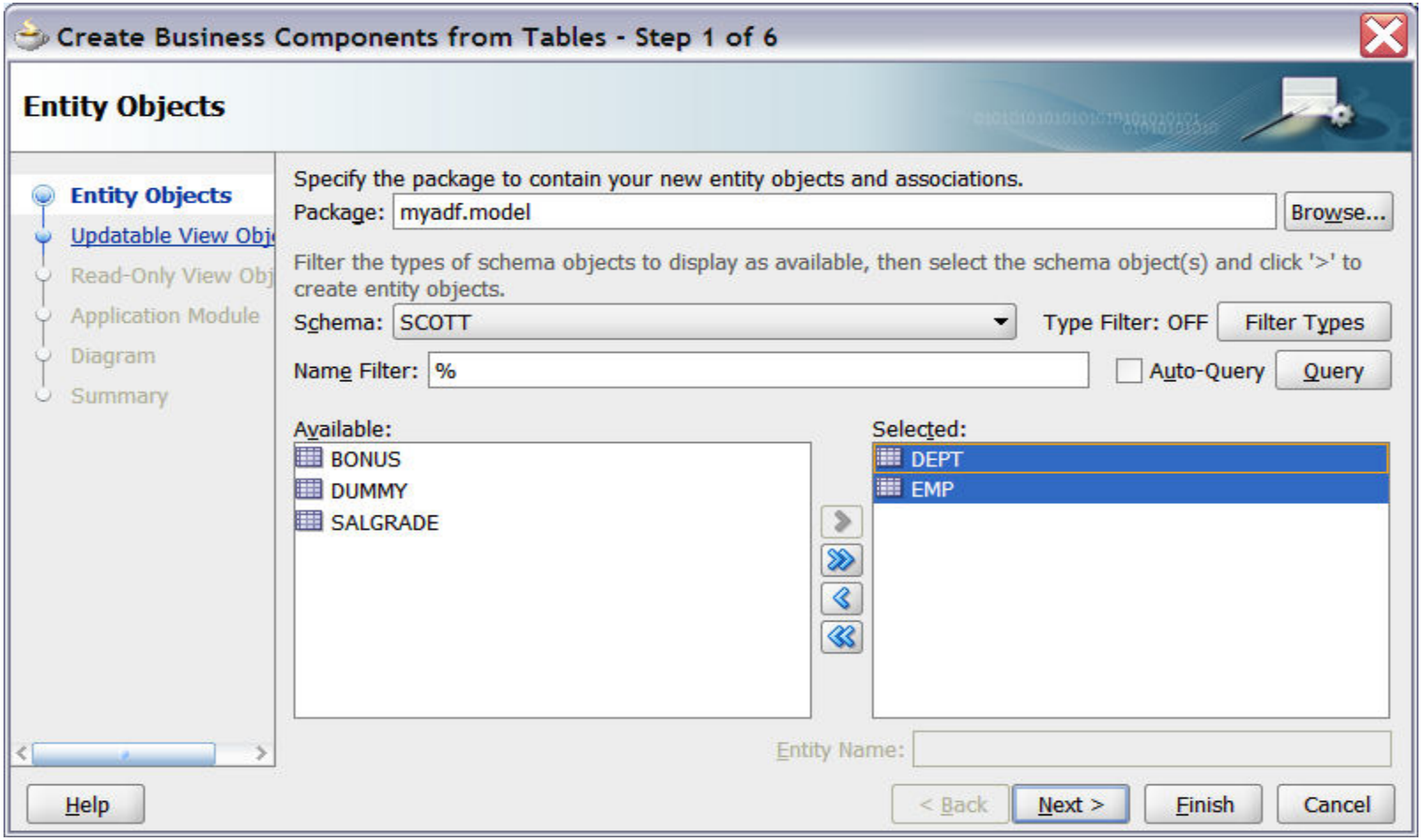
- Add, verify, or alter package name as desired; verify Schema to be used; modify filter (if desired) using SQL “LIKE” wild cards; click “Query” to view accessible database objects




 Copyright @ 2011, John Jay King

# Create BC from Tables, 2

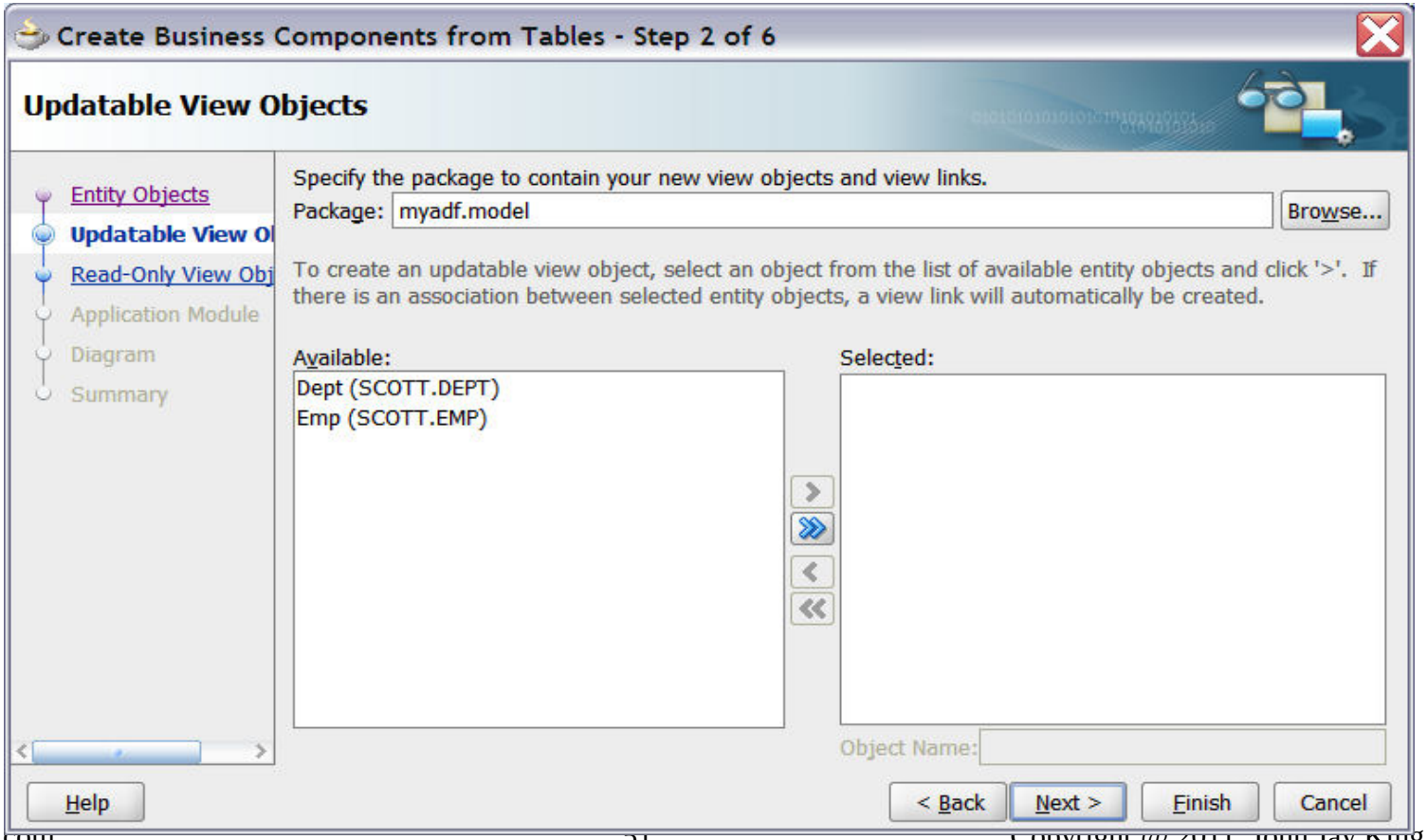
- Choose the tables and/or views to be part of the Entity Object and move them to the “Selected” side of the wizard




 Copyright @ 2011, John Jay King

# Create BC from Tables, 3

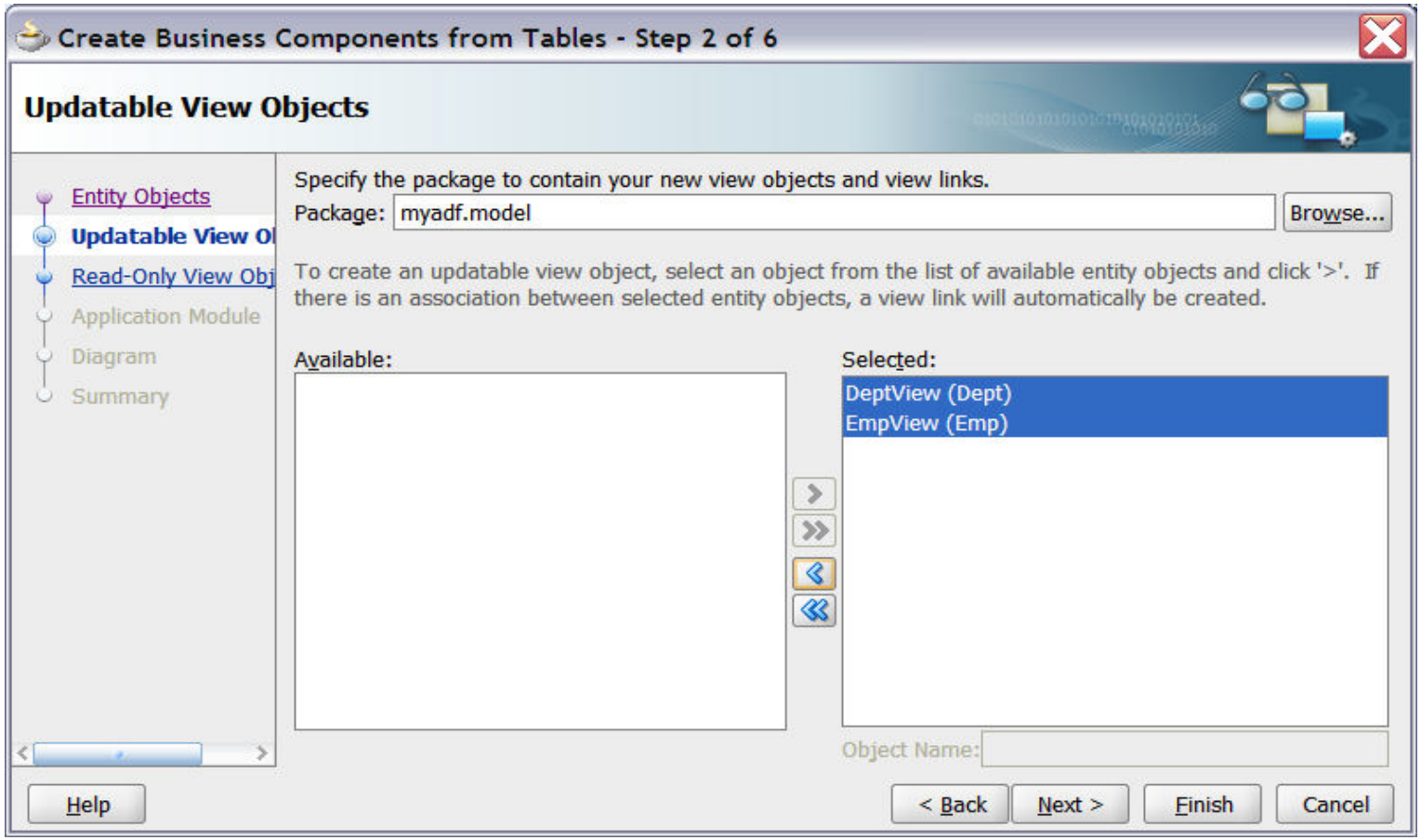
- After creating Entity Objects; the wizard offers to create Updateable View Objects -- View Objects represent the output of SQL (used to query, filter, join, modify, or sequence data)




 Copyright @ 2011, John Jay King

# Create BC from Tables, 4

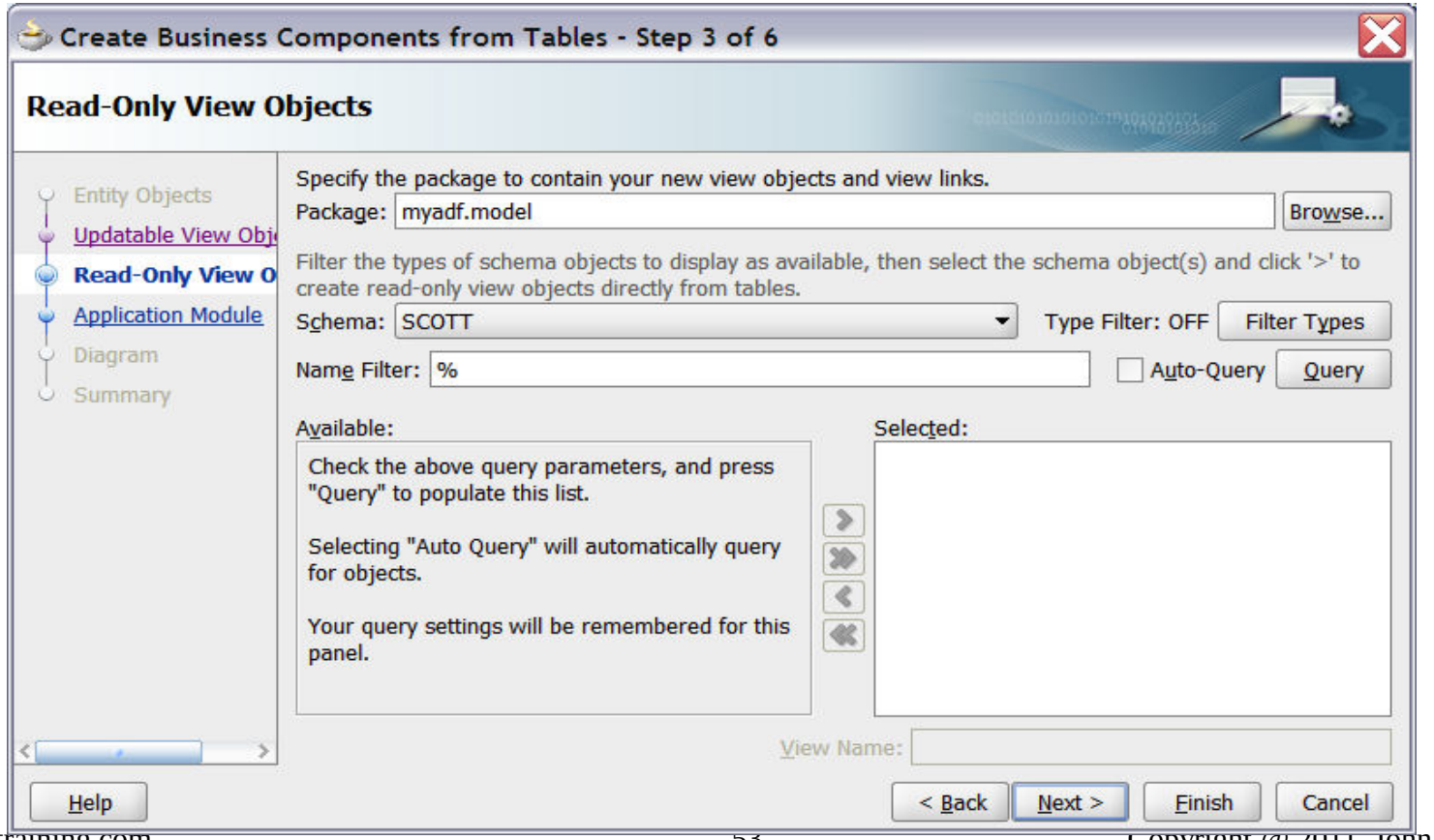
- Select Entity Objects to be used by the view being created; move them to the “Selected” side of panel




 Copyright @ 2011, John Jay King

# Create BC from Tables, 5

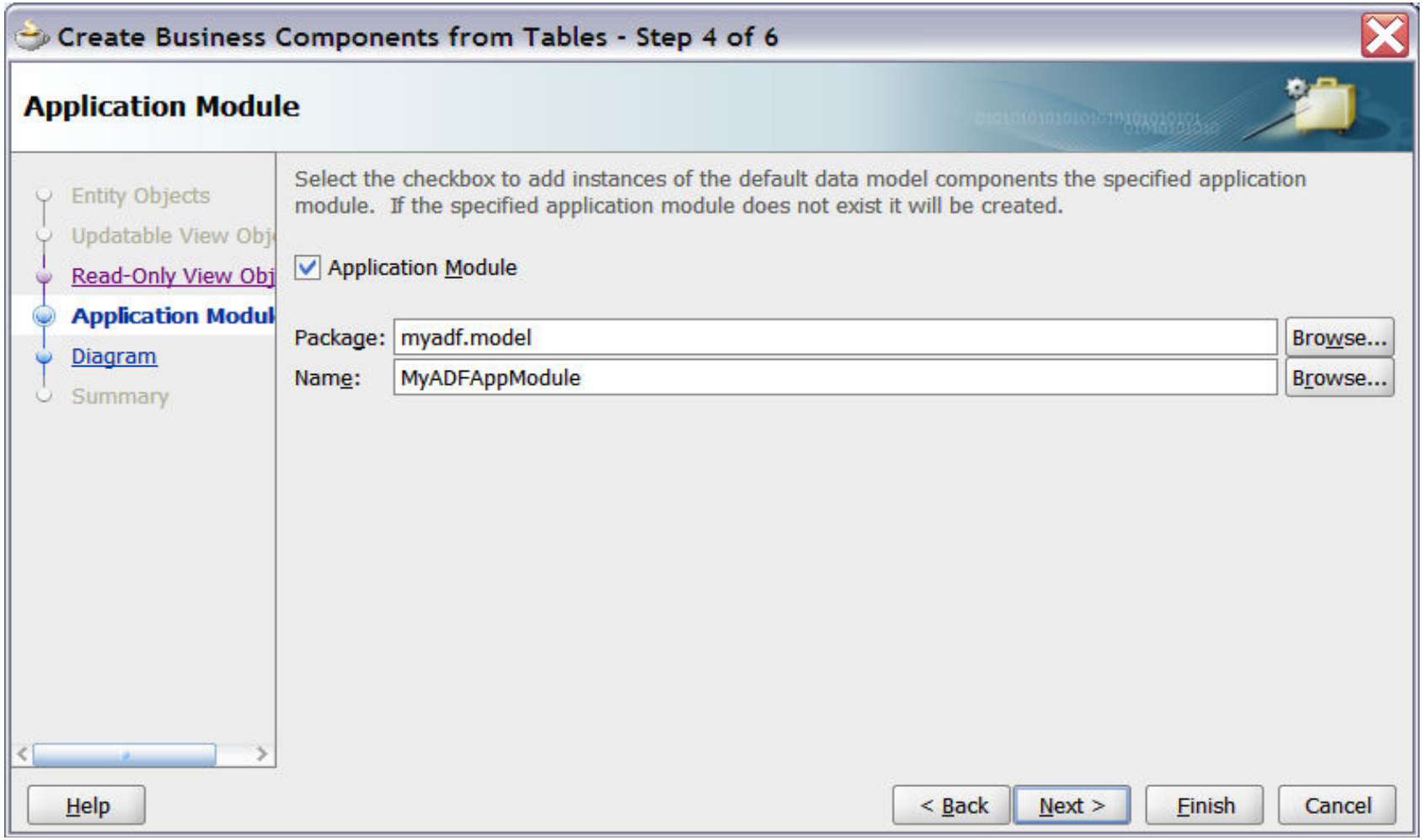
- After creating Updateable View Objects; the wizard goes on to create Read-Only View Objects (might be useful to support an LOV (List-of-Values))




 Copyright @ 2011, John Jay King

# Create BC from Tables, 6

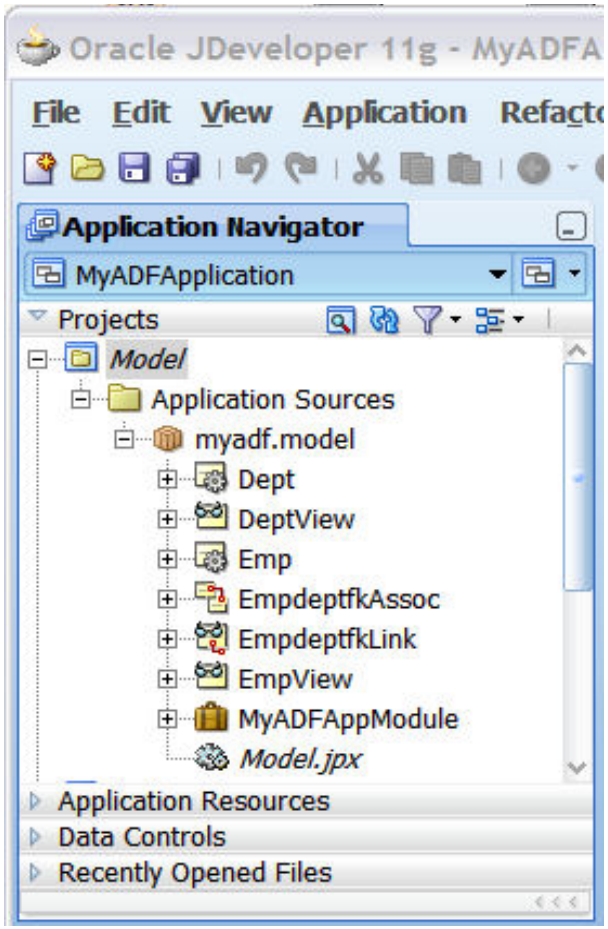
- Name the Application Module and click Finish

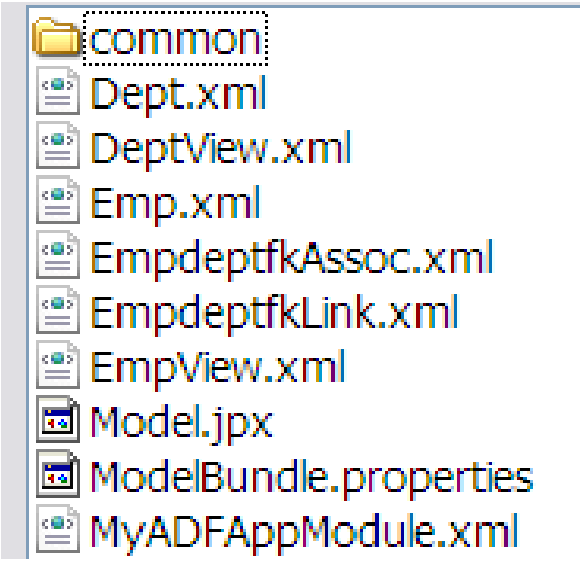



 Copyright @ 2011, John Jay King

# Business Component Files

- Note the use of XML to declaratively support ADF BC

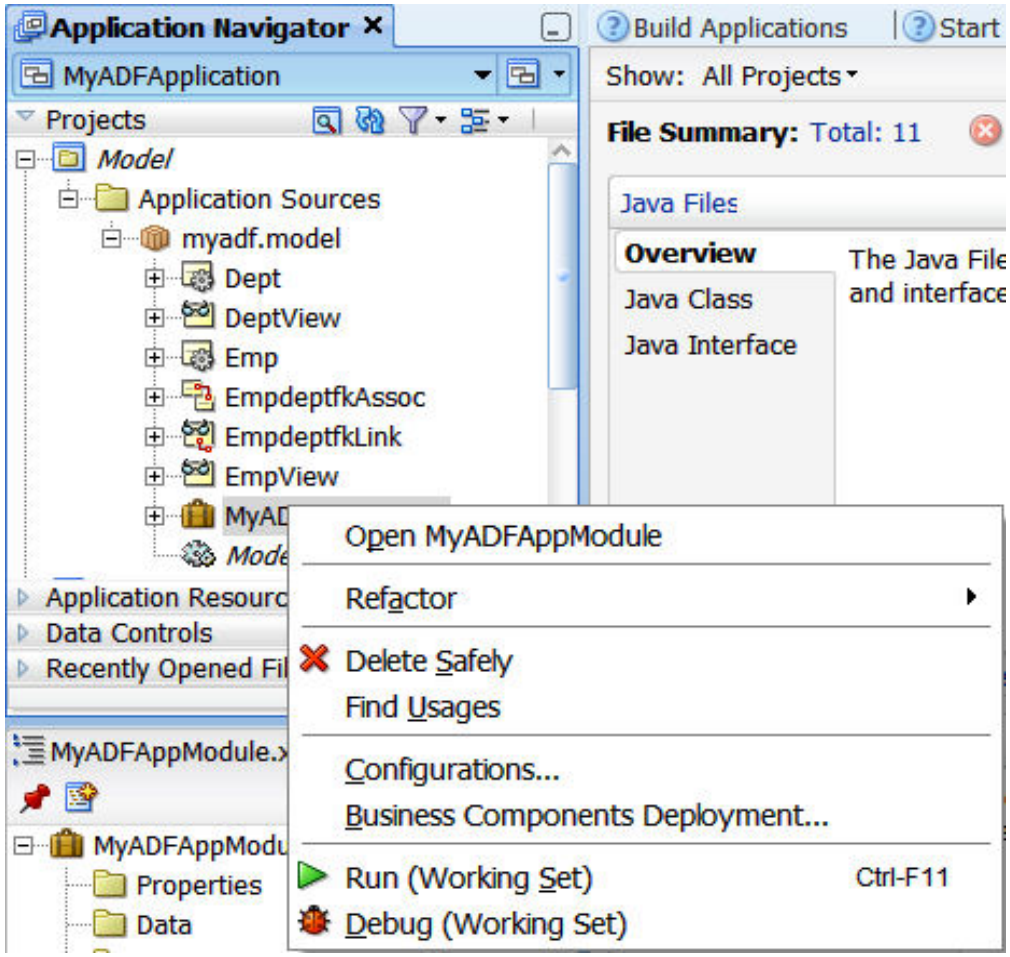



 Copyright @ 2011, John Jay King

# Business Component Browser

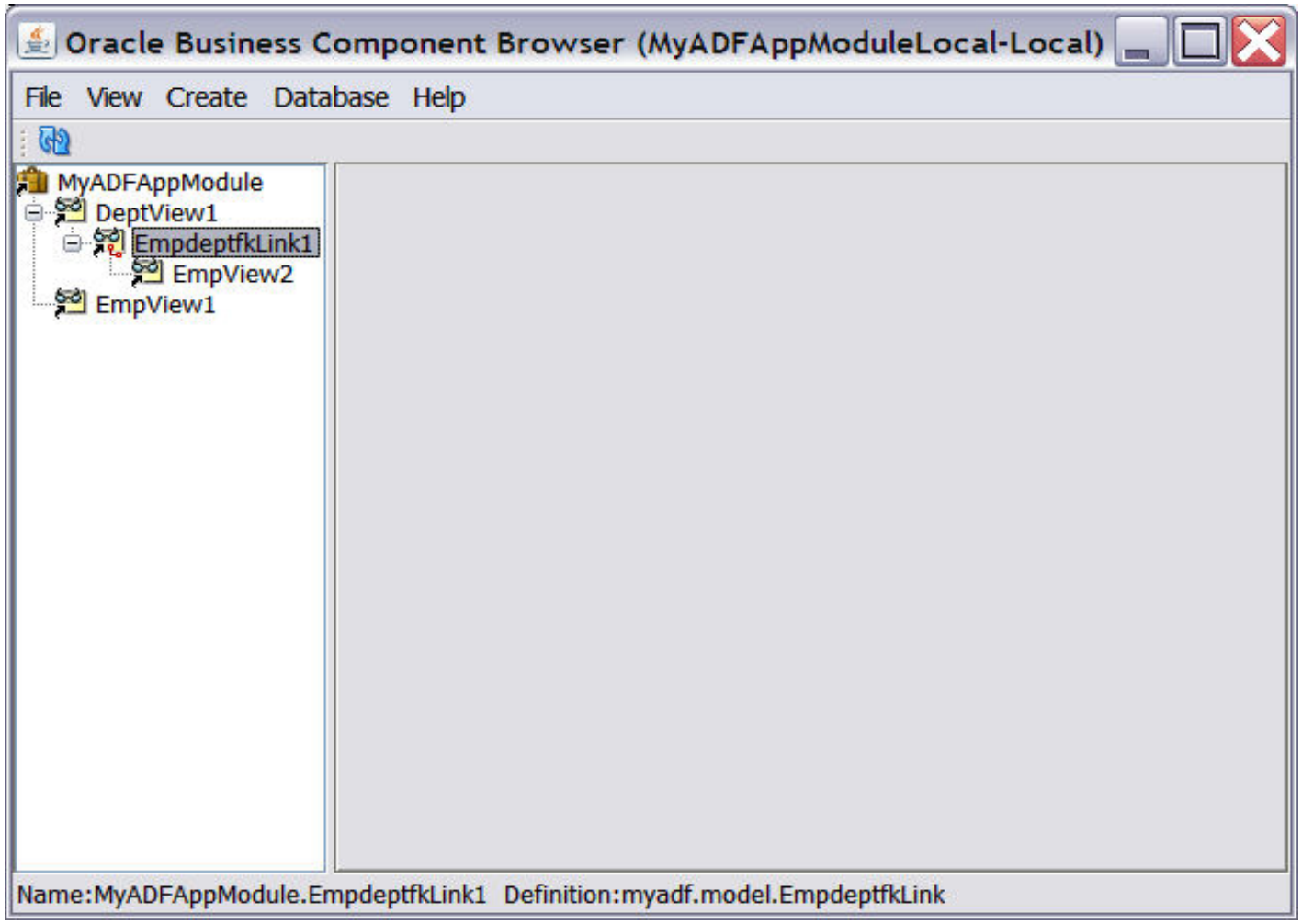
- JDeveloper provides a tool to “browse” ADF BC Application Module objects graphically; using the Application Navigator, find the Application Module to be viewed; right-click and choose “Run” to start




 Copyright @ 2011, John Jay King

# Component Browser Choices

- Choose the Business Component to be tested




 Copyright @ 2011, John Jay King

# Component Browser – Display, 1

- Oracle's Business Component Browser displays data from the underlying database objects (screen should look familiar to Oracle Forms users)
- If referential keys are defined in the database (Primary Keys and Foreign Keys) the ADF BC Wizard automatically arranges the tables into a Master-Detail relationship

 Copyright @ 2011, John Jay King

# Component Browser – Display, 2

Oracle Business Component Browser (MyADFAppModuleLocal-Local)

File View Create Database Help

- MyADFAppModule
  - DeptView1
    - EmpdeptfkLink1
      - EmpView2
  - EmpView1

EmpdeptfkLink1

| | |
|---|---|
| **Deptno** | 10 |
| Dname | ACCOUNTING |
| Loc | NEW YORK |

| Empno | Ename | Job | Mgr | Hired... | Sal | Comm | Deptno |
|---|---|---|---|---|---|---|---|
| 7782 | CLARK | MANAG... | 7839 | 1981-0... | 2450 | | 10 |
| 7839 | KING | PRESID... | | 1981-1... | 5000 | | 10 |
| 7934 | MILLER | CLERK | 7782 | 1982-0... | 1300 | | 10 |

Name:MyADFAppModule.EmpdeptfkLink1 Definition:myadf.model.EmpdeptfkLink


Copyright @ 2011, John Jay King

# Searching Data

- Use the “Specify View Criteria” (Binocular) icon to Search

Oracle Business Component Browser (MyADFAppModuleLocal-Local)

File View Create Database Help

MyADFAppModule
- DeptView1
  - EmpdeptfkLink1
    - EmpView2
- EmpView1

EmpdeptfkLink1

Specify View Criteria

| | |
|---|---|
| **Deptno** | 20 |
| Dname | RESEARCH |
| Loc | DALLAS |

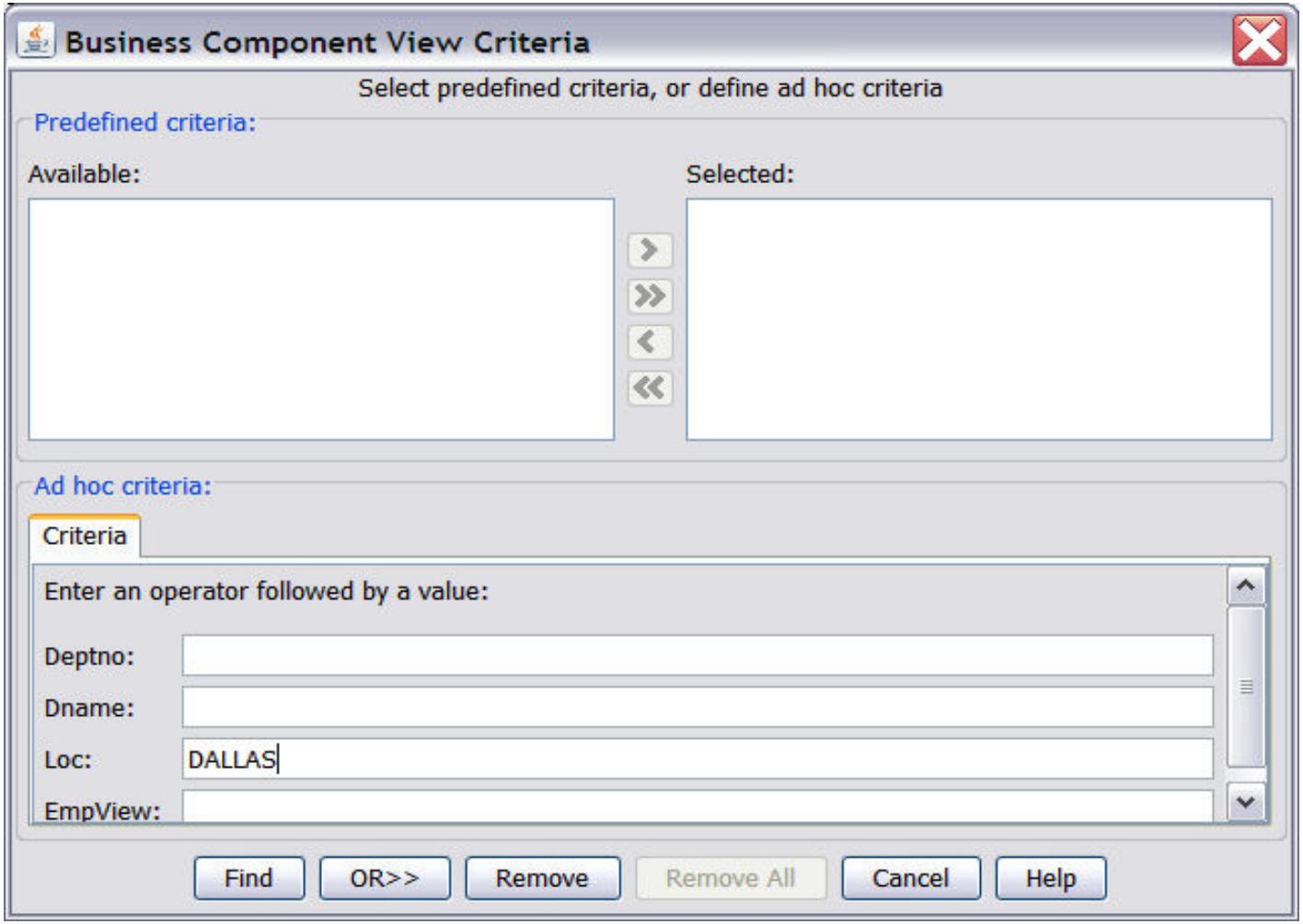
| Empno | Ename | Job | Mgr | Hired... | Sal | Comm | Deptno |
|---|---|---|---|---|---|---|---|
| 7369 | SMITH | CLERK | 7902 | 1980-1... | 800 | | 20 |
| 7566 | JONES | MANAG... | 7839 | 1981-0... | 2975 | | 20 |
| 7788 | SCOTT | ANALYST | 7566 | 1982-1... | 3000 | | 20 |
| 7876 | ADAMS | CLERK | 7788 | 1983-0... | 1100 | | 20 |
| 7902 | FORD | ANALYST | 7566 | 1981-1... | 3000 | | 20 |

Name:MyADFAppModule.EmpdeptfkLink1 Definition:myadf.model.EmpdeptfkLink


Copyright @ 2011, John Jay King

# Search View Criteria

- Enter Search criteria and click “Find”




 Copyright @ 2011, John Jay King

# Search Results

Oracle Business Component Browser (MyADFAppModuleLocal-Local)

File View Create Database Help

- MyADFAppModule
  - DeptView1
    - EmpdeptfkLink1
      - EmpView2
  - EmpView1

EmpdeptfkLink1

| | |
|---|---|
| **Deptno** | 20 |
| Dname | RESEARCH |
| Loc | DALLAS |

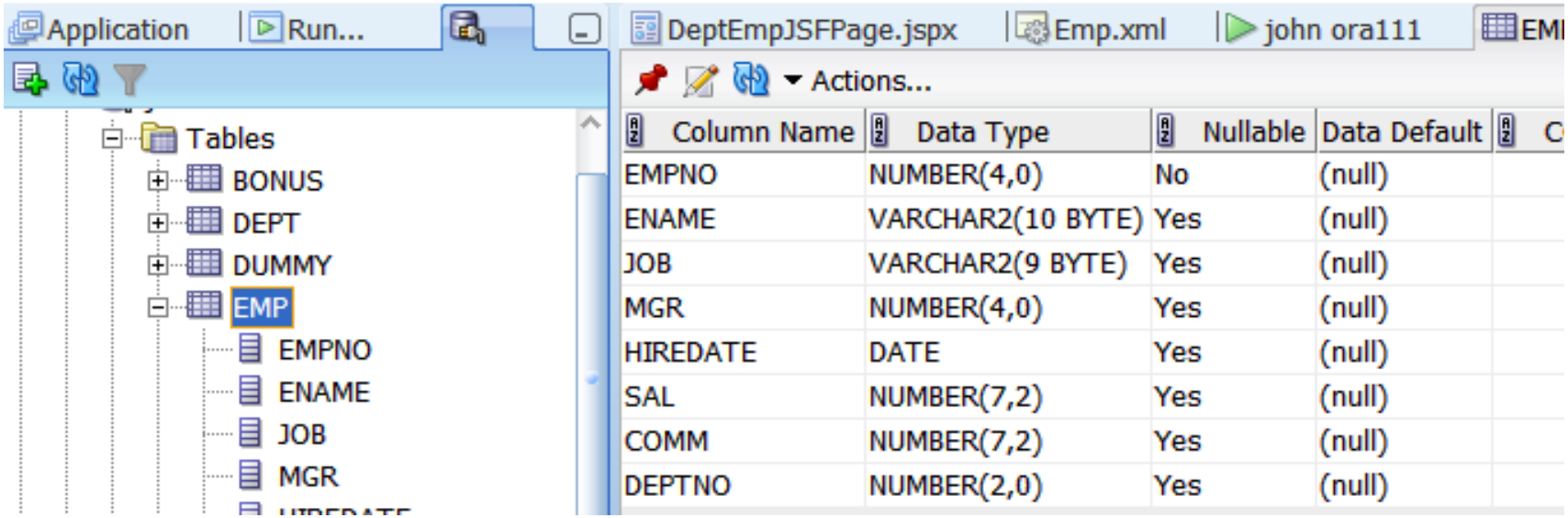
| Empno | Ename | Job | Mgr | Hired... | Sal | Comm | Deptno |
|---|---|---|---|---|---|---|---|
| 7369 | SMITH | CLERK | 7902 | 1980-1... | 800 | | 20 |
| 7566 | JONES | MANAG... | 7839 | 1981-0... | 2975 | | 20 |
| 7788 | SCOTT | ANALYST | 7566 | 1982-1... | 3000 | | 20 |
| 7876 | ADAMS | CLERK | 7788 | 1983-0... | 1100 | | 20 |
| 7902 | FORD | ANALYST | 7566 | 1981-1... | 3000 | | 20 |

Name:MyADFAppModule.EmpdeptfkLink1 Definition:myadf.model.EmpdeptfkLink

 Copyright @ 2011, John Jay King

# Browsing Database Objects

- JDeveloper's Database Navigator allows browsing of database objects (parts of Oracle's SQL Developer tool have been incorporated into JDeveloper)




 Copyright @ 2011, John Jay King

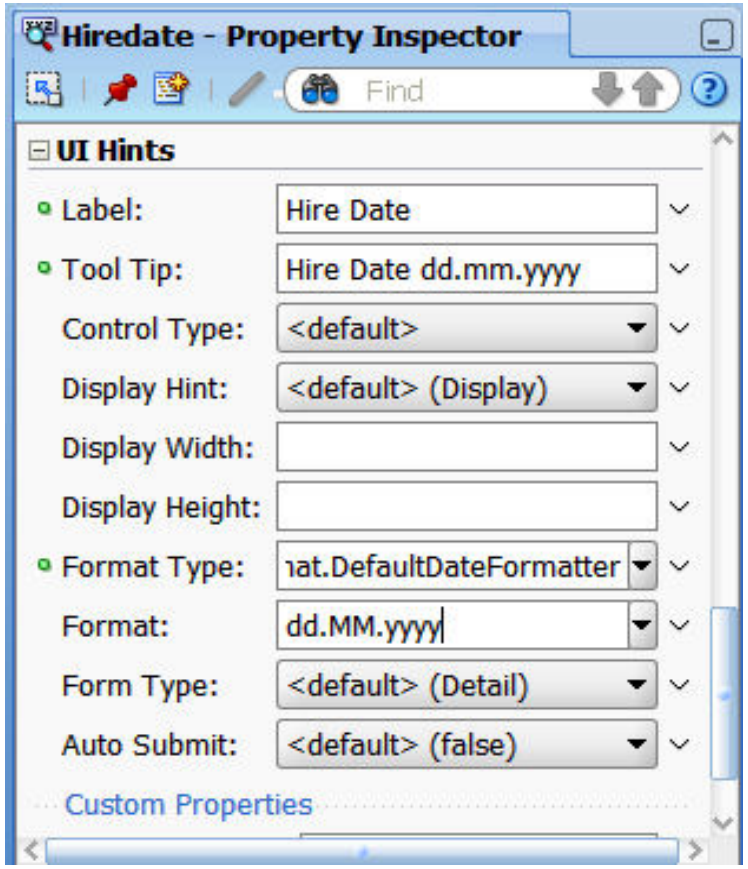
# Modification of Application

- Once the initial Business Components are created in the application, it might be useful to:
  - Set default values
  - Define formatting
  - Validate data

 Copyright @ 2011, John Jay King

# Object Properties

- Like Oracle Forms (and other 4GLs) properties are listed




 Copyright @ 2011, John Jay King

# Properties in XML Files

- ADF uses XML files to store declared definitions

Build Applications | Start Page | MyADFApplication.jws | Emp.xml

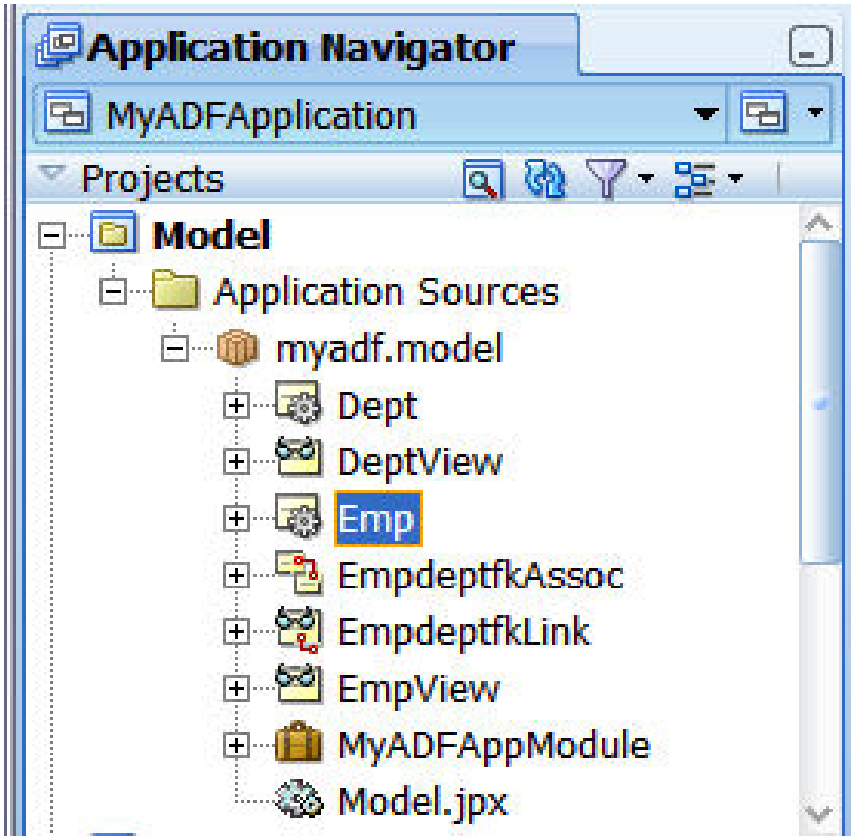
```
70  <Attribute
71    Name="Hiredate"
72    ColumnName="HIREDATE"
73    SQLType="TIMESTAMP"
74    Type="oracle.jbo.domain.Date"
75    ColumnType="DATE"
76    TableName="EMP">
77    <TransientExpression><![CDATA[adf.currentDate]]></TransientExpression>
78    <DesignTime>
79      <Attr Name="_DisplaySize" Value="7"/>
80    </DesignTime>
81    <Properties>
82      <SchemaBasedProperties>
83        <LABEL
84          ResId="myadf.model.Emp.Hiredate_LABEL"/>
85        <TOOLTIP
86          ResId="myadf.model.Emp.Hiredate_TOOLTIP"/>
87        <FMT_FORMATTER
88          ResId="myadf.model.Emp.Hiredate_FMT_FORMATTER"/>
89        <FMT_FORMAT
90          ResId="myadf.model.Emp.Hiredate_FMT_FORMAT"/>
91      </SchemaBasedProperties>
92    </Properties>
93  </Attribute>
94  <Attribute
95    Name="Sal"
```

Overview | Source | History

 Copyright @ 2011, John Jay King

# Modify Appearance and Formatting

- Use JDeveloper to modify appearance of database column values by double-clicking an Entity Object




 Copyright @ 2011, John Jay King

# Entity Object Edit Panel

Oracle JDeveloper 11g - MyADFApplication.jws : Model.jpr : C:\jdevinstance\mywork\MyADFApplication\Model\src\myadf\model\Emp.xml

File Edit View Application Refactor Search Navigate Build Run Versioning Tools Window Help

Application Navigator

MyADFApplication

Projects

- Model
  - Application Sources
    - myadf.model
      - Dept
      - DeptView
      - Emp
      - EmpdeptfkAssoc
      - EmpdeptfkLink
      - EmpView
      - MyADFAppModule
      - Model.jpx
- Application Resources
- Data Controls
- Recently Opened Files

Emp.xml - Structure

- Emp
  - Properties
  - Data
  - PublishEvent
  - Publisher
  - Rules
  - Attributes
  - Association Accessors
  - CustomAttribute
  - AssociationAccessor
  - ViewLink Accessors
  - Bind Variables
  - Constraints
  - PropertySets
  - Permission

Build Applications | Start Page | MyADFApplication.jws | Emp.xml

General
Attributes
Validators
Java
Business Events
View Accessors

**Entity Object: Emp**

Entity objects are used to encapsulate the business logic and database storage details for your business entities.

| | |
|---|---|
| Name: | Emp |
| Package: | myadf.model |
| Extends: | <None> |
| Property Set: | <None> |
| Schema Object: | EMP |

Business Components Project: Model

**Alternate Keys**

**Tuning**

**Custom Properties**

**Security: Emp**

**Business Logic Groups**

Overview | Source | History

Running: Model.jpr - Log

```
Diagnostics: (Properties (re)loaded) Routing diagnostics to standard output (use -D
Oct 21, 2009 10:43:05 AM oracle.jbo.jbotester.MainFrame main
INFO: BC4J Tester started.
Oct 21, 2009 10:55:50 AM oracle.jbo.jbotester.MainFrame exit
INFO: BC4J Tester exit code(0)
Process exited with exit code 0.
```

Resource Palette

- My Catalogs
- IDE Connections
  - Application Server
  - Database

Emp - Property Inspector

**Type**

| | |
|---|---|
| Name *: | Emp |
| PropertySet: | |
| Schema Object Type: | table |
| Schema Object: | EMP |
| Library Private: | <default... |
| Effective Date Type: | <default... |
| Updatable: | <default... |

**Business Logic Group**

**Tuning**

start | Oracle JDevelop... | Corel Paint Sho... | Norton™ | 11:0


Copyright @ 2011, John Jay King

# Entity Object Attributes

Build Applications | Start Page | MyADFApplication.jws | **Emp.xml**

- General
- **Attributes**
- Validators
- Java
- Business Events
- View Accessors

**Attributes**

Add from Table... Override

Entity attributes can be based upon columns in the schema object or can be based upon transient values.

| Name | Type | Column | Column Type | Extends |
|---|---|---|---|---|
| Empno | Number | EMPNO | NUMBER(4, 0) | |
| Ename | String | ENAME | VARCHAR2(10) | |
| Job | String | JOB | VARCHAR2(9) | |
| Mgr | Number | MGR | NUMBER(4, 0) | |
| Hiredate | Date | HIREDATE | DATE | |
| Sal | Number | SAL | NUMBER(7, 2) | |
| Comm | Number | COMM | NUMBER(7, 2) | |
| Deptno | Number | DEPTNO | NUMBER(2, 0) | |

Validation Rules: Empno

Custom Properties: Empno

Security: Empno

Overview | Source | History


Copyright @ 2011, John Jay King

# Entity Object Validators

Build Applications | Start Page | MyADFApplication.jws | **Emp.xml**

- General
- Attributes
- **Validators**
- Java
- Business Events
- View Accessors

**Validators**

Select the entity object or one of its attributes and click the New button to apply a new validation rule.

- Emp
  - Attributes
    - Empno
    - Ename
    - Job
    - Mgr
    - Hiredate
    - Sal
    - Comm
    - Deptno
  - Entity

Overview | Source | History

 Copyright @ 2011, John Jay King

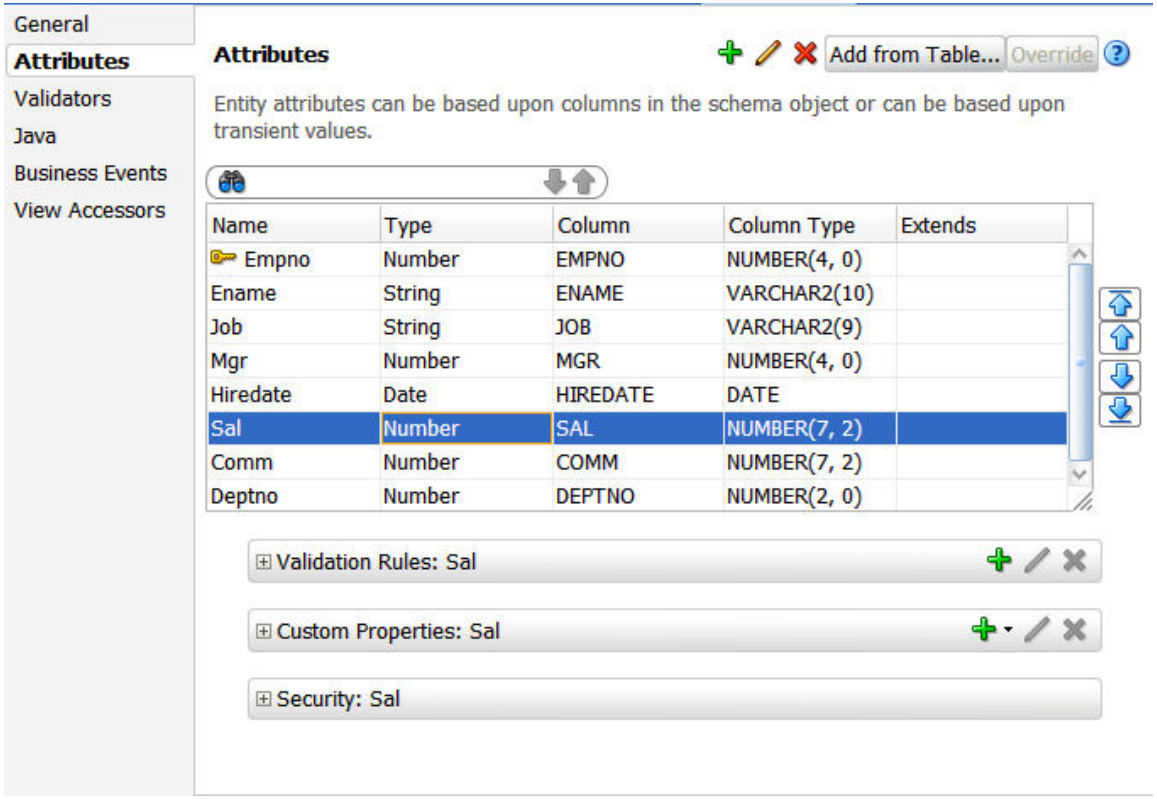
# Validations and Business Logic

- Validations and Business Logic may be added including:
  - Client-side validation
  - Format masks
  - Default Values
  - Declarative Range (and other) Validation
  - CSS (Visual Attributes)
  - List of Values
  - Calculated field
  - Code Validation
  - Extensible for complex application validation
  - Transactional Triggers

 Copyright @ 2011, John Jay King

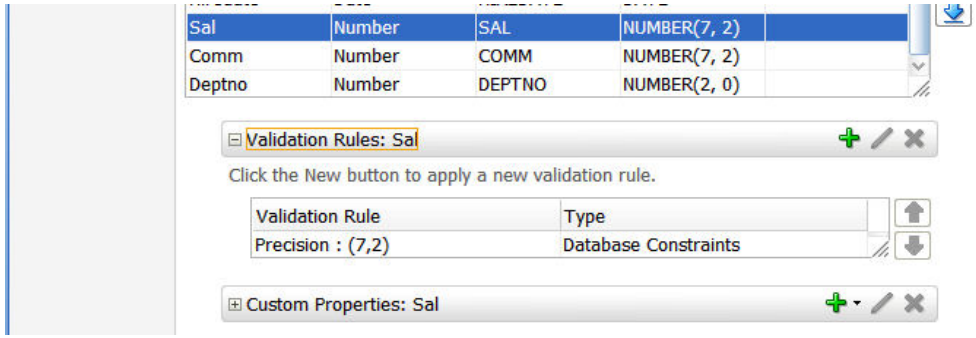
# ADF BC Attributes

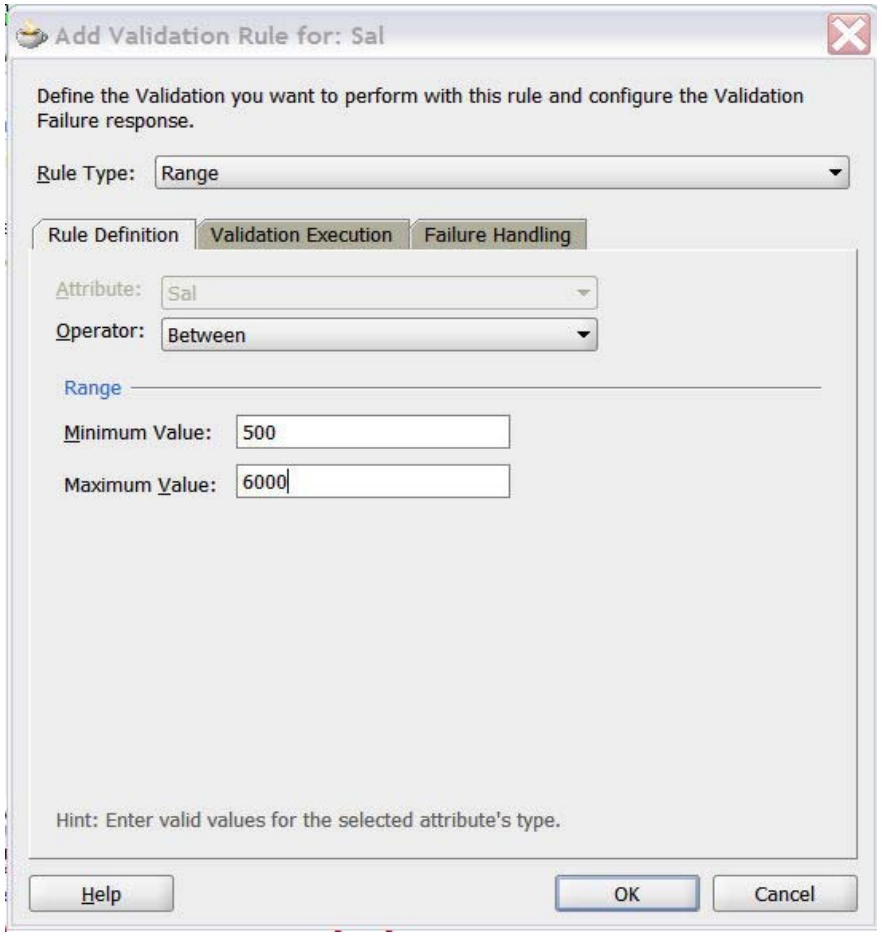
- To view/modify properties for an attribute; highlight the attribute on the Attributes panel and use the Property Palette or one of the property accordions




 Copyright @ 2011, John Jay King

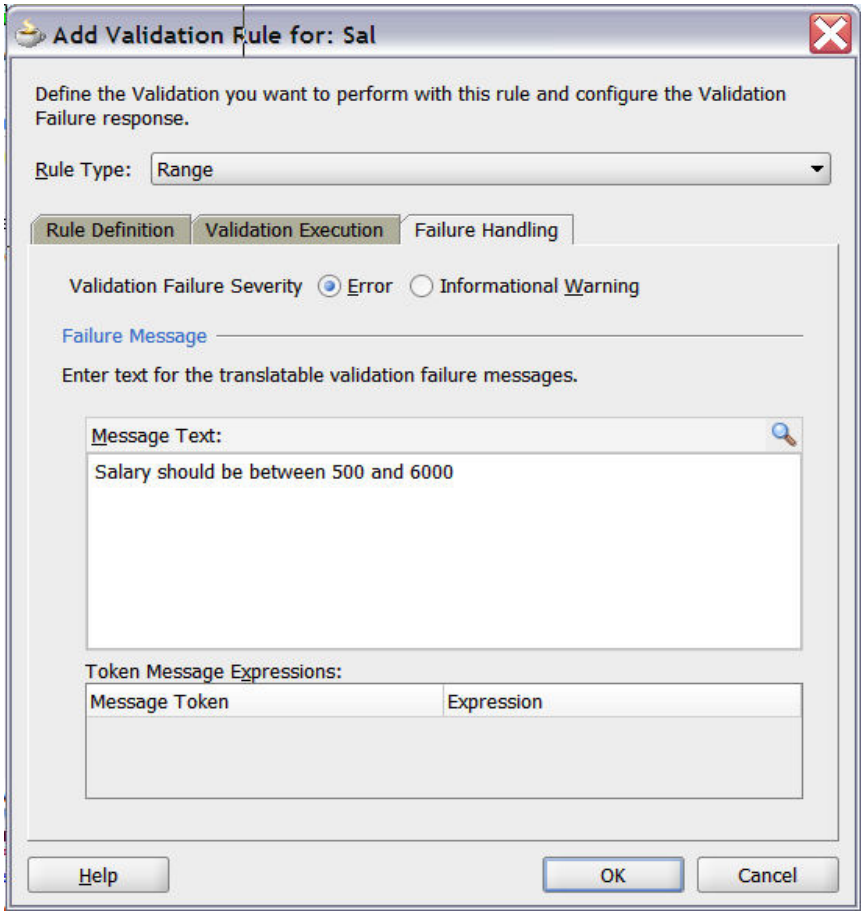
# Validation Rules




 Copyright @ 2011, John Jay King

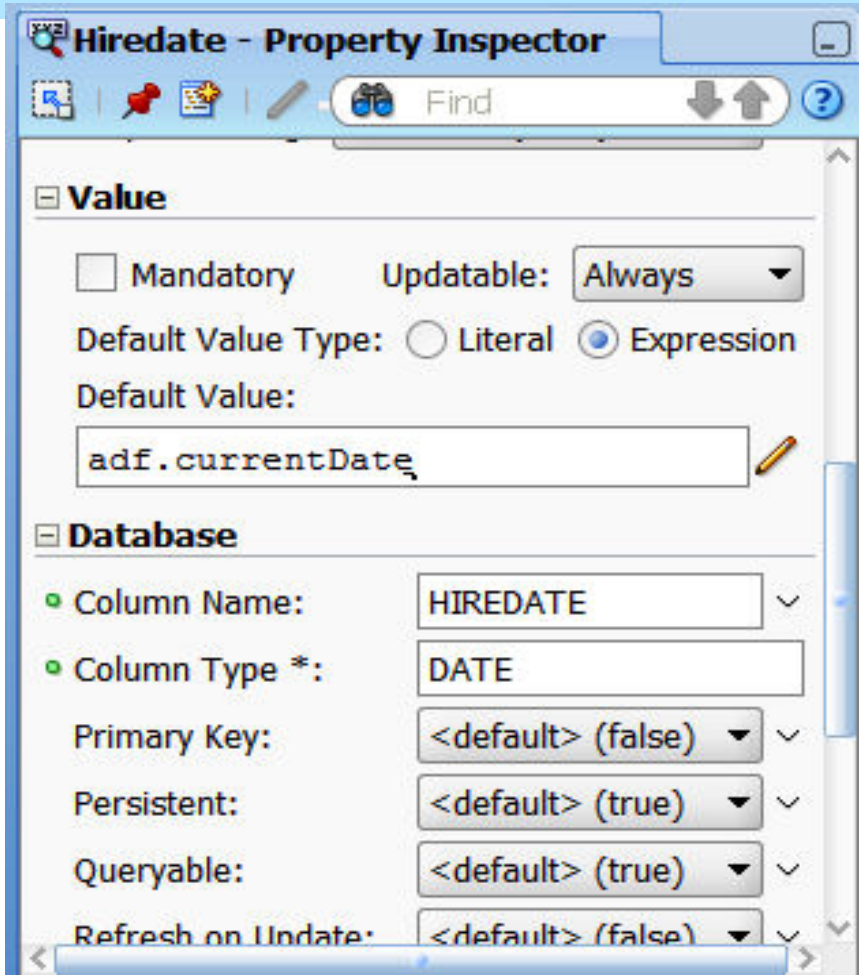
# Validation Error Messages




 Copyright @ 2011, John Jay King

# Attribute Defaults

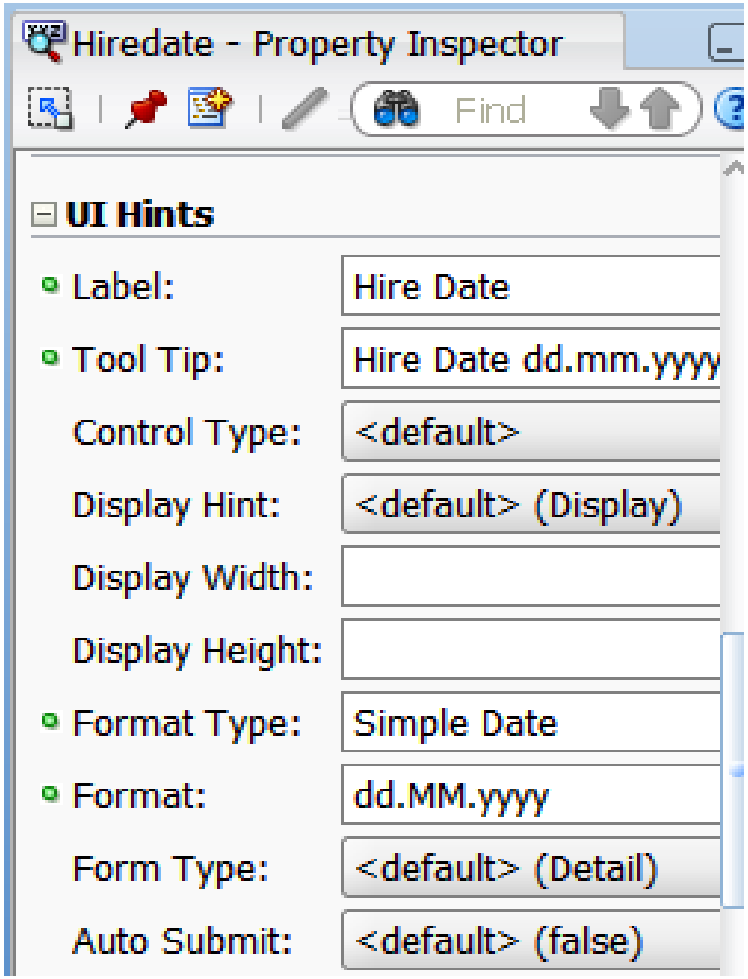
- Using the Property Palette, open the “Value” properties and set the default value (in this case “adf.currentDate” using ADF’s “Groovy” support)




 Copyright @ 2011, John Jay King

# Attribute Formatting

- Use an Attribute's Property Palette “UI Hints” section to control formatting, label, tool tip, etc… (note this formatting uses Java SimpleDateFormat options)




 Copyright @ 2011, John Jay King

# What Does the XML Look Like?

Build Applications | Start Page | MyADFApplication.jws | Emp.xml

Find

```
70  <Attribute
71    Name="Hiredate"
72    ColumnName="HIREDATE"
73    SQLType="TIMESTAMP"
74    Type="oracle.jbo.domain.Date"
75    ColumnType="DATE"
76    TableName="EMP">
77    <TransientExpression><![CDATA[adf.currentDate]]></TransientExpression>
78    <DesignTime>
79      <Attr Name="_DisplaySize" Value="7"/>
80    </DesignTime>
81    <Properties>
82      <SchemaBasedProperties>
83        <LABEL
84          ResId="myadf.model.Emp.Hiredate_LABEL"/>
85        <TOOLTIP
86          ResId="myadf.model.Emp.Hiredate_TOOLTIP"/>
87        <FMT_FORMATTER
88          ResId="myadf.model.Emp.Hiredate_FMT_FORMATTER"/>
89        <FMT_FORMAT
90          ResId="myadf.model.Emp.Hiredate_FMT_FORMAT"/>
91      </SchemaBasedProperties>
92    </Properties>
93  </Attribute>
94  <Attribute
95    Name="Sal"
```

Overview | Source | History


Copyright @ 2011, John Jay King

# Date Mask Properties File

ModelBundle.properties

```
#
myadf.model.Emp.Sal_Rule_0=Salary should be between 500 and 6000
myadf.model.Emp.Hiredate_LABEL=Hire Date
myadf.model.Emp.Hiredate_TOOLTIP=Hire Date dd.mm.yyyy
myadf.model.Emp.Hiredate_FMT_FORMATTER=oracle.jbo.format.DefaultDateFormat
myadf.model.Emp.Hiredate_FMT_FORMAT=dd.MM.yyyy
```

 Copyright @ 2011, John Jay King

# Comparison to Oracle Forms

- In Oracle Forms we defined “data blocks” that represented tables and views that would be used in our forms
- ADF BC components do that and more, plus they may be shared by many applications
- In Oracle Forms once the “data block” was created we would then use it to create the presentation
- With ADF we use ADF Faces to accomplish the same thing and more
  (again creating components that may be reused by other applications)

 Copyright @ 2011, John Jay King

# Comparison to Typical 4GLs

- Most 4GLs offer some type of “Data Object” or “Data Access Object” capability
  - Usually include wizard-based development
  - Usually work with relational database; do not usually support procedure-based data
  - Sometimes provide ability to find and link data objects using database dictionary
  - Sometimes provide stand-alone reusable data objects
  - Sometimes linked to GUI development via “drag-and-drop” capability

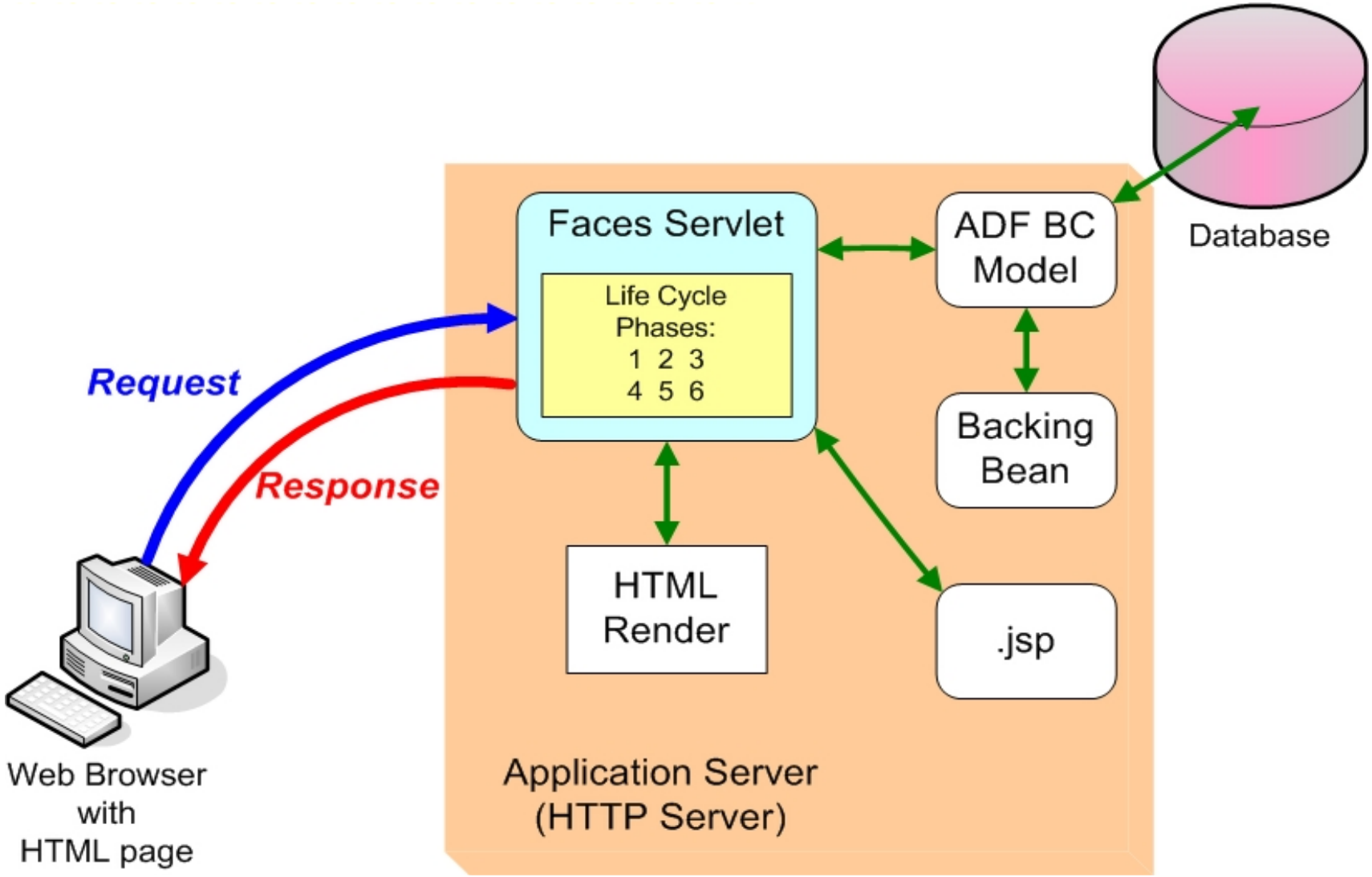
 Copyright @ 2011, John Jay King

# Creating Web Applications

- Oracle's Business Component Browser is impressive, but hardly a customer-facing interface
- ADF Faces extends the Java Server Faces (JSF) framework using XML tags to describe the UI
- ADF Faces provides a Rich-Client Interface that uses JavaScript and AJAX components; users must have a reasonably up-to-date browser (Internet Explorer 7.0 or higher, Mozilla Firefox 2.0 or higher, Safari 3.0 or higher) to use all of its features
- ADF Faces is designed to make creation of "rich-client" (RC) interfaces full-featured and declarative where possible

 Copyright @ 2011, John Jay King

# Review of Web Processing




 Copyright @ 2011, John Jay King

# HTML, CSS, and Forms

- Even though the ultimate page delivered to the Client Browser is HTML; with JDeveloper's Visual Editor and the combination of ADF Faces and JSF Faces it uses to create .jspx pages there is little need for ADF Developers to code HTML or CSS


- Yield to JDeveloper's declarative mechanism and refrain from coding

 Copyright @ 2011, John Jay King

# ADF Controller

- The ADF Controller extends the standard JSF controller and controls the MVC in ADF
- ADF Controller features include:
  - Sequence of page displays (may be conditional)
  - Allows partial-page processing in the same way as full page processing; only the necessary part of a page is rendered, the rest is unchanged (makes page processing faster)
  - Allows reuse of page parts
  - Provides conditional control of page flow

 Copyright @ 2011, John Jay King

# JSF Life Cycle

- JSF (and ADF Faces) follows a predictable cycle:

  1. Restore Components
  2. Apply Request Values
  3. Process Validations
  4. Update Model Values
  5. Invoke Application
  6. Render Response

- This Life Cycle is normally transparent; however, it is useful to understand it when debugging

 Copyright @ 2011, John Jay King

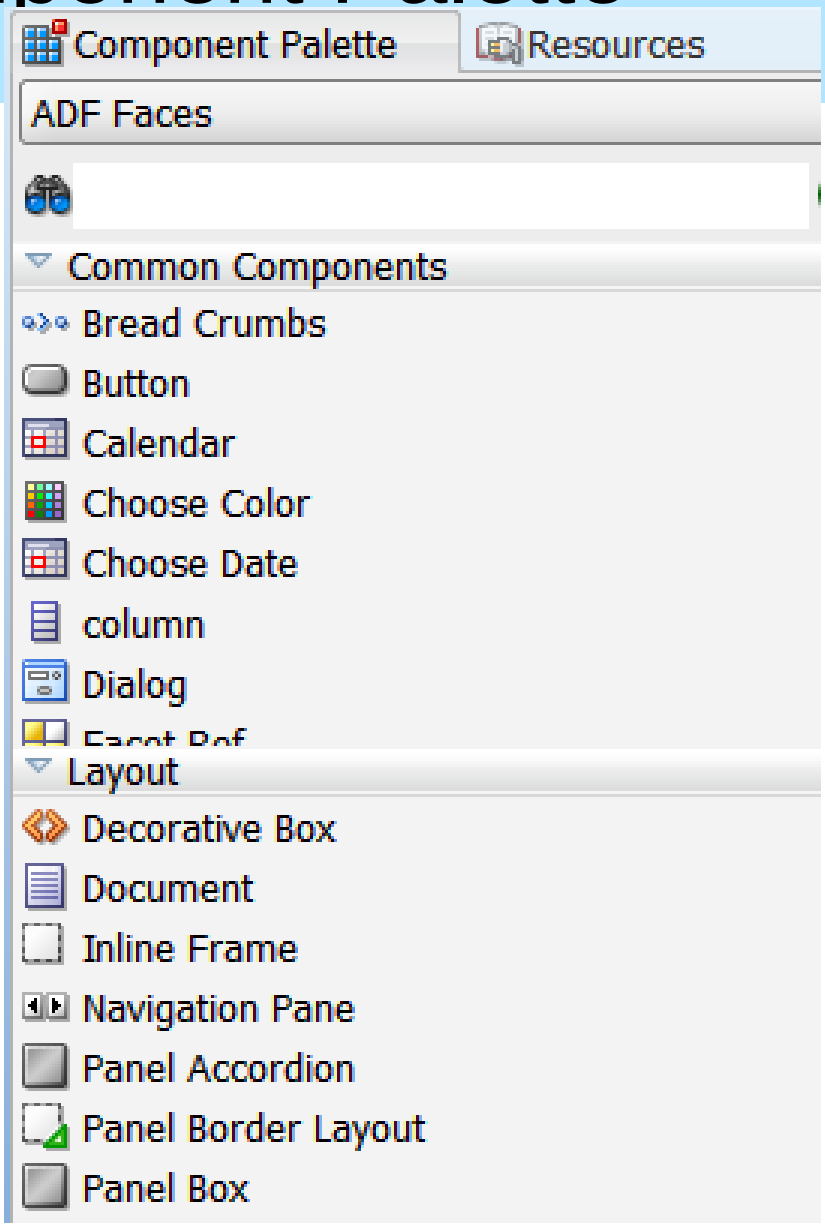
# JDeveloper Visual Designer

- JDeveloper's Visual Designer may be used to “paint” a User Interface using the Component Palette
- The JDeveloper Visual Designer is intended to be WYSIWYG (What You See Is What You Get); however the nature of the web and HTML is that it's really WYSIKOWYG (What You See Is Kind-Of What You Get; thank you Peter Koletzke…)

 Copyright @ 2011, John Jay King

# ADF Faces Component Palette

- The ADF Faces Component Palette includes icons representing various User Interface objects
- Drag-and-drop desired components into the position desired

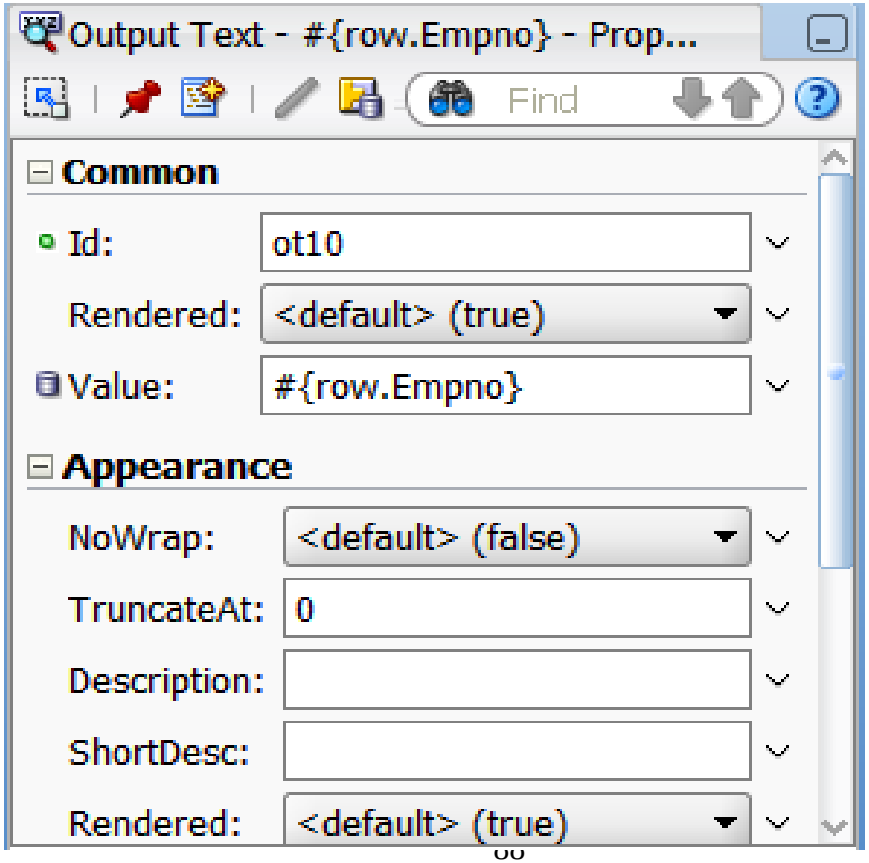




Copyright @ 2011, John Jay King

# Property Inspector

- When editing Web Pages, the Property Inspector shows properties for the various “facets” and components displayed upon the page

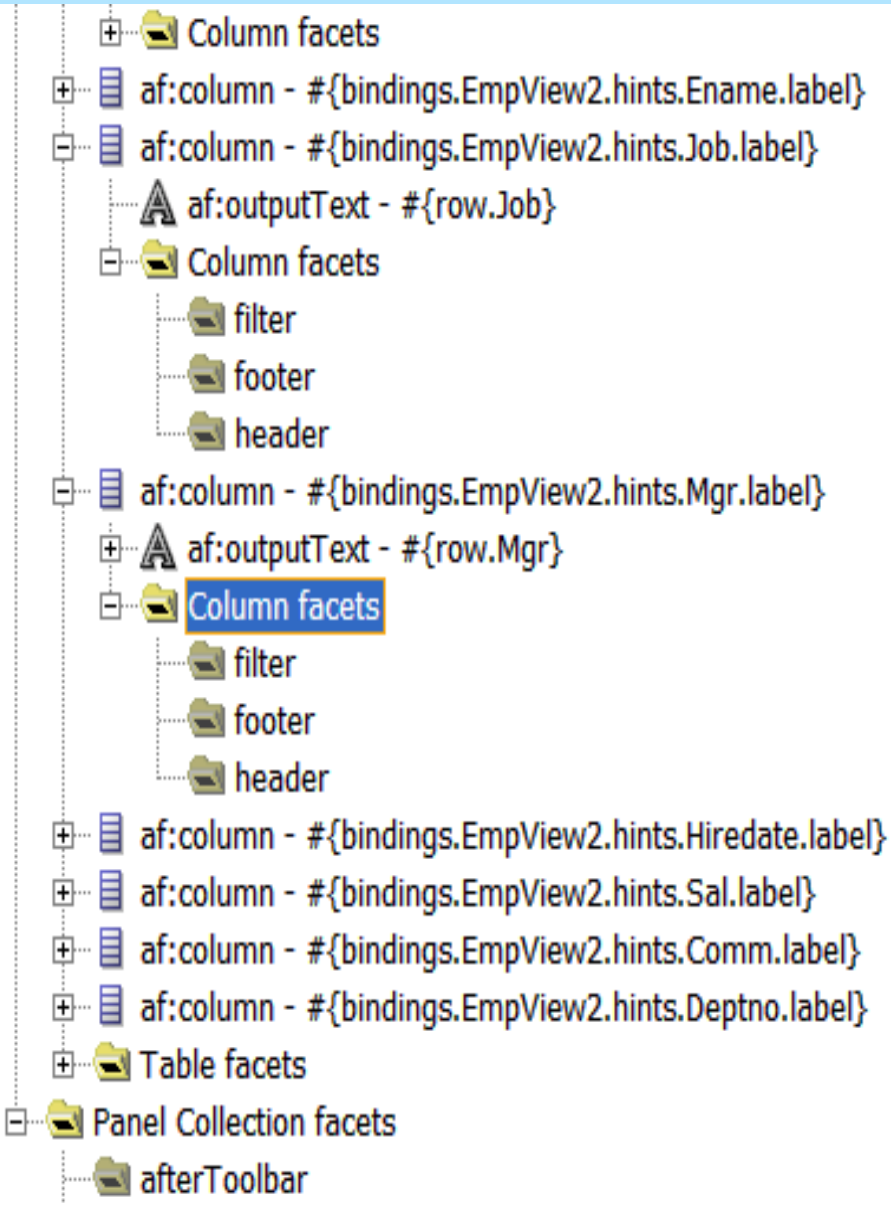




Copyright @ 2011, John Jay King

# Facets in Structure Window

- "Facets" are components used to contain groups of other components
- JDeveloper's "Structure" Window" lists facets for the current page (Note how the cursor position is synchronized between structure and visual editor)




 Copyright @ 2011, John Jay King

# Panel and Panel Splitter

- Pages in ADF are sometimes divided by Panels; pre-existing templates exist to help create the number of desired Panels
- Each Panel in turn may be divided into smaller areas using a Panel Splitter
  - By default Panel Splitters split an area horizontally
  - Panel Splitters have an “Orientation” property that allow the split to be vertical

 Copyright @ 2011, John Jay King

# Panel Collections, Accordions, Tabbed Panels

- Panel Collections are facets that contain other objects
- Panel Accordions are facets that contain other objects but shrink-and-grow depending upon mouse movement
- Tabbed Panels are facets that allow components to be placed into a tabbed structure

 Copyright @ 2011, John Jay King

# User Interface (UI) Components

- UI Components provided by ADF Faces include:
  - Buttons
  - Calendars
  - Choose Color
  - Forms
  - Input Text
  - Output Text
  - Panel Collection
  - Submit
  - Tables
  - more…

 Copyright @ 2011, John Jay King

# Binding Data

- JDeveloper's interface will allow not only the creation of web components using drag-and-drop processing
- Drag-and-drop may also be used to associate View Objects with UI Components
- This has the effect of “binding” the data to the data control object

 Copyright @ 2011, John Jay King

# Creating ADF/JSF Faces Pages

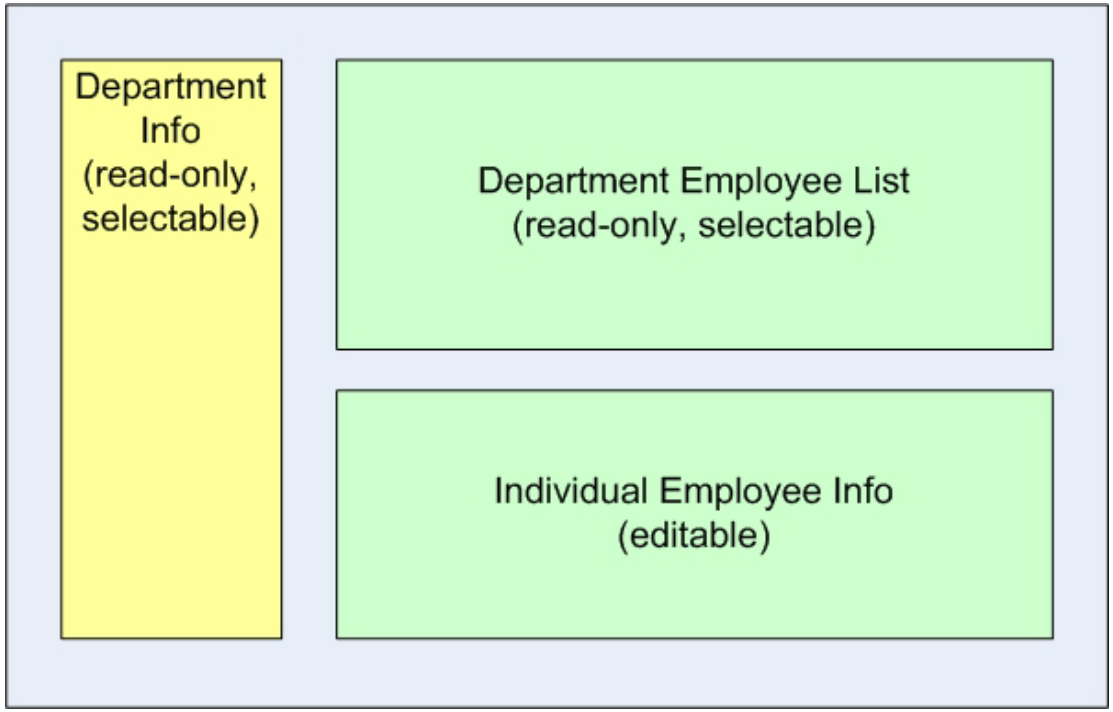
- The following pages walk through the creation of a simple Web Application using ADF Faces and ADF BC objects as follows:

1. Design Web Page
2. Create new JSF Page using JDeveloper
3. Add Visual Components to JSF Page
4. Bind Visual Components to ADF BC Objects

 Copyright @ 2011, John Jay King

# Target Screen Layout

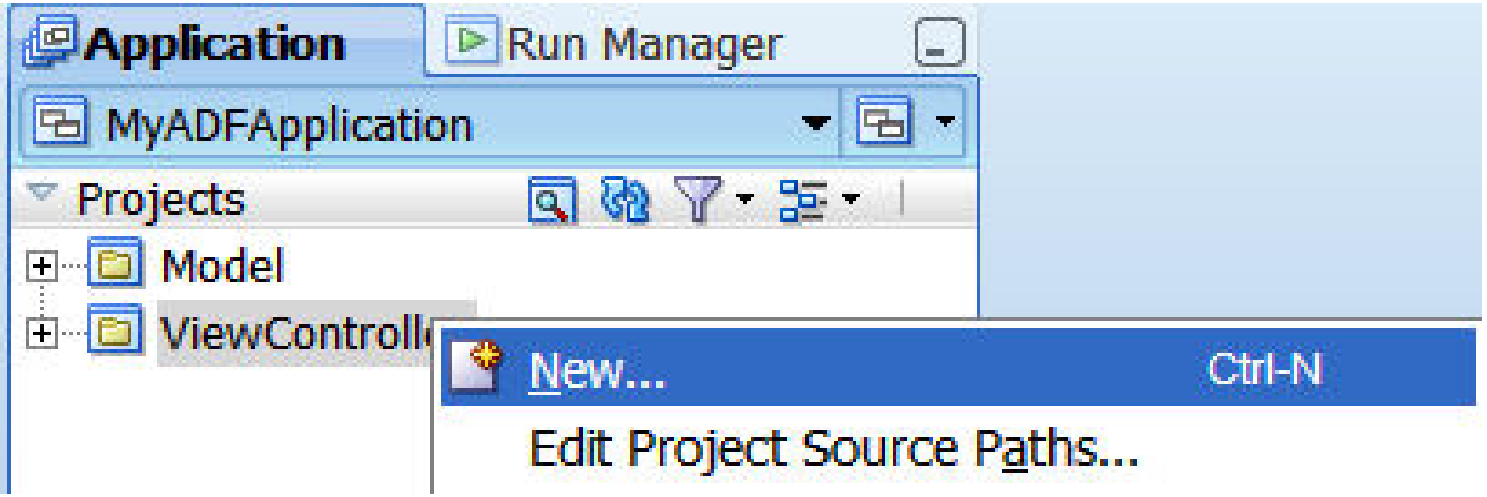
- Rough design: Department info on the left, list of Department Employees (for selected department) in the upper-right, and the information for a single employee on the lower-right (selected from list)




 Copyright @ 2011, John Jay King

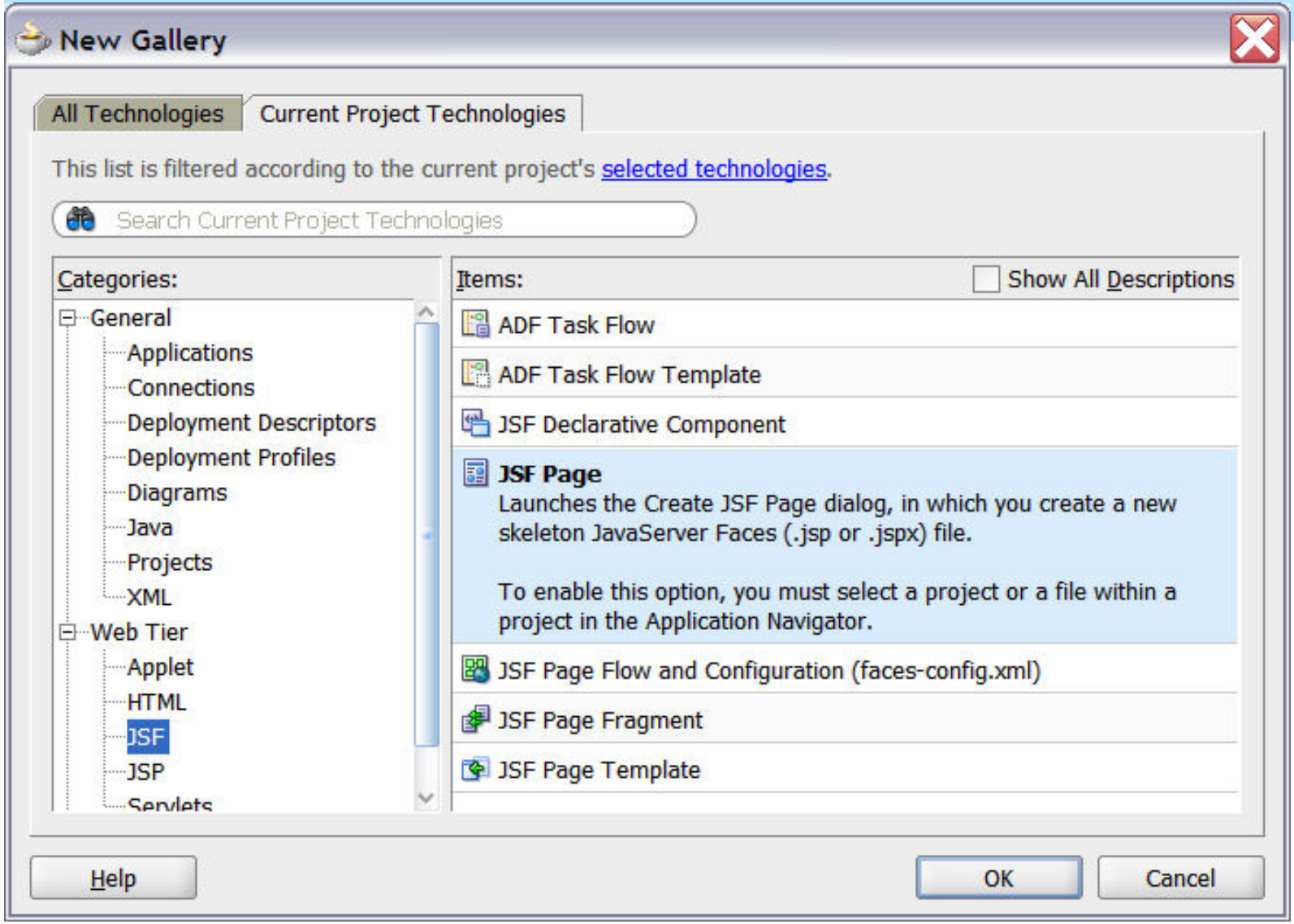
# Create ADF Faces Page

- To create an ADF Faces page, right-click on an Application's ViewController Project and choose “New” to display the “New Gallery” dialog



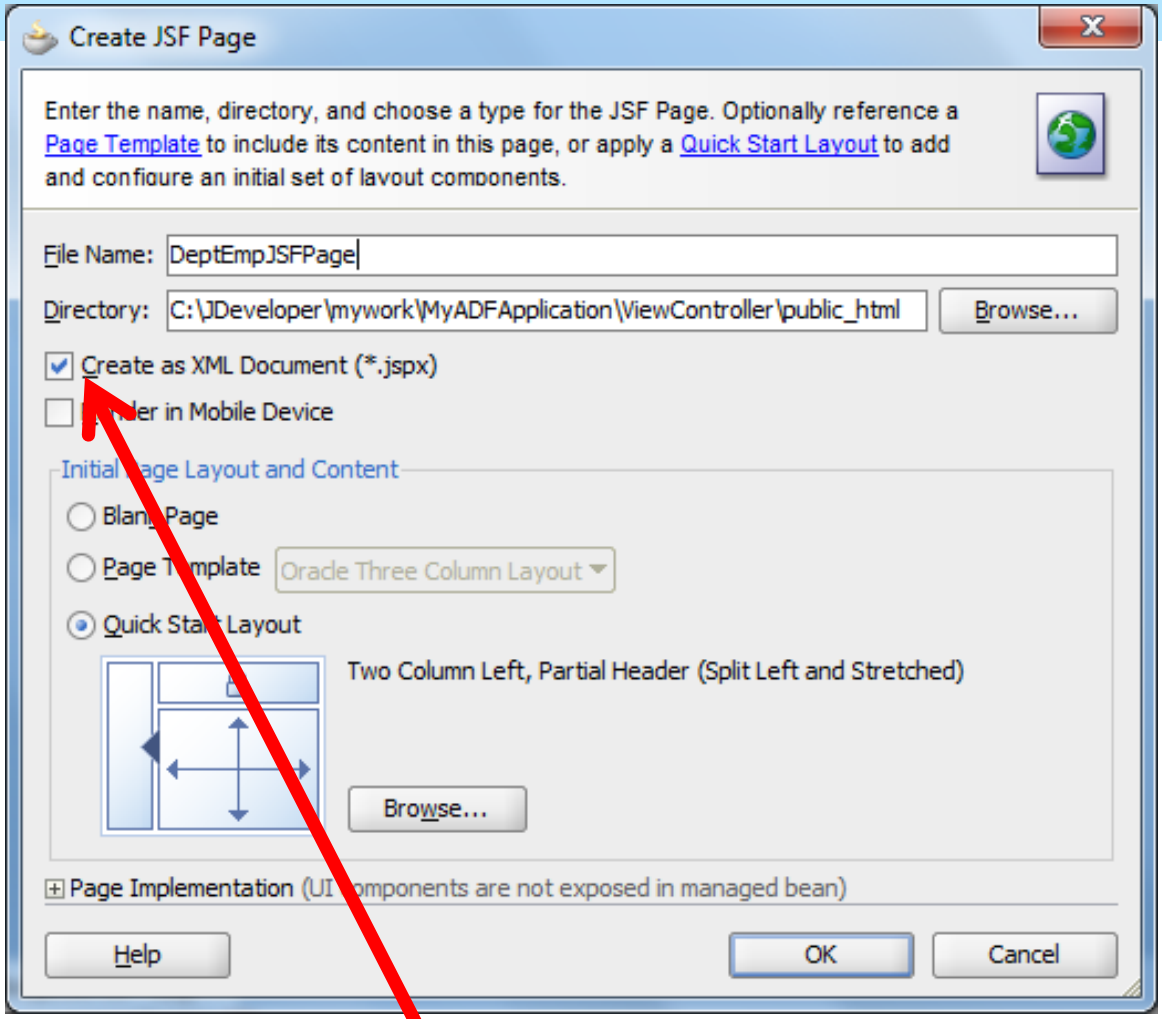

 Copyright @ 2011, John Jay King

# New Gallery




 Copyright @ 2011, John Jay King

# Naming Web Page




– Note the “Create as XML Document (*.jspx)” box

 Copyright @ 2011, John Jay King

# Visual Display with Initial Screen

Oracle JDeveloper 11g Release 1 - MyADFApplication.jws : ViewController.jpr : C:\JDeveloper\mywork\MyADFApplication\ViewController\public_html\DeptEmpJSF...

File Edit View Application Refactor Search Navigate Build Run Design Versioning Tools Window Help

Application Navig... | Run Manager

MyADFApplication

Projects
- Web Content
  - WEB-INF
    - adfc-config.xml
    - faces-config.xml
    - trinidad-config.xml
    - web.xml
  - Page Flows
    - DeptEmpJSFPage.jspx

Application Resources

Data Controls

Recently Opened Files

DeptEmpJSFPage.jspx - Structure
- Warnings (2)
- jsp:root
  - jsp:directive.page
  - f:view
    - af:document
      - af:form
        - af:panelSplitter - horizontal
          - Panel Splitter facets

Source | Design

DeptEmpJSFPage.jspx

Show | Full Screen Size | None | Default | None

top

first

center

f:view > af:document#d1 > af:form#f1 > af:panelsplitter#ps1 > f:facet

Design | Source | Bindings | Preview | History

Running: IntegratedWebLogicServer - Log

Actions

```
<Nov 28, 2010 6:22:14 PM UTC> <Notice> <Server> <BEA-002613> <Cha
<Nov 28, 2010 6:22:14 PM UTC> <Notice> <WebLogicServer> <BEA-0003
<Nov 28, 2010 6:22:14 PM UTC> <Notice> <WebLogicServer> <BEA-0003
<Nov 28, 2010 6:22:14 PM UTC> <Notice> <WebLogicServer> <BEA-0003
IntegratedWebLogicServer startup time: 44318 ms.
IntegratedWebLogicServer started.
```

Messages | Extensions | Running: IntegratedWebLogicServer

Component Palette | Re...

ADF Faces

Common Components
- Bread Crumbs
- Button
- Calendar
- Carousel

Layout
- Decorative Box
- Document
- Inline Frame

Operations

Facet first - Property Inspector

Find

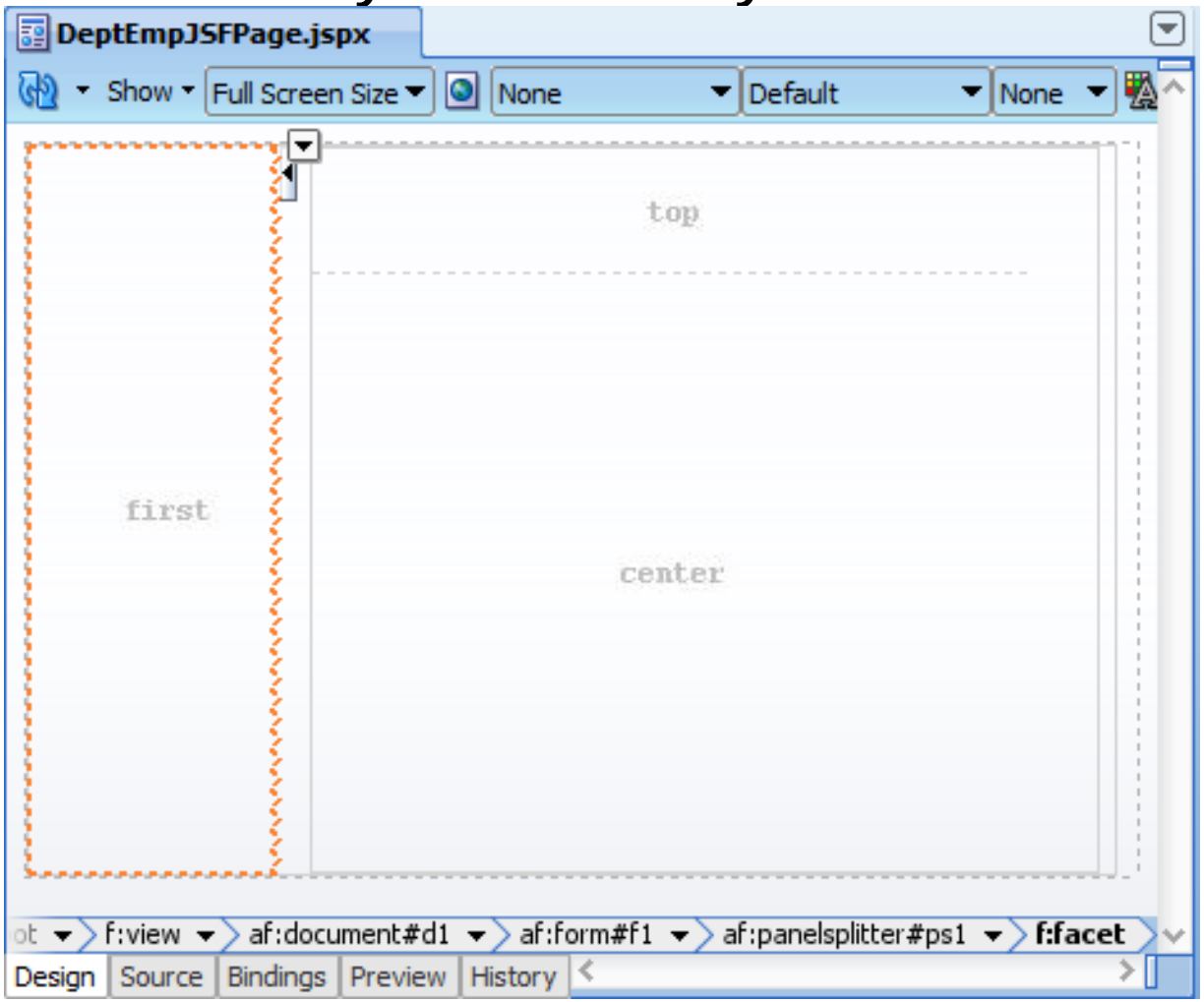
Name *: first

Web Editing


Copyright @ 2011, John Jay King

# Quick-Start Layout

- The supplied quick-start layout is ready to have objects dropped into it




 Copyright @ 2011, John Jay King

# Common Components

Component Palette

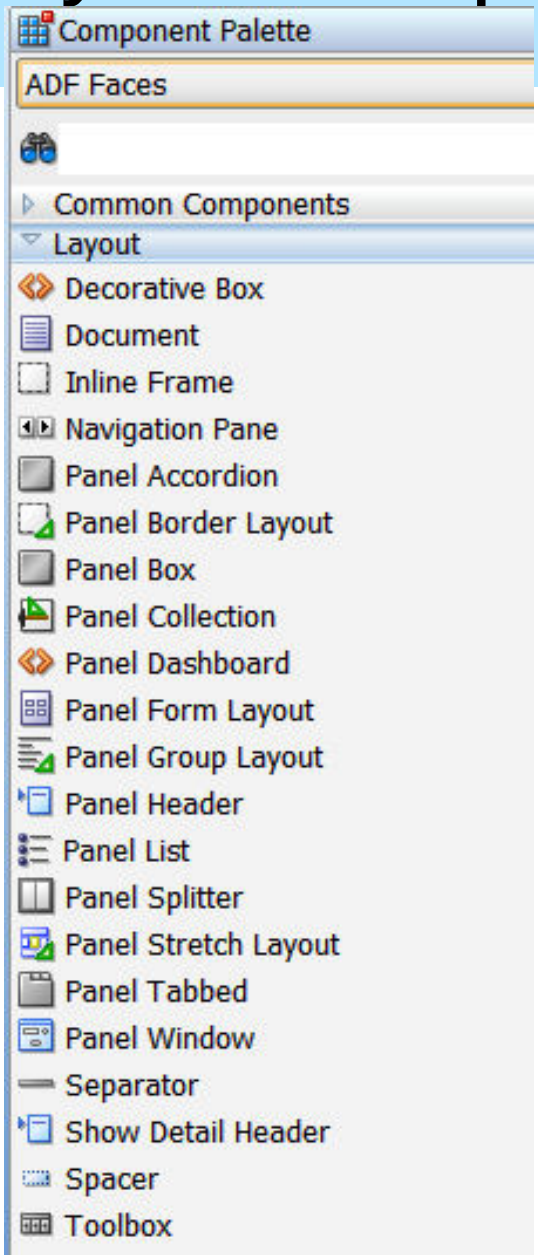
ADF Faces

Common Components

- Bread Crumbs
- Button
- Calendar
- Choose Color
- Choose Date
- column
- Dialog
- Facet Ref
- Form
- Go Button
- Go Link
- Go Menu Item
- Icon
- Image
- Image (Active)
- Image Link
- Input Color
- Input Combobox List Of Values
- Input Date
- Input File
- Input List Of Values
- Input Number Slider
- Input Number Spinbox
- Input Range Slider
- Input Text
- Input Text
- Link
- Media
- Menu
- Menu Item
- Message
- Messages
- Navigation Item
- Note Window
- Output Formatted
- Output Label
- Output Text
- Output Text (Active)
- Panel Label and Message
- Panel Menu Bar
- Popup
- Progress Indicator
- Query
- Quick Query
- Reset Button
- Rich Text Editor
- Select Boolean Checkbox
- Select Boolean Radio
- Select Item
- Select Many Checkbox
- Select Many Listbox
- Select Many Shuttle
- Select One Choice
- Select One Listbox
- Select One Radio
- Select Order Shuttle
- Show Detail
- Show Detail Item
- Status Indicator
- Subform
- Table
- Toolbar
- Toolbar Button
- Toolbar Button (Active)
- Train
- Train Button Bar
- Tree
- Tree Table
- Xml Content


Copyright @ 2011, John Jay King

# Layout Components






Copyright @ 2011, John Jay King

# Operations Components

Component Palette

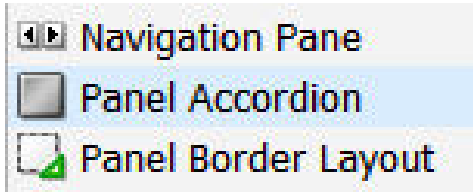
ADF Faces

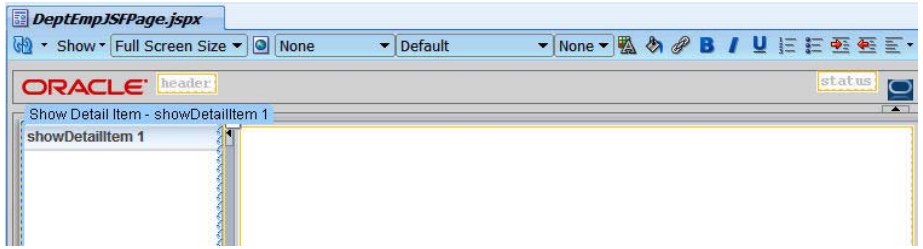
- Common Components
- Layout
- Operations
  - Attribute Drag Source
  - Attribute Drop Target
  - Calendar Drop Target
  - Client Attribute
  - Client Listener
  - Collection Drag Source
  - Collection Drop Target
  - Component Drag Source
  - Convert Color
  - Convert Date Time
  - Convert Number
  - Data Flavor
  - Drop Target
  - Export Collection Action Listener
  - File Download Action Listener
  - For Each
  - Group
  - Insert Text Behavior
  - Iterator
  - Panel Dashboard Behavior
  - Poll
  - Reset Action Listener
  - Resource
  - Return Action Listener
  - Rich Text Editor Insert Behavior
  - Scroll Component Into View Behavior
  - Server Listener
  - Set Action Listener
  - Set Property Listener
  - Show Popup Behavior
  - Show Printable Page Behavior
  - Skip Link Target
  - Switcher
  - Validate Byte Length
  - Validate Date Restriction
  - Validate Date Time Range
  - Validate Double Range
  - Validate Length
  - Validate Long Range
  - Validate Reg Exp


Copyright @ 2011, John Jay King

# Adding Accordion Component

- To add an Accordion Component to the web page; Panel Accordion component from the pallet to the desired column (“start”)

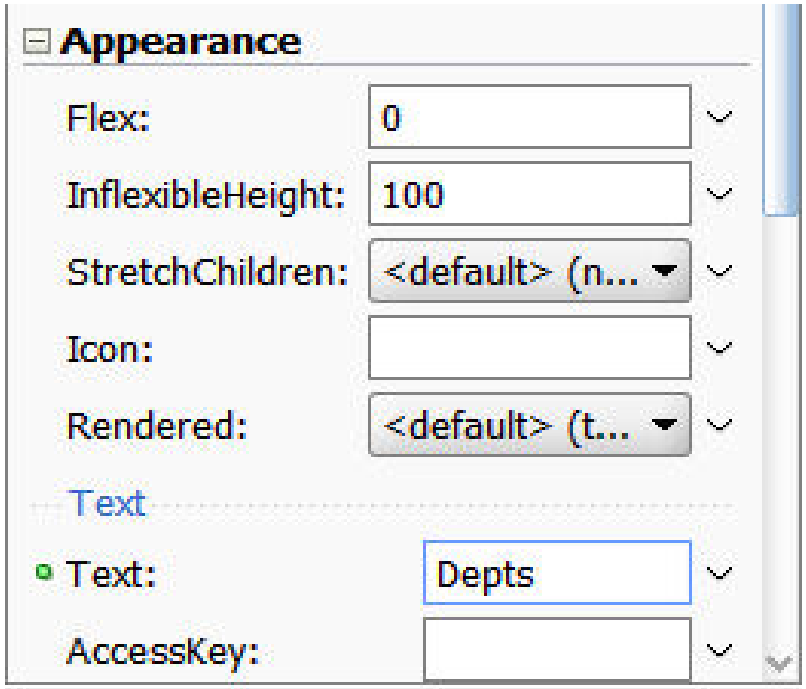



 Copyright @ 2011, John Jay King

# Change Accordion Title Property

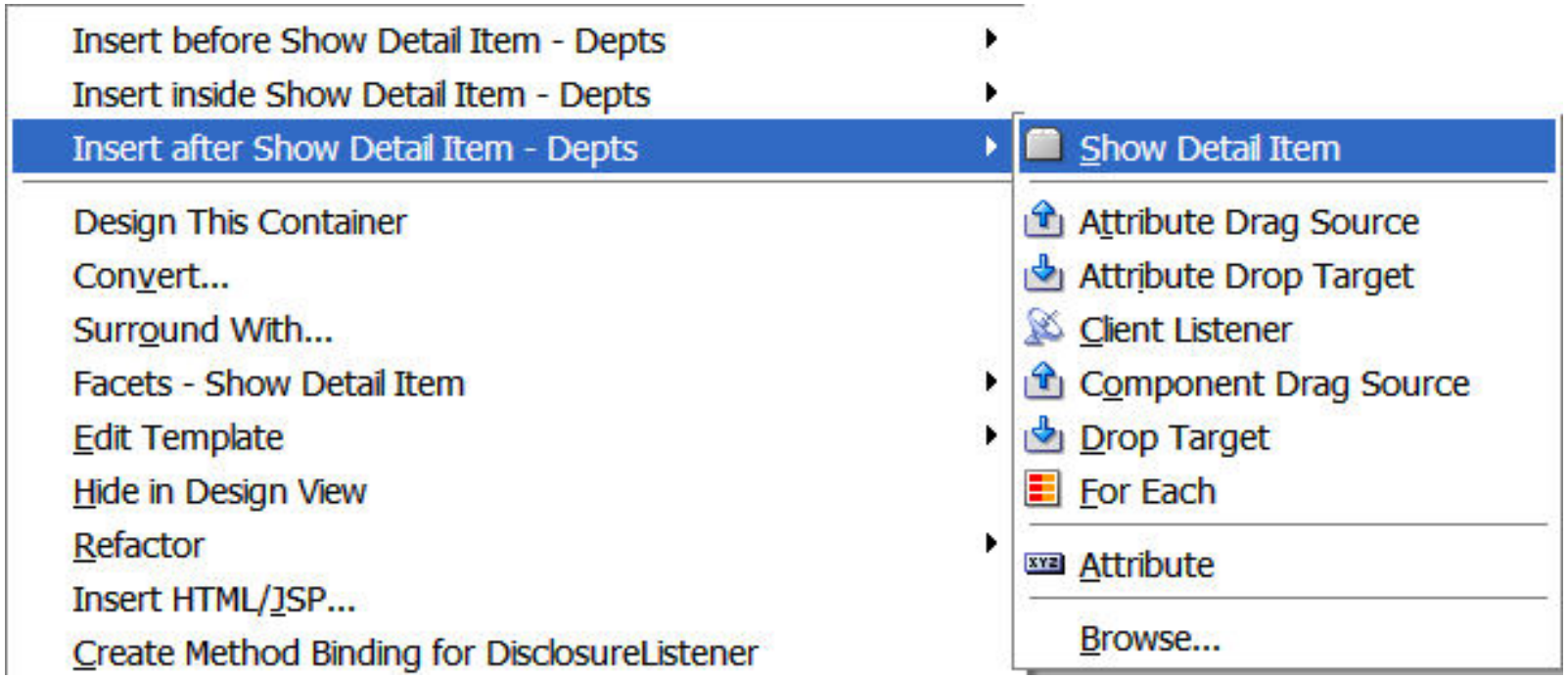
- To alter the Accordion's title, click on the Accordion and modify its Property Inspector Text item (changed to “Depts”)




 Copyright @ 2011, John Jay King

# Add Data Component

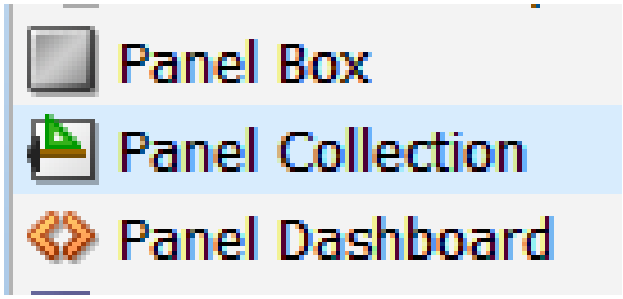
- Right-click in the “Depts” Accordion; when prompted choose “Insert After Show Details Item - Depts -> Show Detail Item” to add another Accordion to the page (not used further in this demo…)



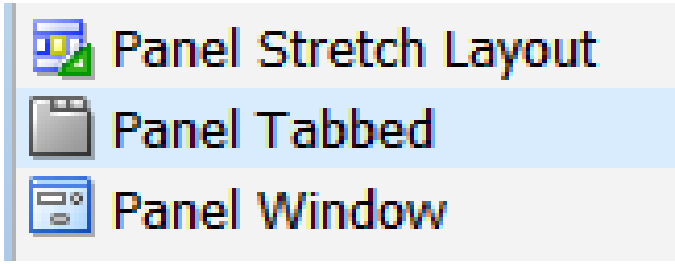

 Copyright @ 2011, John Jay King

# Adding Collection and Tabbed Area

- Find the “Panel Collection” component in the Layout components and drag it to the “first” (top) part of the right half of the screen



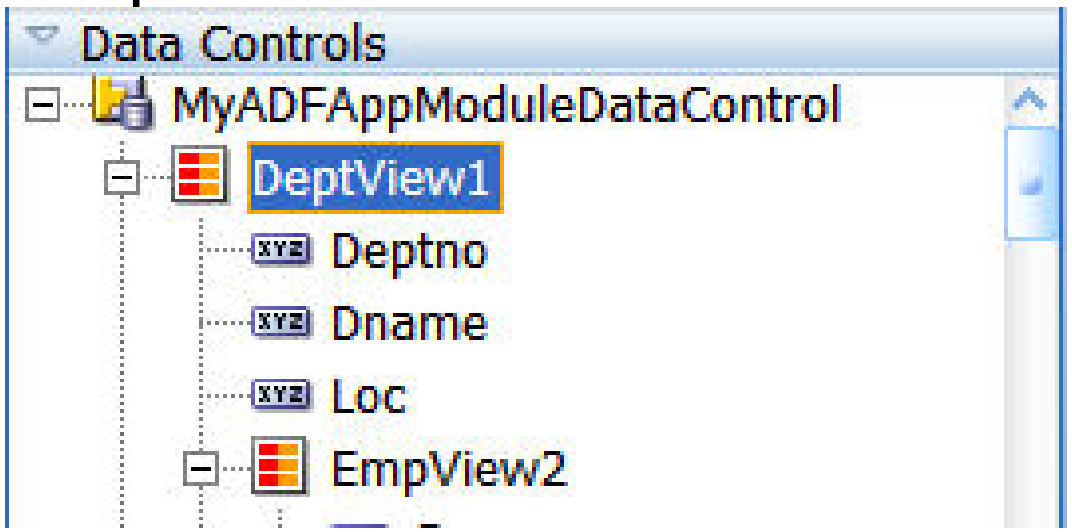

- Find the “Panel Tabbed” component in the Layout components and drag it to the “second” (center/bottom) part of the right half of the screen




 Copyright @ 2011, John Jay King

# Data Binding: Adding Data, 1

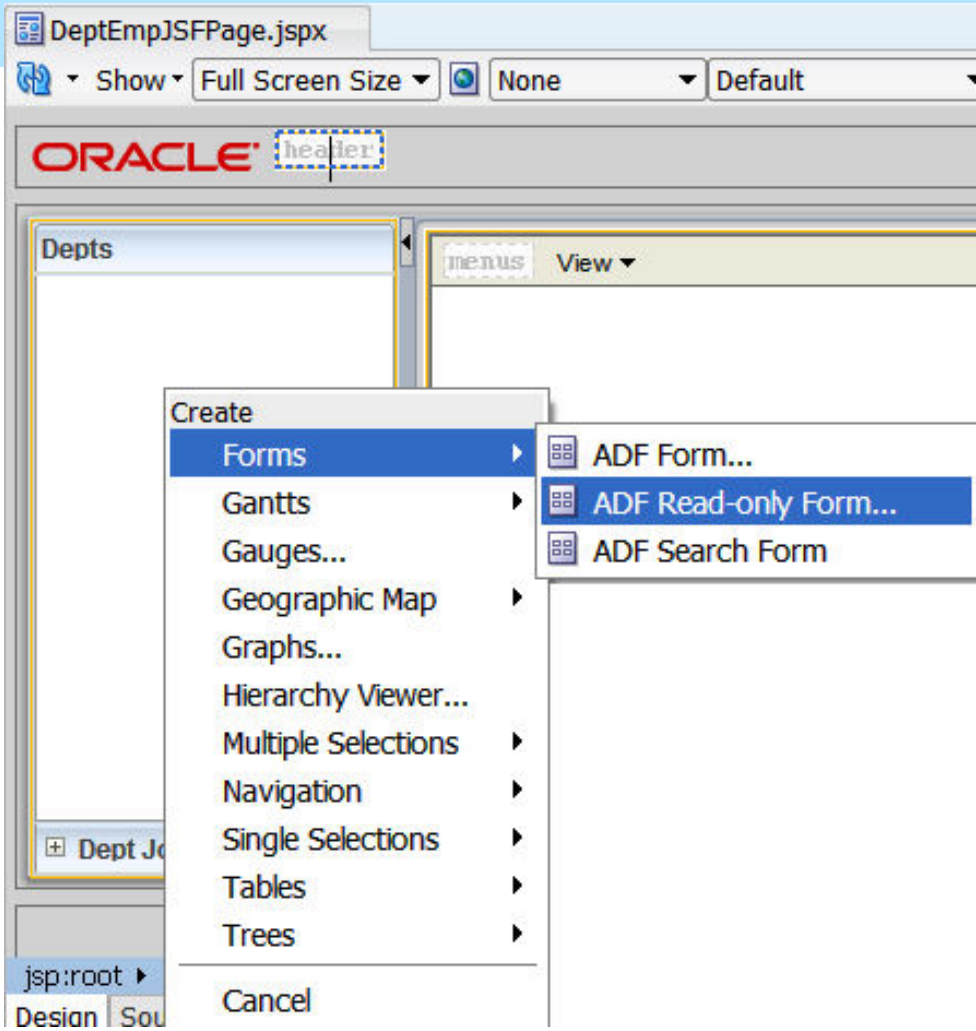
- To “bind” data to web page components, simply drag ADF BC data objects to the Visual Editor
- Open the “Application Navigator” and expand the “Data Controls” accordion to see the ADF BC components created earlier then drag “DeptView1” to the “Depts” accordion




 Copyright @ 2011, John Jay King

# Data Binding: Adding Data, 2

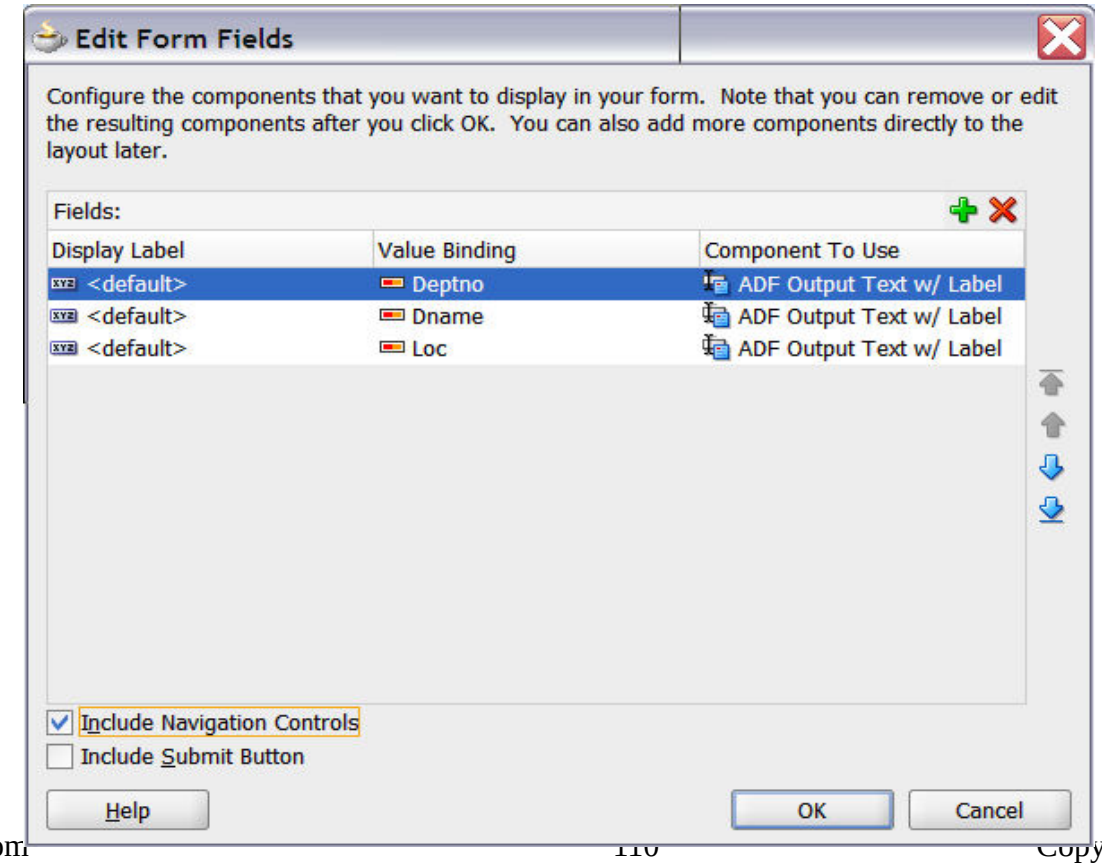
- When prompted; choose “Create Forms -> ADF Read-Only Form” to populate the Department data display




 Copyright @ 2011, John Jay King

# Adding Navigation Controls

- Check the “Include Navigation Controls” box
- You may also modify display labels and add, delete, or reorganize the values displayed

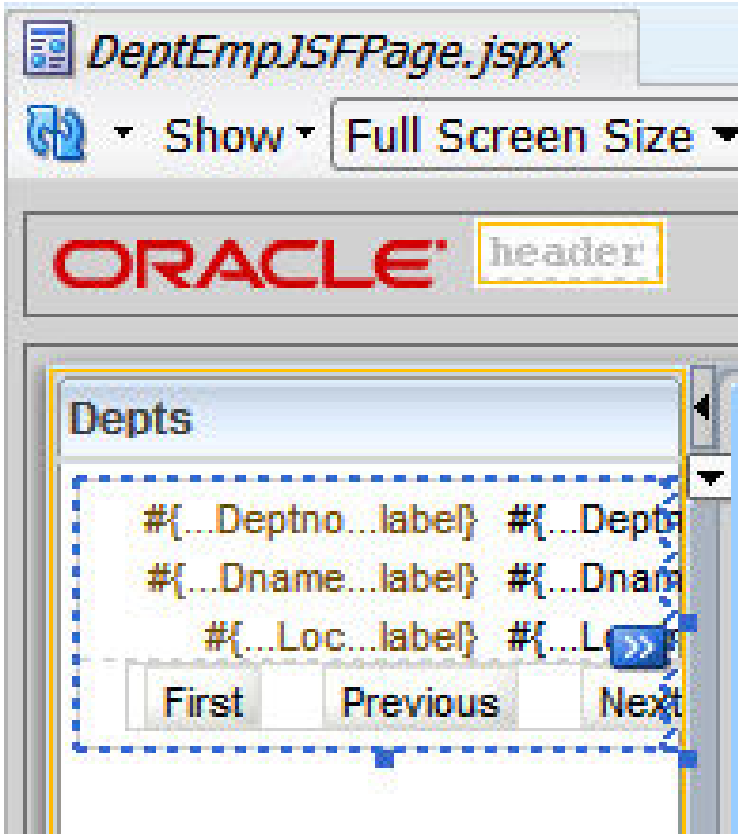




Copyright @ 2011, John Jay King

# Department Display Area

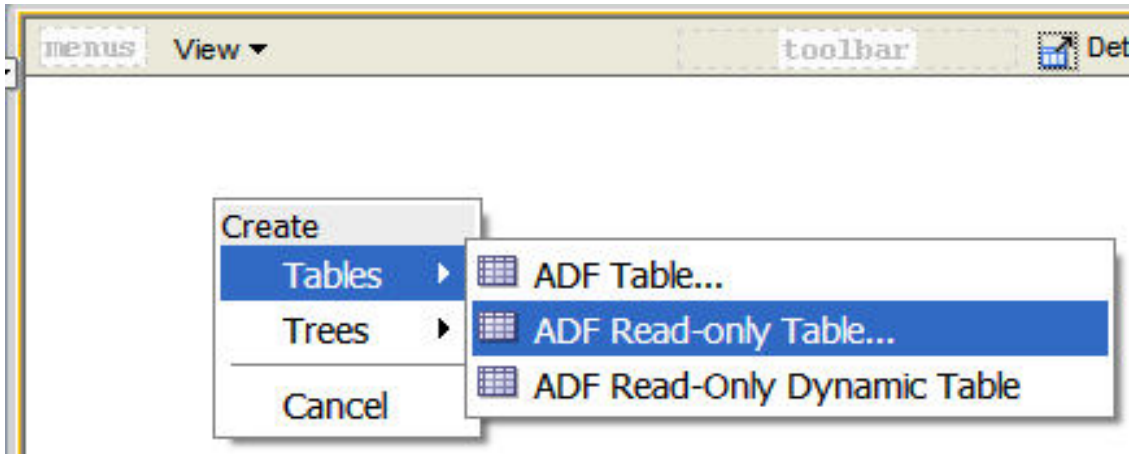
- After adding the Department information; the "Depts" accordion should look like the following

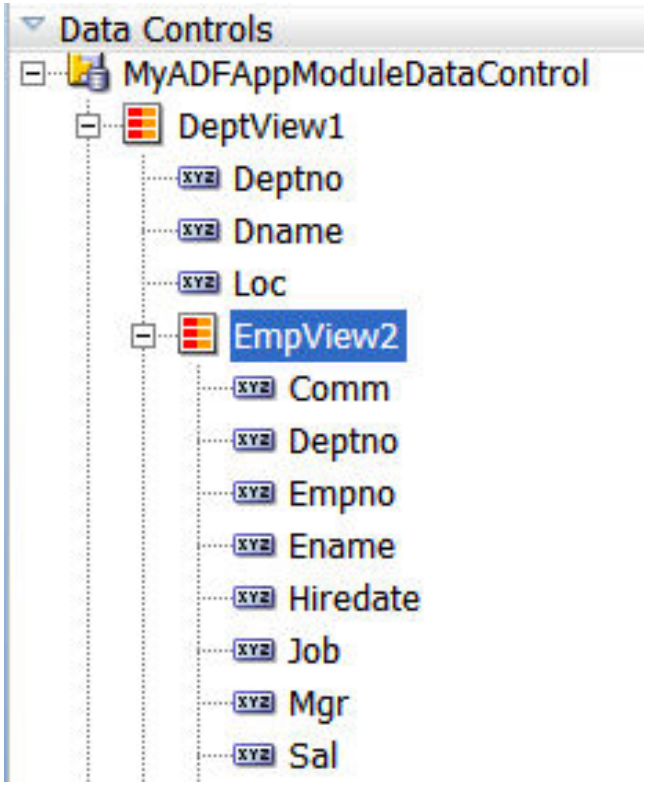



 Copyright @ 2011, John Jay King

# Adding Department Employees

- Next, to add Department Employees to the page, drag the EmpView2 data



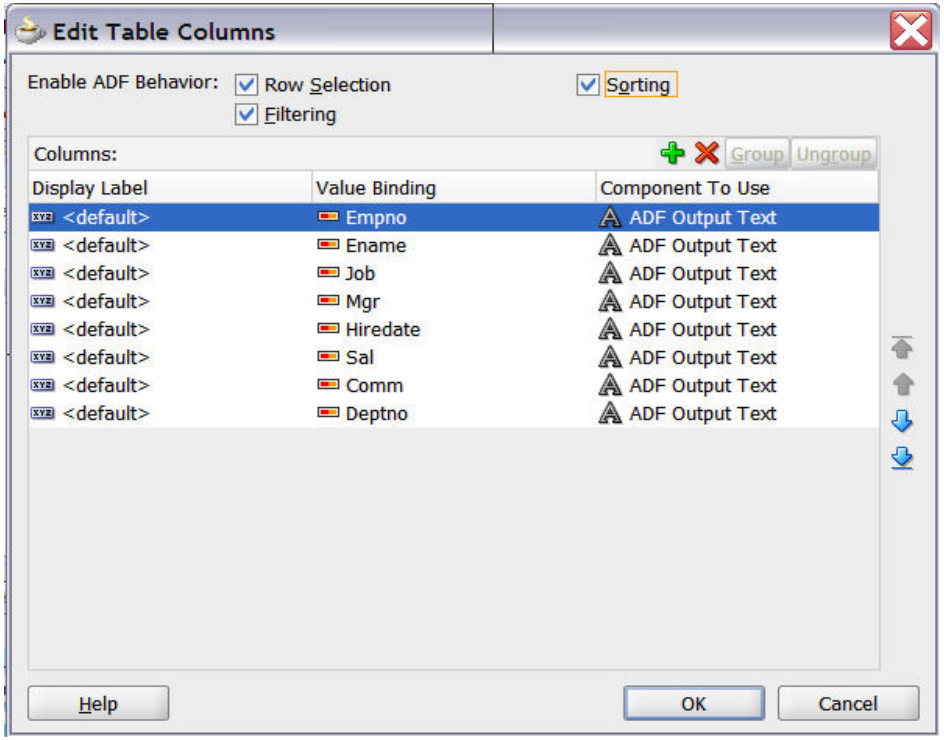




When prompted, choose "Create Tables -> ADF Read-Only Table"

 Copyright @ 2011, John Jay King

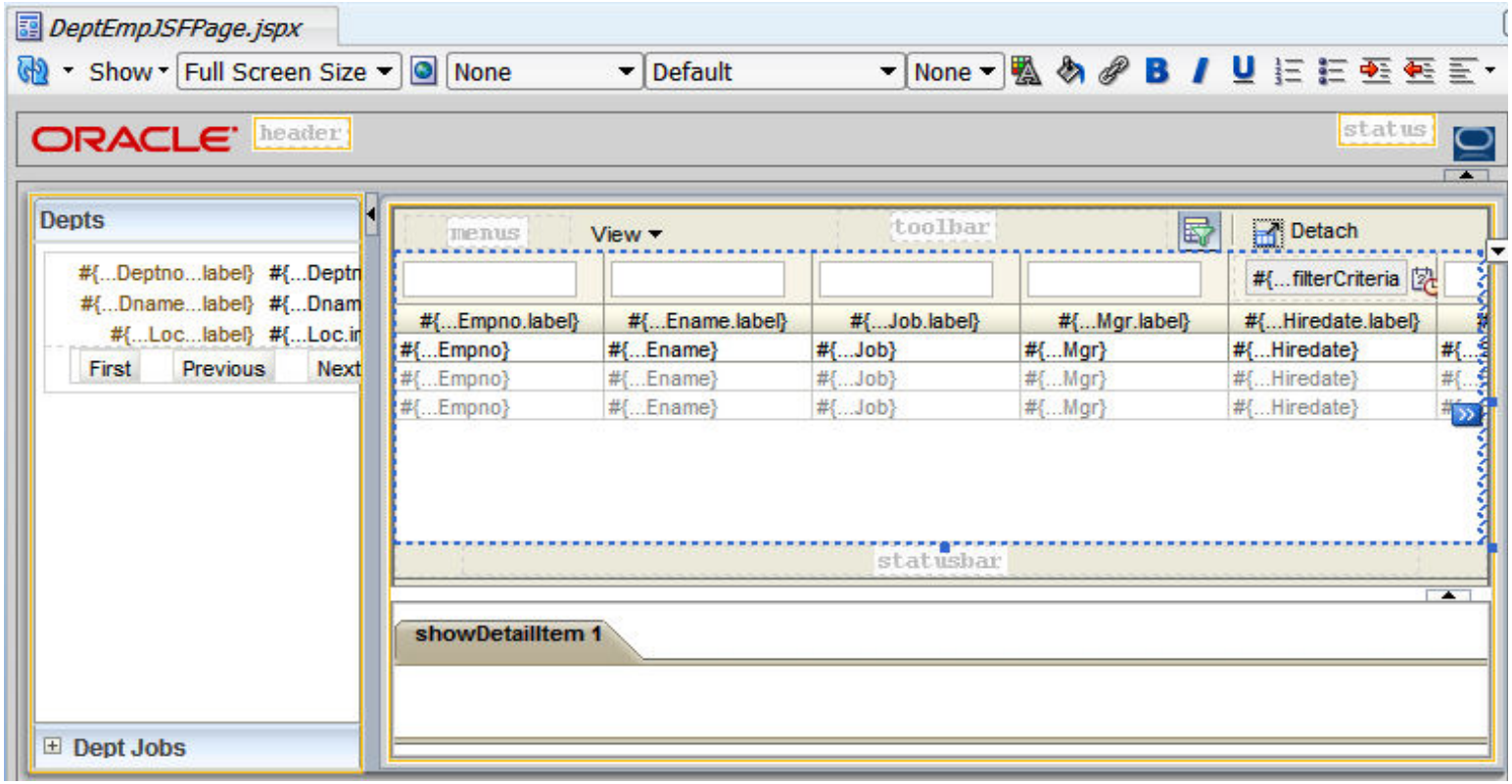
# Add Employee Navigation Controls

- Check all three navigation controls:
- Row Selection (user may select), filtering (user may search), and sort; as before columns may be relabeled, added, deleted, reorganized




 Copyright @ 2011, John Jay King

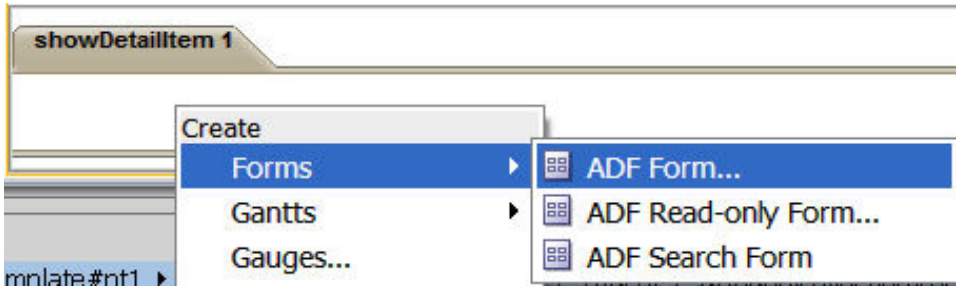
# Department Employee Area

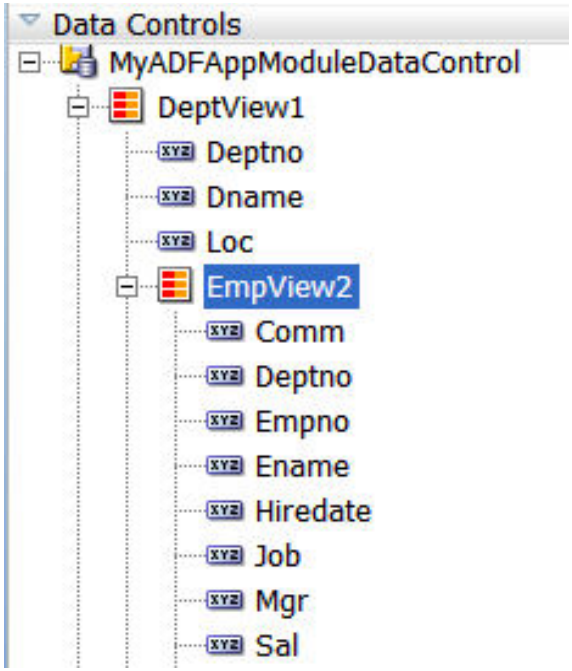



 Copyright @ 2011, John Jay King

# Adding Individual Employee

- Finally, add the individual Employee display to the Tabbed area at the bottom of the page



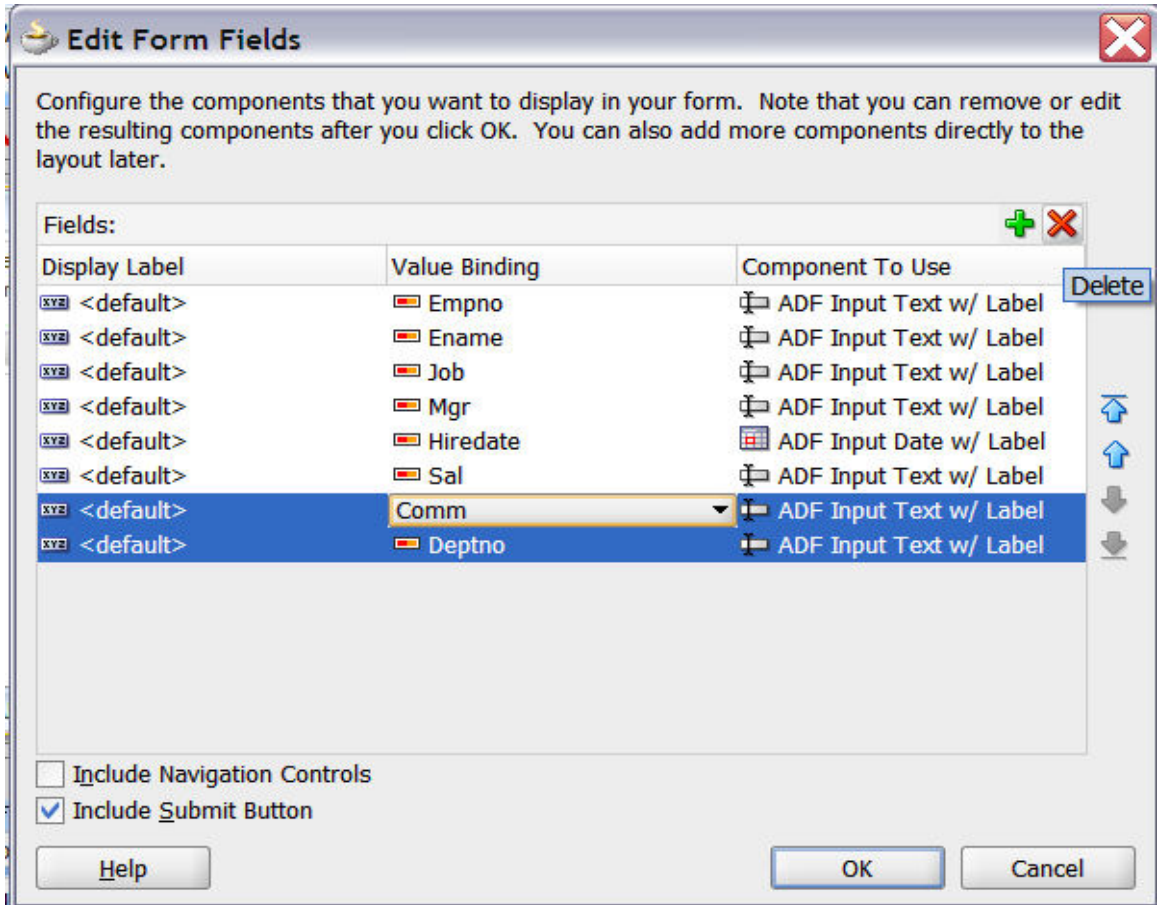




- When prompted, choose “Create Forms -> ADF Form” to select the display format (this part of the form will be editable)

 Copyright @ 2011, John Jay King

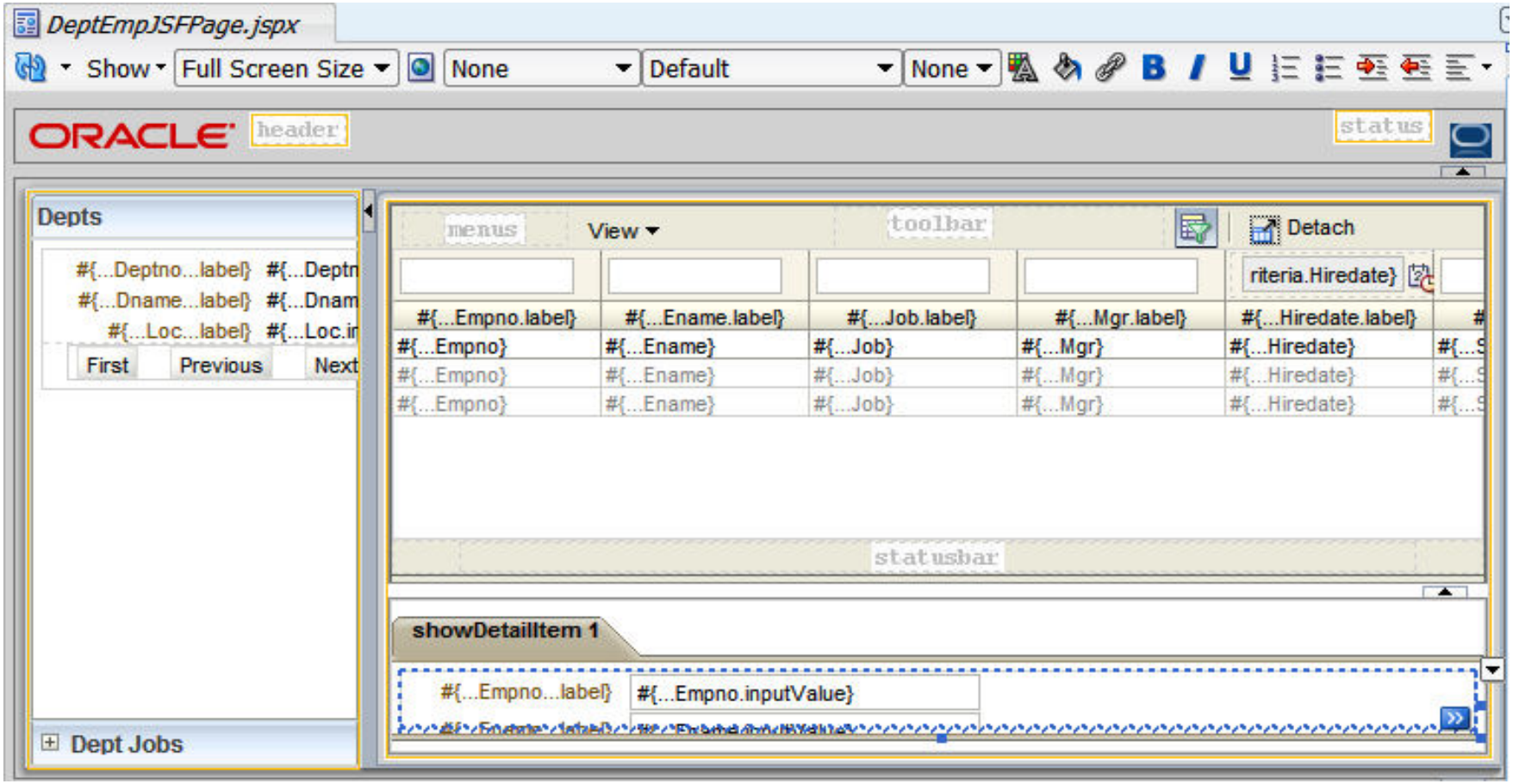
# Add Employee Navigation

- Delete the COMM and DEPTNO data from the display (highlight & click X); check “Include Submit Button”




 Copyright @ 2011, John Jay King

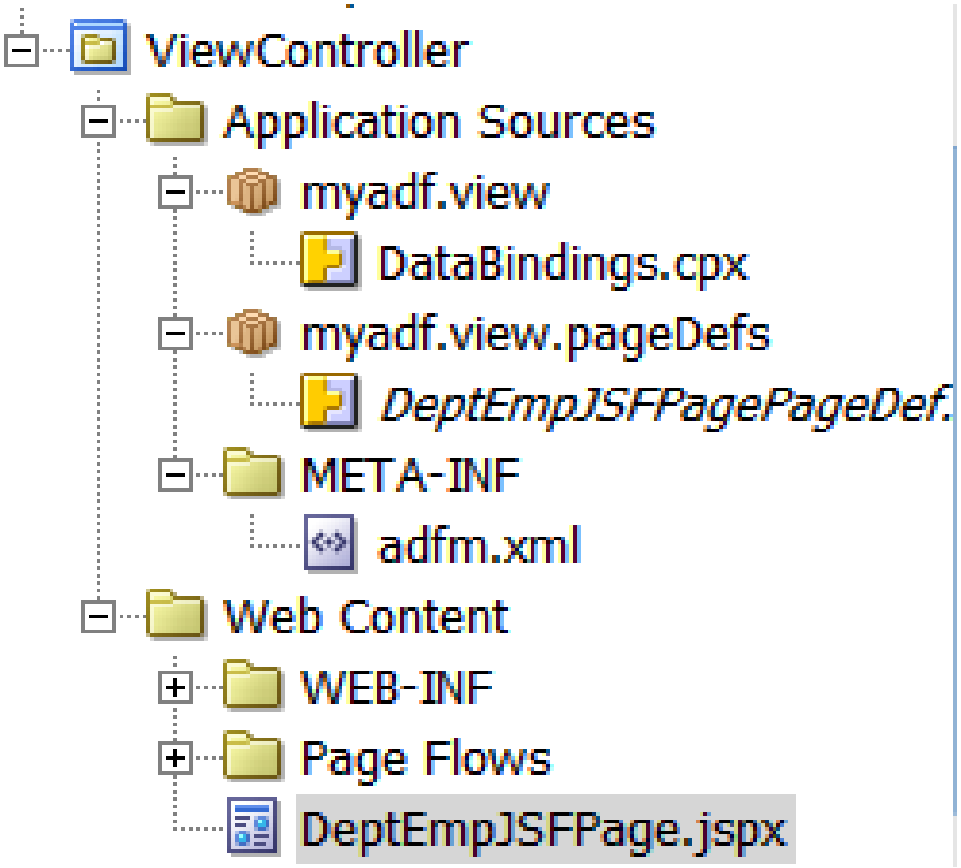
# Completed Web Application Page




 Copyright @ 2011, John Jay King

# Testing the Web Application

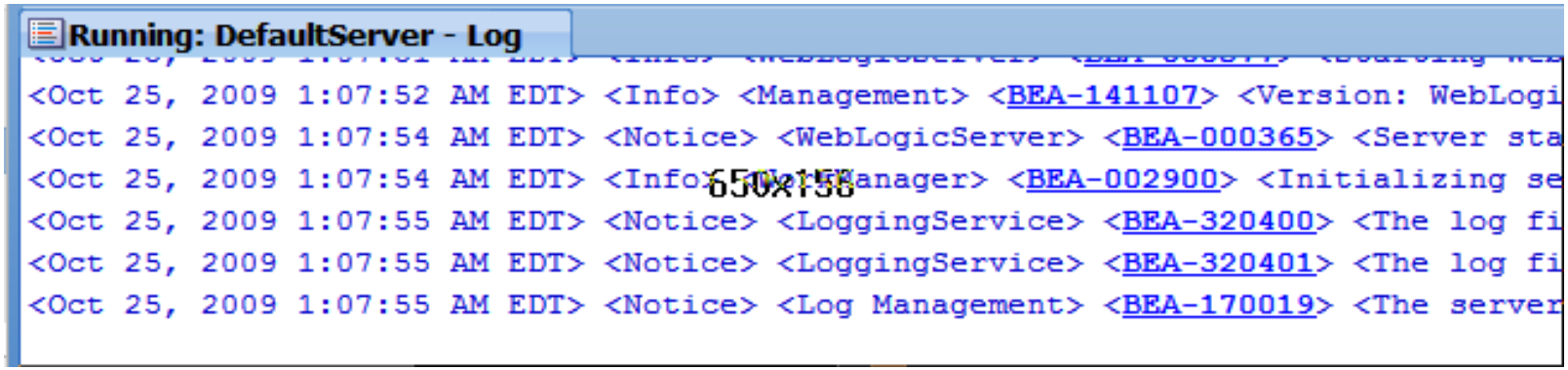
- To begin testing the Web Application; right-click the ".jspx" file created in the ViewController project and choose "Run"




 Copyright @ 2011, John Jay King

# Be Patient!

- The first time you execute a Web application JDeveloper starts its built-in WebLogic Application Server; this takes a while
- You can track the progress of the Server's startup in JDeveloper's DefaultServer Log




- Once the Server is “up” your web page should be displayed in a browser (again, please be patient!)

 Copyright @ 2011, John Jay King

# Web Page in Browser

http://127.0.0.1:7101/MyADFApplicatio...

Page Safety Tools

ORACLE

**Depts**

Deptno 10
Dname ACCOUNTING
Loc NEW YORK

First Previous Next

View Detach

| Empno | Ename | Job | Mgr | Hire Date | Sal |
|---|---|---|---|---|---|
| 7782 | CLARK | MANAGER | 7839 | 09.06.1981 | 2450 |
| 7839 | KING | PRESIDENT | | 17.11.1981 | 5000 |
| 7934 | MILLER | CLERK | 7782 | 23.01.1982 | 1300 |

**showDetailItem 1**

* Empno 7782
Ename CLARK
Job MANAGER
Mgr 7839
Hire Date 09.06.1981
Sal 2450

Submit

**Dept Jobs**


Copyright @ 2011, John Jay King

# Using ADF Runtime Controls

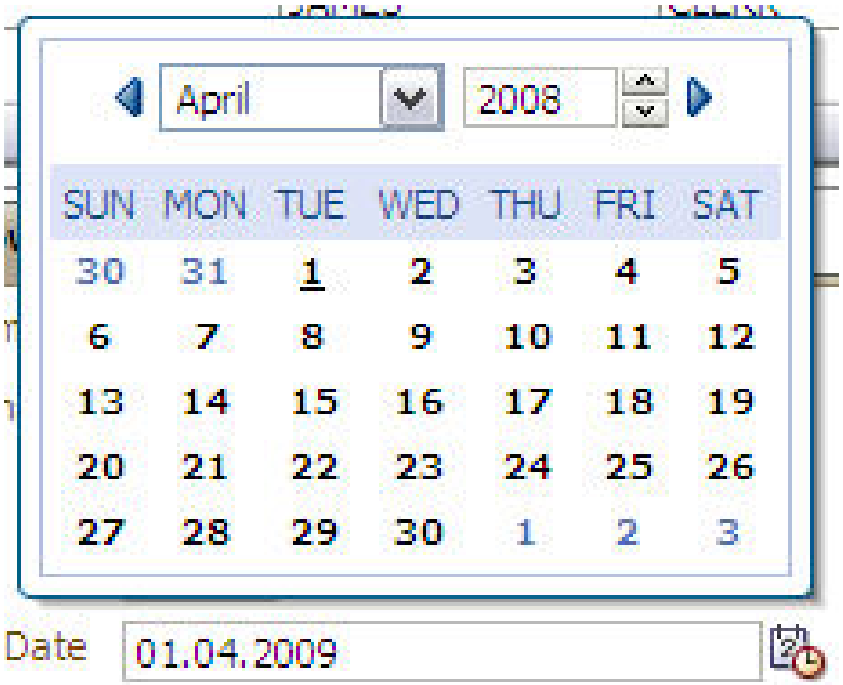
- ADF Faces enable a great deal of runtime customization of the output
- When “binding” tables you have the option to allow three features:


  - Row Selection — Allow user to select rows
  - Filtering — User may specify search
  - Sorting — Allow user to sort output
- The Property Inspector may be used to enable/disable these features on a column-by-column basis; in addition, “Column Selection” may be enabled

 Copyright @ 2011, John Jay King

# Date Calendar Rich-Client Object

- The Date “Calendar” icon is used to set a date; click the time icon next to the date field to display a full-function Calendar component




 Copyright @ 2011, John Jay King

# Files Supporting Web Application

- Several files make up the typical ADF Web Application
  - A .jspx file is used to define each web page
  - Web pages reference a page definition XML file (.xml)
  - Bindings are described in another XML file (.cpx)

 Copyright @ 2011, John Jay King

# JSF .jspx File

- ADF defines a web page using an XML .jspx file

DeptEmpJSFPage.jspx | Emp.xml

```
7   <f:view>
8     <af:document id="d1">
9       <af:messages id="m1"/>
10      <af:form id="f1">
11        <af:pageTemplate viewId="/oracle/templates/threeColumnTemplate.jspx"
12                         id="pt1">
13          <f:facet name="center">
14            <af:panelSplitter id="psl" orientation="vertical">
15              <f:facet name="first">
16                <af:panelCollection id="pc1">
17                  <f:facet name="menus"/>
18                  <f:facet name="toolbar"/>
19                  <f:facet name="statusbar"/>
20                  <af:table value="#{bindings.EmpView2.collectionModel}"
21                            var="row" rows="#{bindings.EmpView2.rangeSize}"
22                            emptyText="#{bindings.EmpView2.viewable ? 'No data to display
23                            fetchSize="#{bindings.EmpView2.rangeSize}"
24                            rowBandingInterval="0"
25                            filterModel="#{bindings.EmpView2Query.queryDescriptor}"
26                            queryListener="#{bindings.EmpView2Query.processQuery}"
27                            filterVisible="true" varStatus="vs"
28                            selectedRowKeys="#{bindings.EmpView2.collectionModel.selected
29                            selectionListener="#{bindings.EmpView2.collectionModel.makeCu
```

cument#d1 ▸ af:form#f1 ▸ af:pagetemplate#pt1 ▸ f:facet ▸ af:panelaccordion#pa1 ▸ **af:showdetailitem#sdi1**

Design | Source | Bindings | Preview | History

 Copyright @ 2011, John Jay King

# ADF Web Page Definition file (.xml)

DeptEmpJSFPage.jspx | DeptEmpJSFPagePageDef.xml | Emp.xml

Find

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<pageDefinition xmlns="http://xmlns.oracle.com/adfm/uimodel"
                version="11.1.1.54.7" id="DeptEmpJSFPagePageDef"
                Package="myadf.view.pageDefs">
  <parameters/>
  <executables>
    <variableIterator id="variables"/>
    <iterator Binds="DeptView1" RangeSize="25"
              DataControl="MyADFAppModuleDataControl" id="DeptView1Iterator"
              ChangeEventPolicy="ppr"/>
    <iterator Binds="EmpView2" RangeSize="25"
              DataControl="MyADFAppModuleDataControl" id="EmpView2Iterator"
              ChangeEventPolicy="ppr"/>
    <searchRegion Binds="EmpView2Iterator" Criteria=""
                  Customizer="oracle.jbo.uicli.binding.JUSearchBindingCustomizer"
                  id="EmpView2Query"/>
  </executables>
  <bindings>
    <attributeValues IterBinding="DeptView1Iterator" id="Deptno">
      <AttrNames>
        <Item Value="Deptno"/>
      </AttrNames>
    </attributeValues>
    <attributeValues IterBinding="DeptView1Iterator" id="Dname">
```

Overview | Source | History


Copyright @ 2011, John Jay King

# ADF Bindings XML file (.cpx)

DeptEmpJSFPage.jspx | DeptEmpJSFPagePageDef.xml | **DataBindings.cpx** | Emp.xml

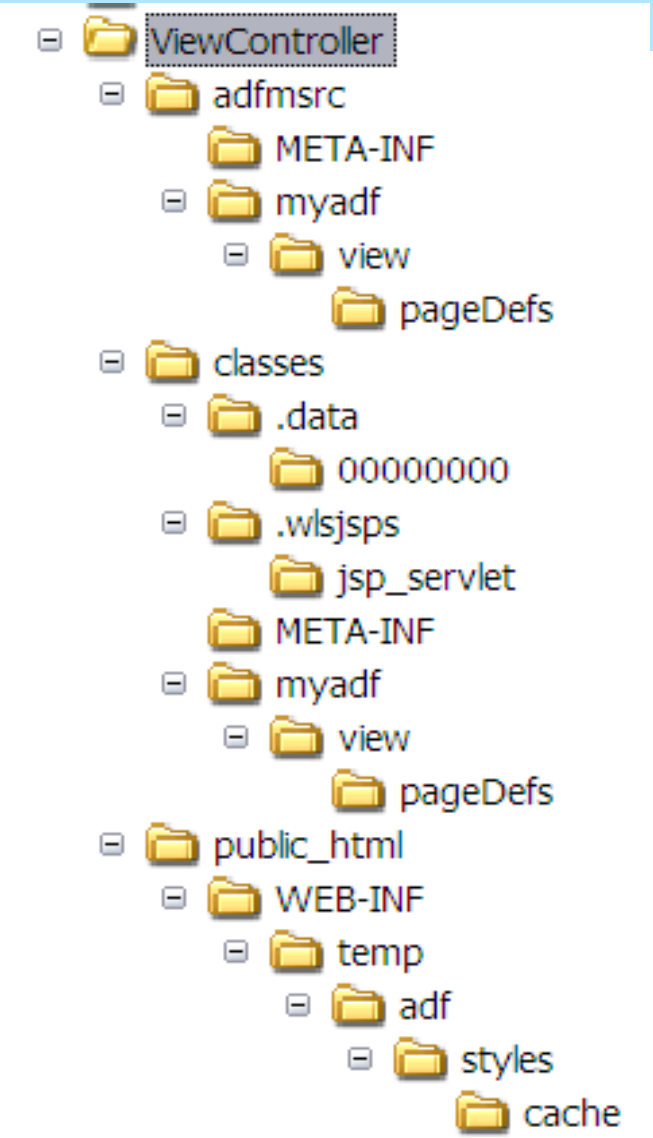
Find

```
1  <?xml version="1.0" encoding="UTF-8" ?>
2  <Application xmlns="http://xmlns.oracle.com/adfm/application"
3               version="11.1.1.54.7" id="DataBindings" SeparateXMLFiles="false"
4               Package="myadf.view" ClientType="Generic">
5    <pageMap>
6      <page path="/DeptEmpJSFPage.jspx"
7            usageId="myadf_view_DeptEmpJSFPagePageDef"/>
8    </pageMap>
9    <pageDefinitionUsages>
10     <page id="myadf_view_DeptEmpJSFPagePageDef"
11           path="myadf.view.pageDefs.DeptEmpJSFPagePageDef"/>
12   </pageDefinitionUsages>
13   <dataControlUsages>
14     <BC4JDataControl id="MyADFAppModuleDataControl" Package="myadf.model"
15                      FactoryClass="oracle.adf.model.bc4j.DataControlFactoryImpl
16                      SupportsTransactions="true" SupportsFindMode="true"
17                      SupportsRangesize="true" SupportsResetState="true"
18                      SupportsSortCollection="true"
19                      Configuration="MyADFAppModuleLocal" syncMode="Immediate"
20                      xmlns="http://xmlns.oracle.com/adfm/datacontrol"/>
21   </dataControlUsages>
22 </Application>
```

 Copyright @ 2011, John Jay King

# ADF Faces ViewController Files

- The XML files representing the ViewController project are distributed using a directory structure




 Copyright @ 2011, John Jay King

# ADF vs. Forms

| Feature | Forms | ADF |
|---|---|---|
| Declarative database access | Yes | Yes |
| Reuse of database access | Some | Yes |
| Declarative user interface development | Yes | Yes |
| Automatic screen generation | Yes | Some |
| Reuse of user interface | Some | Yes |
| Web Deployment | Yes | Yes |
| Client-Server Deployment | No | Yes |
| Fusion Applications development tool | No | Yes |
| Customizable | Yes | Yes |
| Built with open standards | No | Yes |

 Copyright @ 2011, John Jay King

# ADF Features Not Discussed

- ADF has many features; only some have been shown in this demo; others include:

- Graphics
- Complex Views
- List of Values (LOV)
- ADF task Flow
- AJAX/Partial Page Refresh
- JSF Navigation
- Transaction Control
- Layouts and Containers
- Special visual components
- Events and Listeners
- Generating Excel output
- Printing
- Tree Components & Tables
- ADF Reusability Features
- ADF Libraries
- Skinning
- Templates
- Debugging and Logging
- Security Issues
- WebLogic Management
- Application Deployment
- Custom Java Components

 Copyright @ 2011, John Jay King

# Available Books

- Quick Start Guide to Oracle Fusion Development
  - Grant Ronald
  - Oracle Press
- Oracle JDeveloper 11g Handbook
  - Duncan Mills, Peter Koletzke, Dr. Avrom Roy-Federman
  - Oracle Press
- Oracle Fusion Developer's Guide
  - Frank Nimphius, Lynn Munsinger
  - Oracle Press

 Copyright @ 2011, John Jay King

# Wrapping It Up

- Oracle's design emphasis and new features will support the Java-based ADF mechanism and enhance it for the foreseeable future
- JDeveloper and ADF allow creation of simple web applications easily:
  - ADF BC for data creates reusable components
  - ADF Faces for view creates reusable components
- Oracle Forms is not going anywhere; it is not necessary to "convert" things to ADF
- I did not write a single line of Java in this demo!

 Copyright @ 2011, John Jay King

# Need More on ADF?

- ODTUG's KScope 11 is coming June 26-29 in Long Beach, California

  Featuring an entire ADF track with top speakers from the user community and Oracle

  See http://kscope11.com/ for more

 Copyright @ 2011, John Jay King

# ODTUG Kscope11

LONG BEACH, CALIFORNIA * JUNE 26-30

## WHY SHOULD I ATTEND Kscope11?

Kscope11 offers the best content and most sessions on these topics than any other conference

## TOPICS AND SUB-TOPICS

**APPLICATION EXPRESS:** Infrastructure/Management, Plugins & Dynamic Actions, Security, Introduction, Integration & Migration, Core Functionality

**FUSION MIDDLEWARE:** SOA Suite including BPM and OSB, ADF, WebLogic, WebCenter

**DATABASE DEVELOPMENT:** Design/Data Modeling, Coding, Maintenance, Best Practices

**BI and EPM:** Financial Management Solutions, Planning Solutions, Essbase Solutions, OBIEE Administration, Data Warehousing, Reporting Solutions, Keeping It All Running

**MySQL:** Architecture, Performance Tuning & Optimization, Development & Coding, Best Practices, Case Studies

## CONFERENCE HIGHLIGHTS

- 250+ Sessions
- Hands-on Training
- All-day Symposiums
- Product Updates
- Lunch & Learn
- Vendor Showcase
- One-on-one time with Oracle Experts
- Networking events everyday

GET THE LATEST UPDATES BY FOLLOWING US > > > > > > > > > > > > > > > > > >

# Need Effective Training?

- **Learn new skills quickly and cost-effectively:**
  - ❑ Instructor-led on-site  Best classroom training, at ***your*** site
  - ❑ Instructor-led via web  Instructor-led, delivered on-line
- **We provide many other training services; please see our website for more information:**

  **www.kingtraining.com**

 Copyright @ 2011, John Jay King

# Instructor-Led at it's Best!

- **Outstanding Presentation** — Instructors provide lively, engaging, presentations.
- **Instructors Who Lead** — Non-threatening learning environment; instructors coach.
- **Up-to-date** — All courses current to the latest releases. (We also offer down-level training too…)
- **First-Class Materials** — Course books are complete with many illustrations and indexes. Past students refer to them frequently and many co-workers borrow them too.
- **Hands-On Exercises** — Most courses use lots of hands-on allowing students to feel immediate success.

 Copyright @ 2011, John Jay King

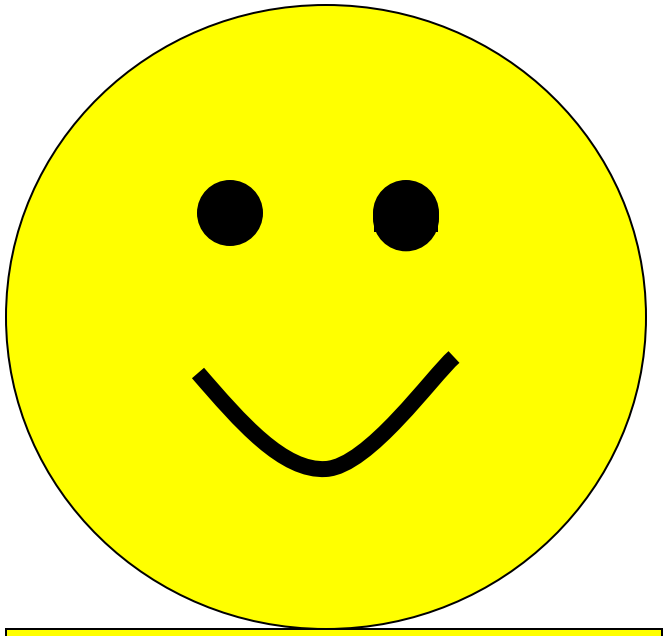
# Ask Us!

- King Training Resources strives to be easy to work with; we can be flexible in many different ways; ask us!
  - **Customization**
    - We often customize course outlines and topics to meet client needs.
    - Exercises are frequently customized to meet the standards and practices of the client organization; students are ready to "hit the ground running" when they return to work.
    - Most customization is accomplished at no cost!
  - **Pre and Post Assessments**
    - King Training Resources provides pre-assessments and post-assessments of student knowledge in the skills covered.
    - We provide a recap of pre-and-post comparisons after the course is complete.
- Contact: **Peggy King - peggy@kingtraining.com**

**1.303.798.5727 – 1.800.252.0652**

 Copyright @ 2011, John Jay King

# Thank you for your interest and attention!



**Thanks for your attention!**

Today's presentation is on the web! http://www.kingtraining.com

John King's email: john@kingtraining.com