

Just Get it Written: Deploying Applications to WebLogic Server Using JDeveloper and WLS Console

Peter Koletzke

Technical Director &
Principal Instructor

(coauthored by Duncan Mills, Oracle)

QUOVERA



Moral

Don't get it right,
just get it written.

—James Thurber,
*The Sheep in Wolf's Clothing;
Fables for our Time*

QUOVERA

2

Survey

- Job responsibilities?
 - DBA, developer
- Platforms
 - OAS – OC4J
 - WLS
 - Other
- Tools?
 - Eclipse
 - JDeveloper
 - Other



QUOVERA

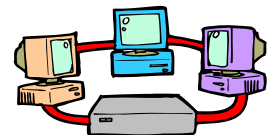
3

Agenda

- Deployment Concepts
- JDeveloper Support
- Deploying from WLS Console

Slides will be on the
NYOUG website.
White paper will be in
the tech journal.

The white
paper contains
hands-on
practices.



QUOVERA

4

Deployment Concepts

- Deployment: copying and installing application files to a runtime environment
 - Development, testing, QA, production
 - Occurs multiple times for an app
 - Bug fixes, enhancements, software updates
- You may be responsible
 - So you, as a developer, need to know this
- Someone else may be responsible
 - So you need to provide the right files to deploy
 - Testing those files beforehand is useful



Preparation for Deployment

- Development and testing complete
- Required libraries and supporting files are identified
- Application server is available
 - Has an appropriate web container (JVM) process
 - For example, WebLogic Server
 - You know the server administration password
 - Or know someone who knows
 - Needed to copy files to the server



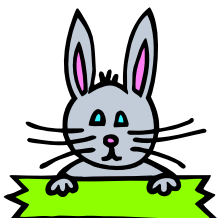
Deployment Objectives

- Ensure that all required libraries are available
- Ensure that library versions match between development and deployment
- All necessary JDeveloper projects must be deployed
 - Model
 - ViewController
 - ADF Libraries (usually bundled into the EAR)
- Files must be archived into specially formatted Java EE JAR files
 - WAR or
 - EAR (if more than one application)
 - ADF Library JARs

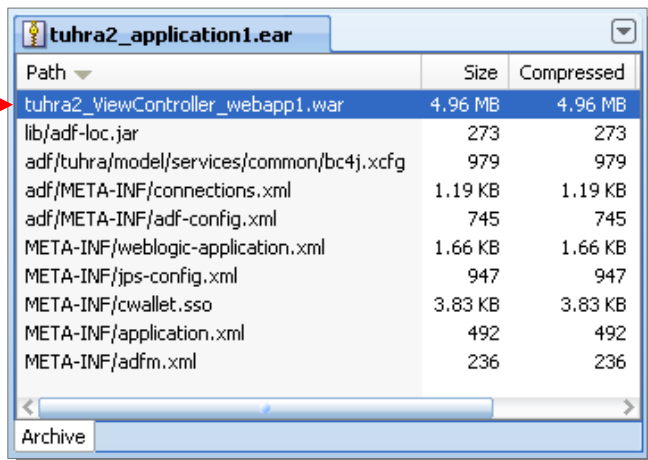


Enterprise Application Archive (EAR)

- Contains one or more WAR (application) files and other libraries that are required
- A file you deploy
- Copies to the app server directory: `FMW_HOME>\user_projects\domains`
 - App server processes new EAR files when restarting
 - Can also process EAR files without restarting
- The app server expands this into its component WAR files
 - Each WAR file goes into a directory corresponding to its name
 - Then it expands the WAR files



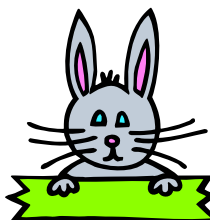
Sample EAR File



Path	Size	Compressed
tuhra2_ViewController_webapp1.war	4.96 MB	4.96 MB
lib/adf-loc.jar	273	273
adf/tuhra/model/services/common/bc4j.xcfg	979	979
adf/META-INF/connections.xml	1.19 KB	1.19 KB
adf/META-INF/adf-config.xml	745	745
META-INF/weblogic-application.xml	1.66 KB	1.66 KB
META-INF/jps-config.xml	947	947
META-INF/cwallet.sso	3.83 KB	3.83 KB
META-INF/application.xml	492	492
META-INF/adfm.xml	236	236

JDeveloper
Archive
Viewer

- One app (WAR) file in this case
- Several XML configuration files



QUOVERA

9

Web Application Archive (WAR)

- Also called, *Web Archive*
- Can alternatively be deployed without an EAR
- Contains all files required for an application
- Contains files or directories inside WEB-INF (the main runtime directory)
- Copied to server and expanded into separate files by the server
 - Inside a directory named the same as the WAR file

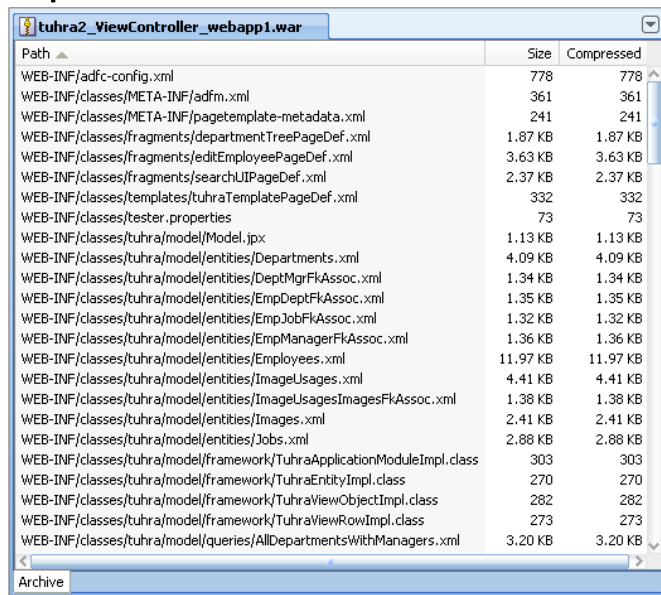


QUOVERA

10

Sample WAR File

- App server extracts files into the `../<app>/` directory
 - `<app>` is the name of the WAR file



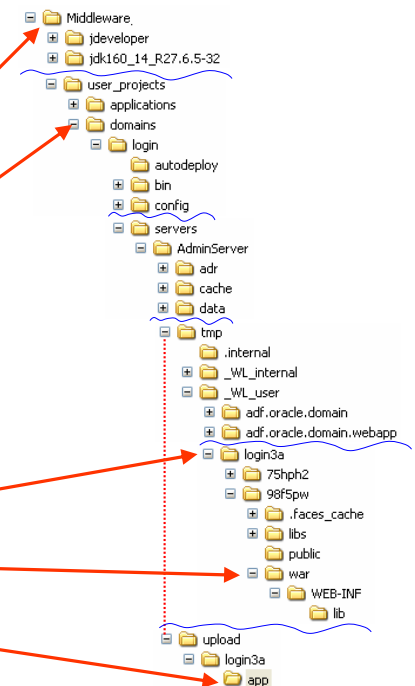
Path	Size	Compressed
WEB-INF/adf-config.xml	778	778
WEB-INF/classes/META-INF/adfm.xml	361	361
WEB-INF/classes/META-INF/pagetemplate-metadata.xml	241	241
WEB-INF/classes/fragments/departmentTreePageDef.xml	1.87 KB	1.87 KB
WEB-INF/classes/fragments/editEmployeePageDef.xml	3.63 KB	3.63 KB
WEB-INF/classes/fragments/searchUIPageDef.xml	2.37 KB	2.37 KB
WEB-INF/classes/templates/tuhraTemplatePageDef.xml	332	332
WEB-INF/classes/tester.properties	73	73
WEB-INF/classes/tuhra/model/Model.jpx	1.13 KB	1.13 KB
WEB-INF/classes/tuhra/model/entities/Departments.xml	4.09 KB	4.09 KB
WEB-INF/classes/tuhra/model/entities/DeptMgrFkAssoc.xml	1.34 KB	1.34 KB
WEB-INF/classes/tuhra/model/entities/EmpDeptFkAssoc.xml	1.35 KB	1.35 KB
WEB-INF/classes/tuhra/model/entities/EmpJobFkAssoc.xml	1.32 KB	1.32 KB
WEB-INF/classes/tuhra/model/entities/EmpManagerFkAssoc.xml	1.36 KB	1.36 KB
WEB-INF/classes/tuhra/model/entities/Employees.xml	11.97 KB	11.97 KB
WEB-INF/classes/tuhra/model/entities/ImageUsages.xml	4.41 KB	4.41 KB
WEB-INF/classes/tuhra/model/entities/ImageUsagesImagesFkAssoc.xml	1.38 KB	1.38 KB
WEB-INF/classes/tuhra/model/entities/Images.xml	2.41 KB	2.41 KB
WEB-INF/classes/tuhra/model/entities/Jobs.xml	2.88 KB	2.88 KB
WEB-INF/classes/tuhra/model/framework/TuhraApplicationModuleImpl.class	303	303
WEB-INF/classes/tuhra/model/framework/TuhraEntityImpl.class	270	270
WEB-INF/classes/tuhra/model/framework/TuhraViewObjectImpl.class	282	282
WEB-INF/classes/tuhra/model/framework/TuhraViewRowImpl.class	273	273
WEB-INF/classes/tuhra/model/queries/AllDepartmentsWithManagers.xml	3.20 KB	3.20 KB

QUOVERA

11

Deployed Directory Structures

- FMW_Home
 - Fusion Middleware
- Domain
 - Created using the Configuration Wizard
- WAR file and app files under this level
 - App root here
- EAR file here

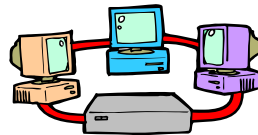


QUOVERA

12

Agenda

- Deployment Concepts
- JDeveloper Support
- Deploying from WLS Console



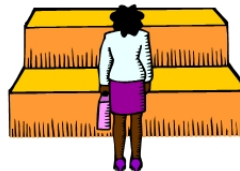
Yes, Let's

However beautiful the strategy,
you should occasionally
look at the results.

—Winston Churchill (1874-1965)

Two Big Steps

1. Prepare the WAR and EAR deployment profiles
 - Specify what to deploy
 - Application and ADF BC projects
 - Required libraries
2. Deploy using the deployment profile
 - This creates the WAR and EAR
 - JDeveloper copies them to the server



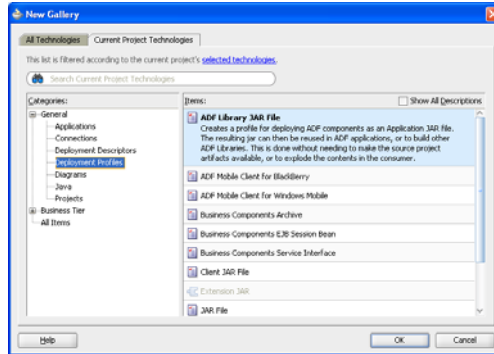
Testing in Standalone WebLogic

- Between running the app in JDeveloper and deploying to the server run this test
- You need to set up the WLS standalone instance
 - WLS is part of Oracle Fusion Middleware
 - Included with JDeveloper for standalone
- Ensure that the versions are compatible
 - WLS version
 - Application development runtime

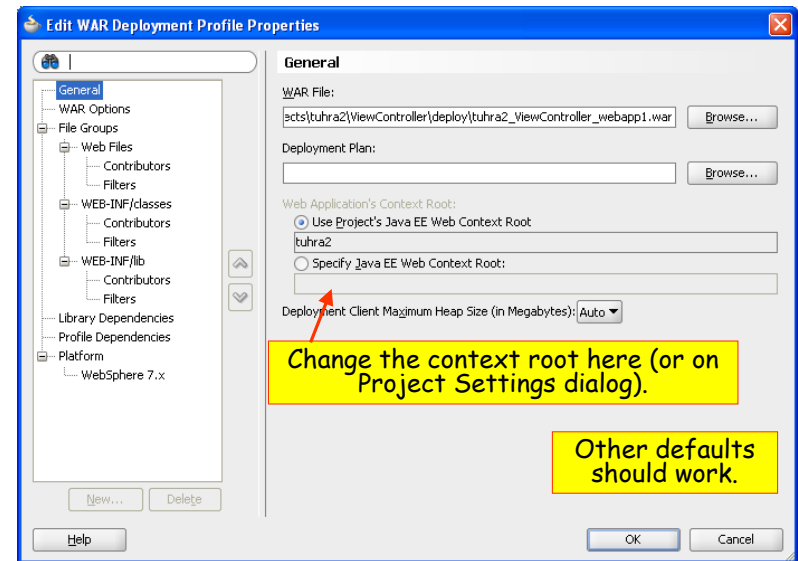


Deployment Profiles

- Embedded in the .jws and .jpr files
- Access from Application / Project properties, or from the New Gallery (context sensitive)
- Default EAR and WAR profiles are pre-created with the Fusion App template
- Create separate deployment profiles for separate purposes – multiple targets, etc.

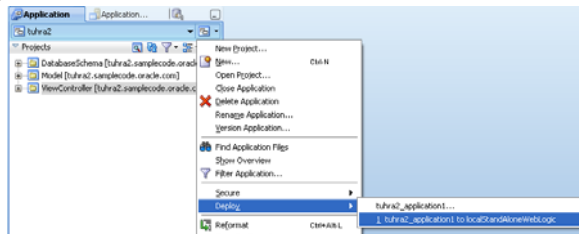


Deployment Profile Wizard



Deploying From JDeveloper

- Assumes WLS standalone has been set up and is running and the JDeveloper app server connection is set up
 - See hands-on practice for those steps
- **Application menu | Application Properties | Edit**
 - Application Assembly
 - Check Model
- **Application menu | Deploy | <app>**
 - Local App Server

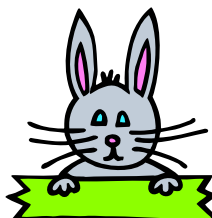


What Next?

- Wait and watch
 - JDeveloper builds the WAR and EAR files in the project directory

```
[02:42:45 PM] Wrote Web Application Module to C:\projects\tuhra2\ViewController\deploy\tuhra2_ViewController_webapp1.war
[02:42:47 PM] Uploading jaxn-data users.
[02:42:47 PM] Uploading jaxn-data roles.
[02:42:48 PM] Wrote Enterprise Application Module to C:\projects\tuhra2\deploy\tuhra2_application1.ear
[02:42:48 PM] Deploying Application...
```

- Then it copies the EAR file to the server instance
- Message log window will show progress



Deployment Log Window

```

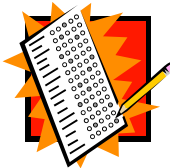
[02:42:14 PM] ---- Deployment started. ----
[02:42:14 PM] Target platform is (Weblogic 10.3).
[02:42:14 PM] Retrieving existing application information
[02:42:16 PM] Running dependency analysis...
[02:42:16 PM] Building...
[02:42:38 PM] Deploying 2 profiles...
[02:42:45 PM] Wrote Web Application Module to C:\projects\tuhra2\deploy\tuhra2_ViewController_webapp1.war
[02:42:47 PM] Uploading j2ee-data users.
[02:42:47 PM] Uploading j2ee-data roles.
[02:42:48 PM] Wrote Enterprise Application Module to C:\projects\tuhra2\deploy\tuhra2_application1.ear
[02:42:48 PM] Deploying Application...
[02:43:04 PM] [Deployer:149192]Operation 'deploy' on application 'tuhra2_application1 [Version=V2.0]' is in progress on 'AdminServer'
[02:43:04 PM] [Deployer:149192]Operation 'deploy' on application 'tuhra2_application1 [Version=V2.0]' is in progress on 'AdminServer'
[02:43:09 PM] [Deployer:149194]Operation 'deploy' on application 'tuhra2_application1 [Version=V2.0]' has succeeded on 'AdminServer'
[02:43:09 PM] [Deployer:149194]Operation 'deploy' on application 'tuhra2_application1 [Version=V2.0]' has succeeded on 'AdminServer'
[02:43:09 PM] Application Deployed Successfully.
[02:43:09 PM] Elapsed time for deployment: 56 seconds
[02:43:09 PM] ---- Deployment finished. ----
    
```

- The Log window usually appears in the bottom of the JDeveloper IDE window

Test the Application

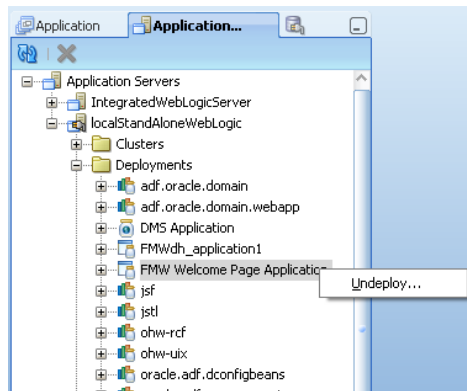
- Close JDeveloper if you are skeptical
- In browser, run the application URL
 - In this case:
http://localhost:7001/tuhra2
 - This opens the default page
“welcome file” defined in web.xml

- Test the app
- Close the browser



Application Server Navigator

- Lists available servers
- Shows deployed apps
- Allows new server creation
- You can act on these using JDeveloper
 - Undeploy
 - Start Admin Console for server



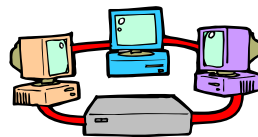
The Best of Both Worlds

- OJDeploy
 - ‘Headless’ JDeveloper
 - Command line utility
 - Packaged with JDeveloper 11g
 - Search for ojdeploy in the JDeveloper Help Center
 - Allows automation of the pre-defined deployment profiles
 - Also runnable from ANT
- Manages
 - Packaging and deployment
 - JDBC data sources
 - Security permissions propagation
 - Deploying users and roles for testing



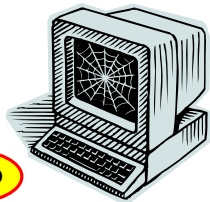
Agenda

- Deployment Concepts
- JDeveloper Support
- Deploying from WLS Console



Deploying From WLS Console

- The app server is usually under the control of the app server administrator
 - (Be kind to your app server administrator)
- You need to package the EAR file and let them know where it is
 - Select “Deploy to <app name>” from the Application menu in the navigator
- The administrator will deploy the EAR file to a new or existing application node
 - Setting up the application is a one-time task
 - Deploying the app after the setup is easier than before the setup

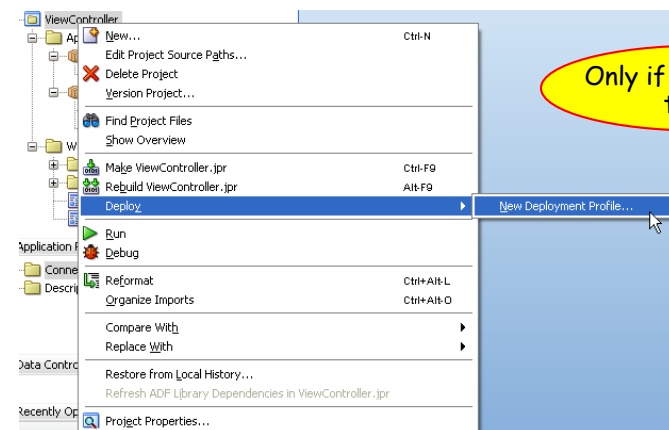


Demo

Configure the App Server

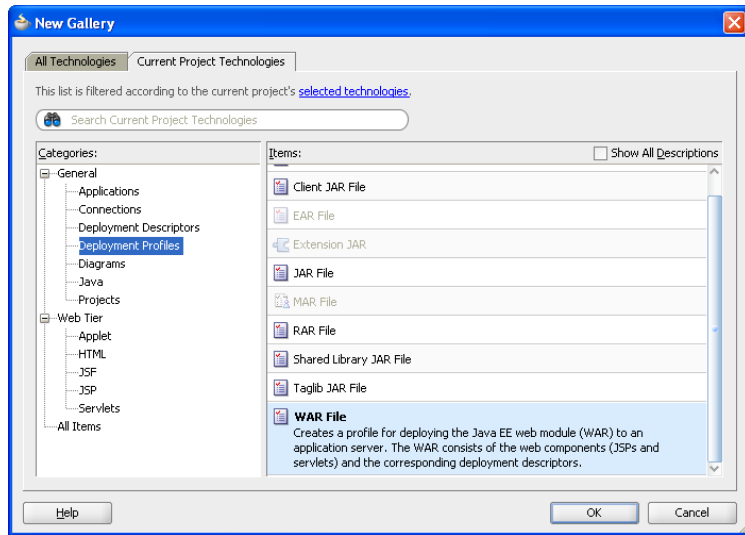
- Results in a standalone WLS server
- Windows Start menu
 - Start | Programs | Oracle Fusion Middleware 11.1.1.x | WebLogic Server 11g | Tools | Configuration Wizard
- Follow the steps in the white paper
 - The following assumes this is complete and the app server is started
- Note: Use the following as a guide for deploying with JDev after 11.1.1.1

Create a Deployment Profile on the ViewController Project



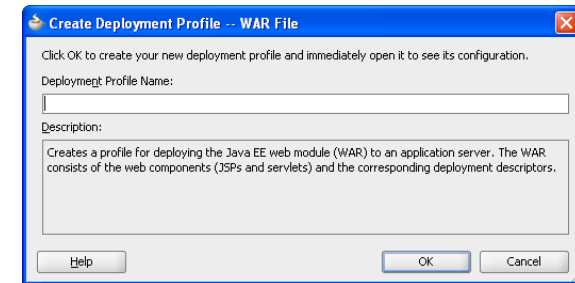
- Also accessible in the Deployment page of Project Properties

New Gallery



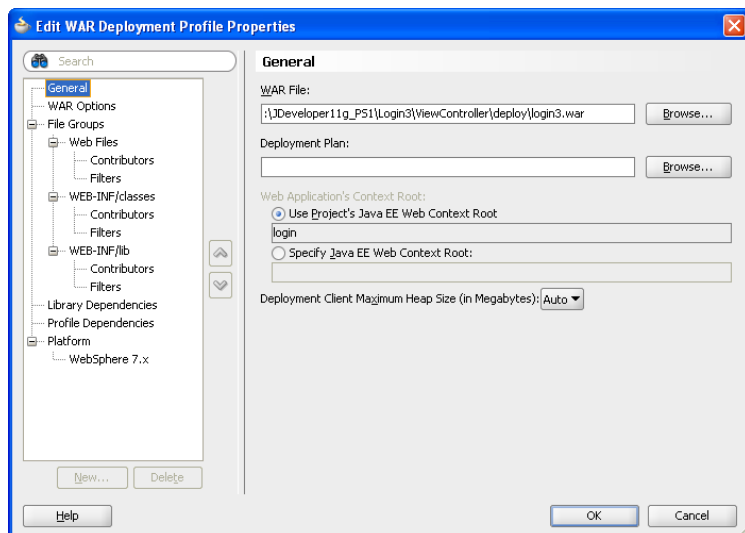
- Double click “WAR File”

Create WAR Deployment Profile



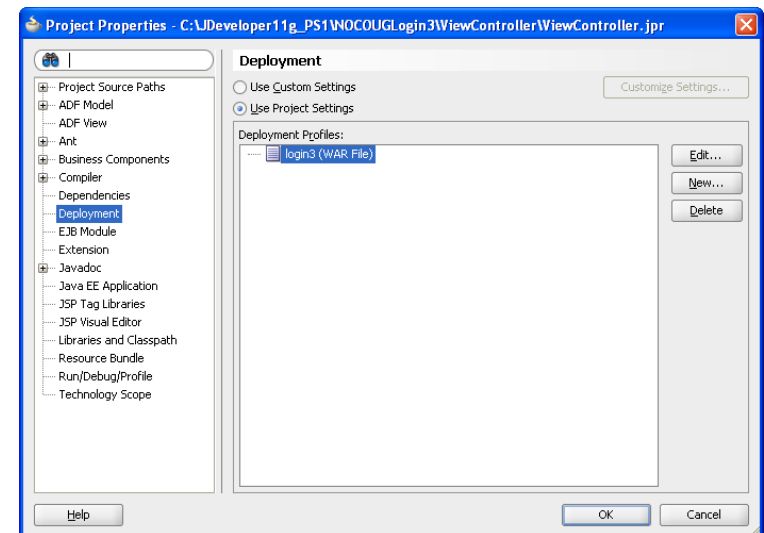
- Enter a name and click OK

Deployment Profile Properties



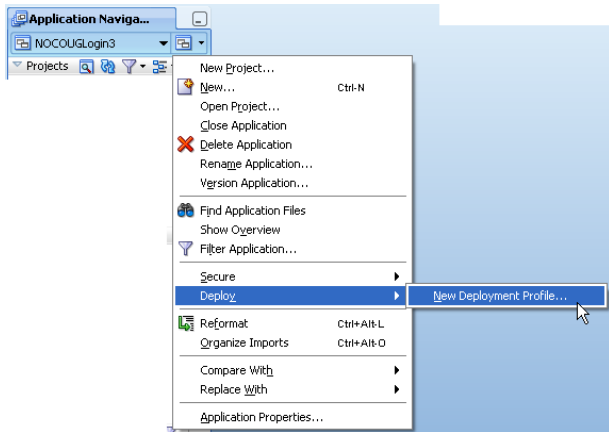
- Defaults are OK

Project Deployment Page



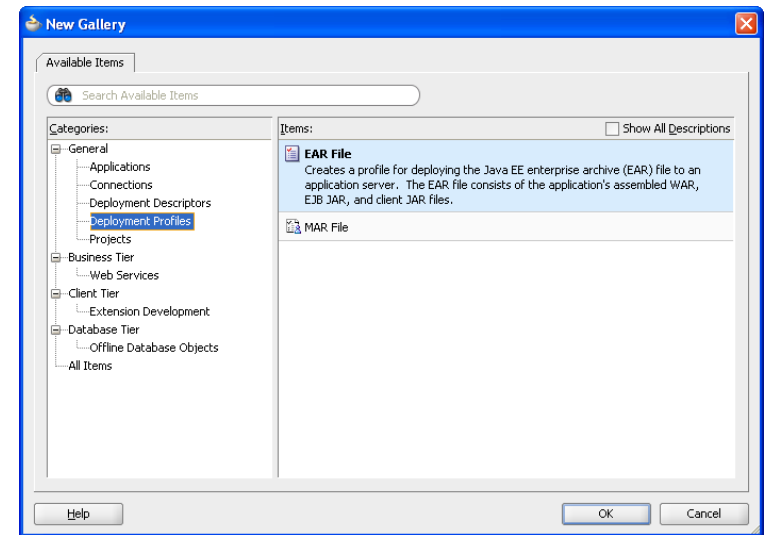
- Defaults are OK

Application Menu - Deploy



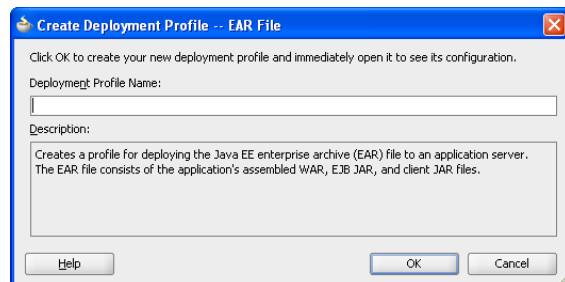
- First time in app, no deployment profile

New Gallery - EAR File



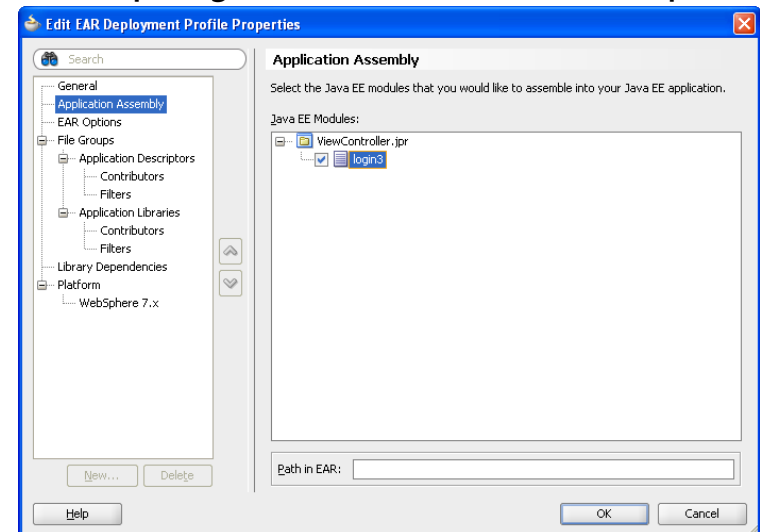
- Double click "EAR File"

Create EAR Deployment Profile



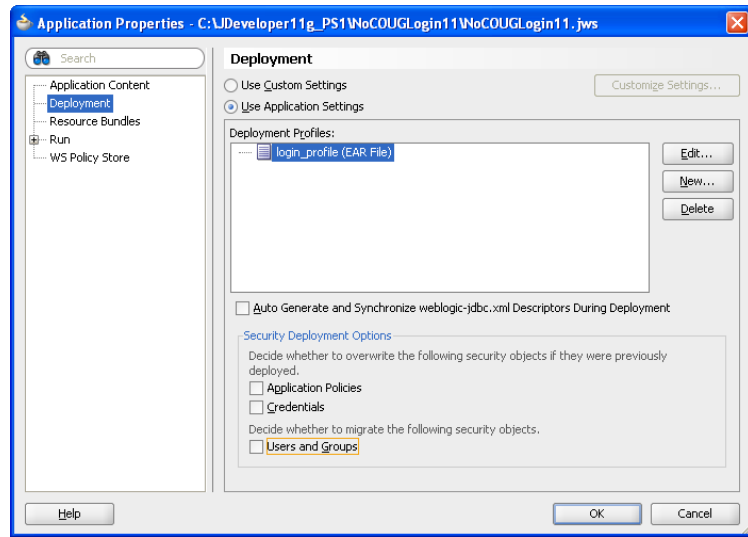
- Enter the name and click OK

EAR Deployment Profile Properties



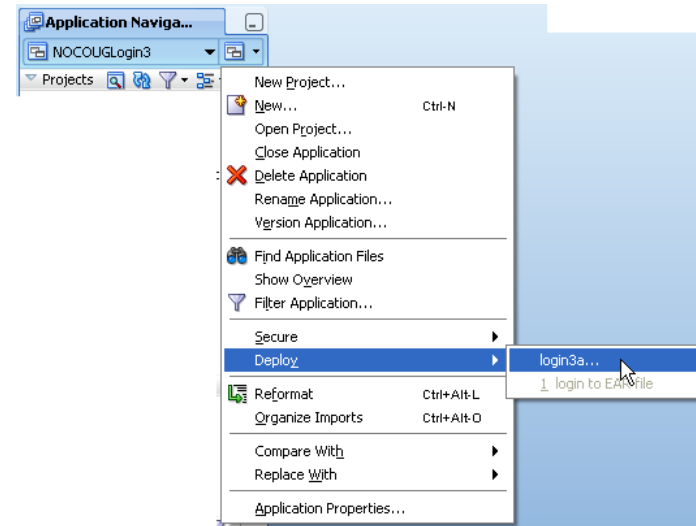
- Application Assembly Page
 - Select WAR deployment profile

Application Deployment Properties

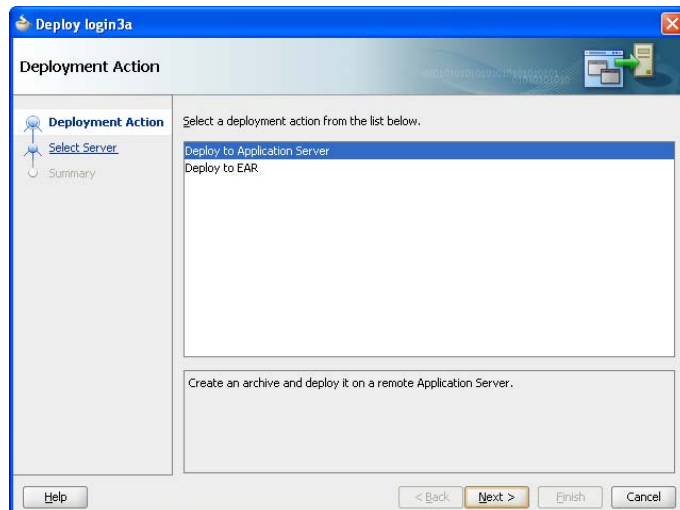


- You can visit the Deployment page later in the Application menu (Application Properties)

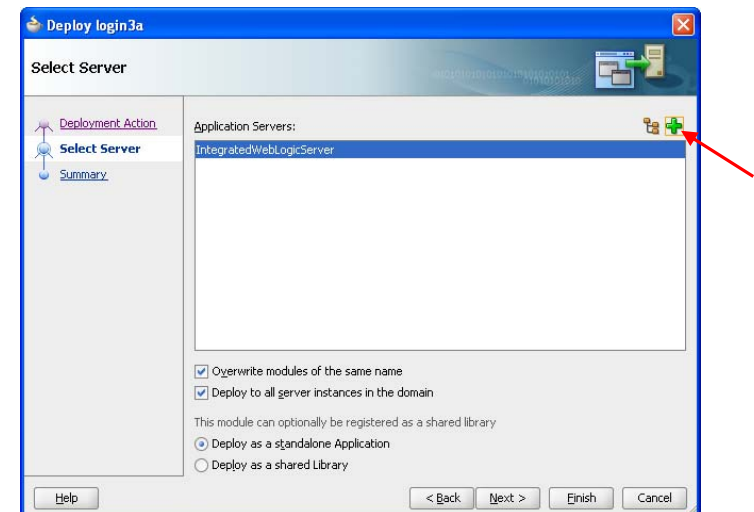
Application Menu - Deploy EAR



Deploy Application - Action

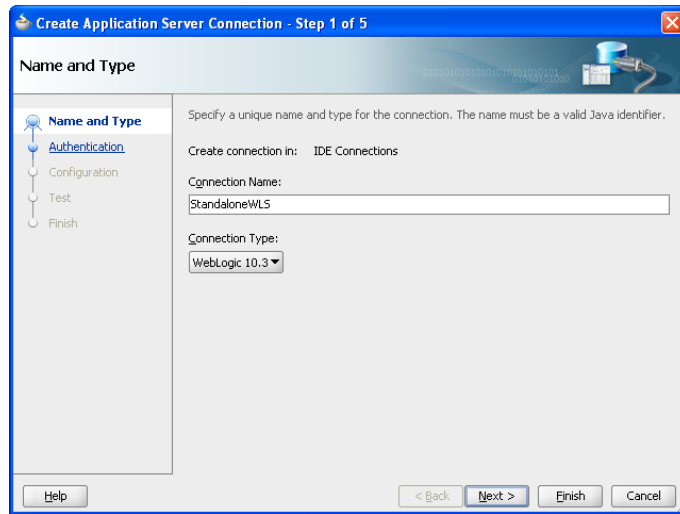


Deploy to Server



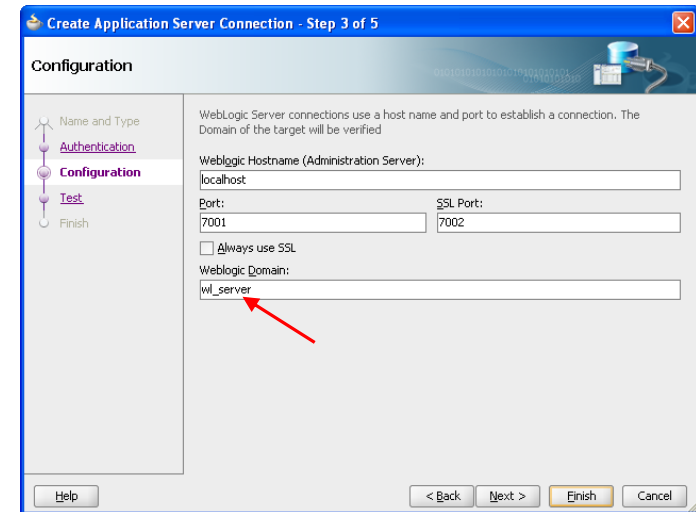
- Click "+" (green plus) to add a connection

Define the Server Connection



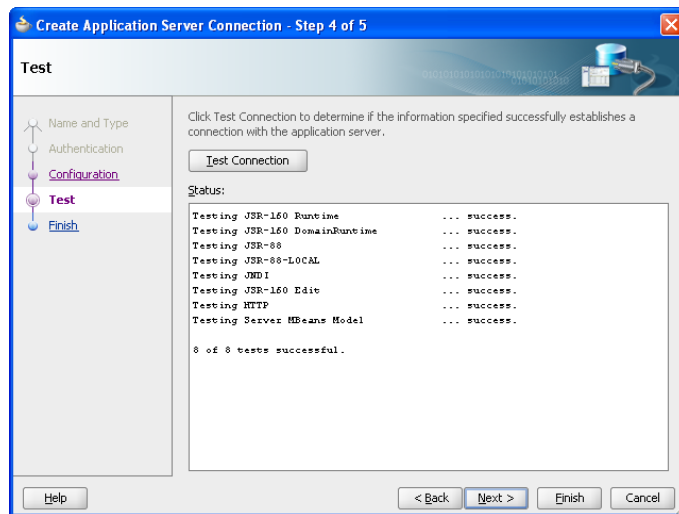
- Step 2 - password
 - From the Configuration Wizard

Server Details



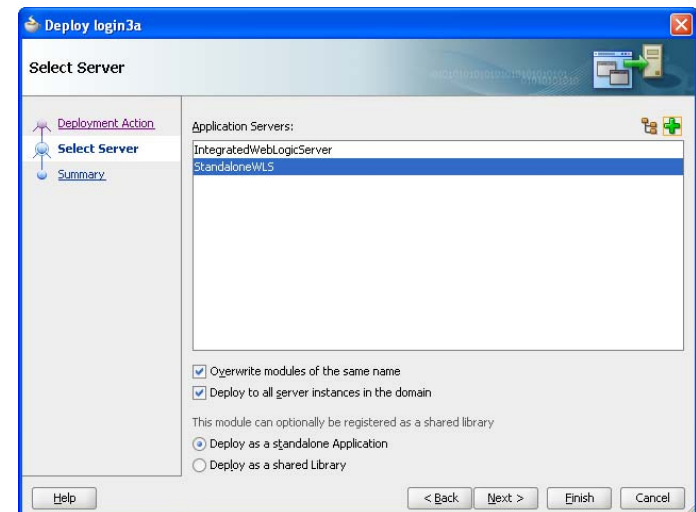
- Weblogic Domain from the Configuration Wizard

Test



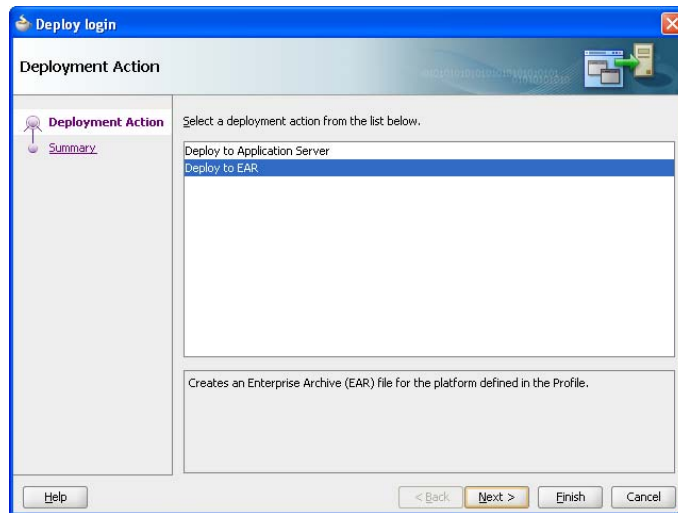
- Be sure the server is running
- Finish

Select the New Connection



- Finish & Test

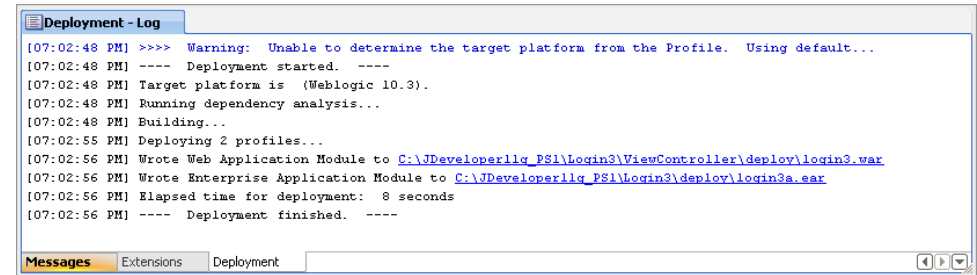
Deploy Application - Action



QUOVERA

45

Log Window

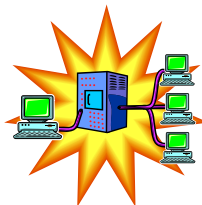


QUOVERA

46

Then Deploy in the App Server

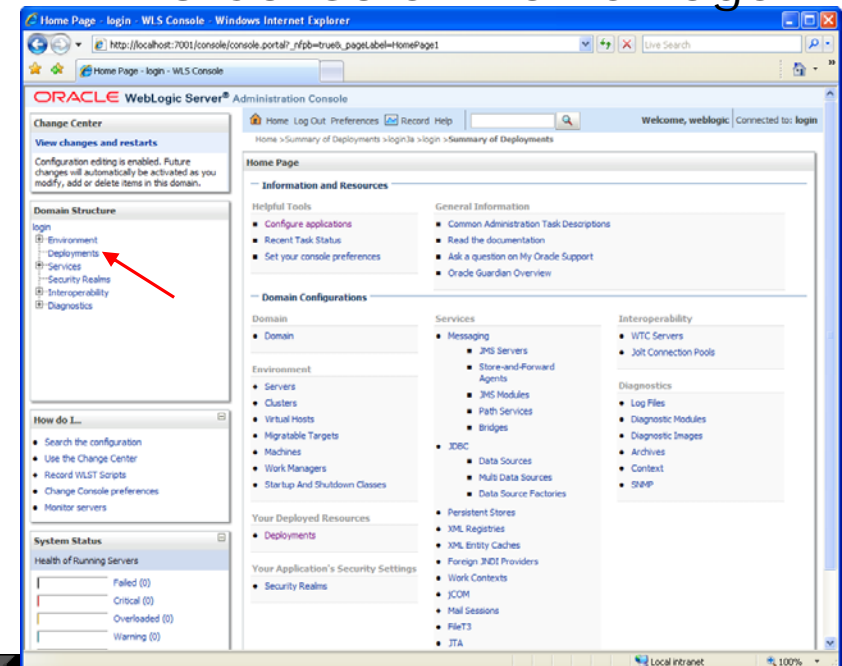
- You can invoke the console from
 - A URL: <http://localhost:7001/console>
 - Or the Application Server Navigator
- Basic steps:
 - Drill into “Deployments”
 - Click the “Install” button
 - Find EAR file in file system and go!
- Additional common tasks
 - Define data sources
 - Deploy permissions



QUOVERA

47

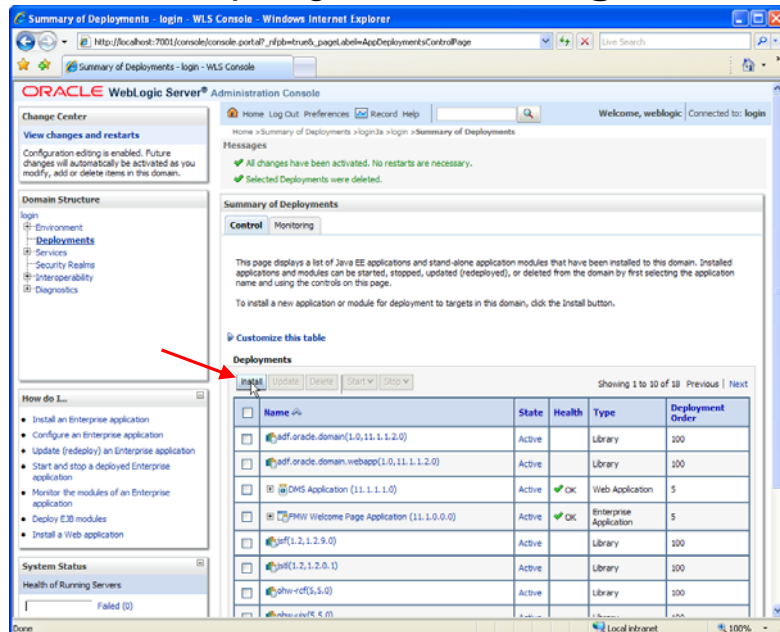
WLS Console - Home Page



QUOVERA

48

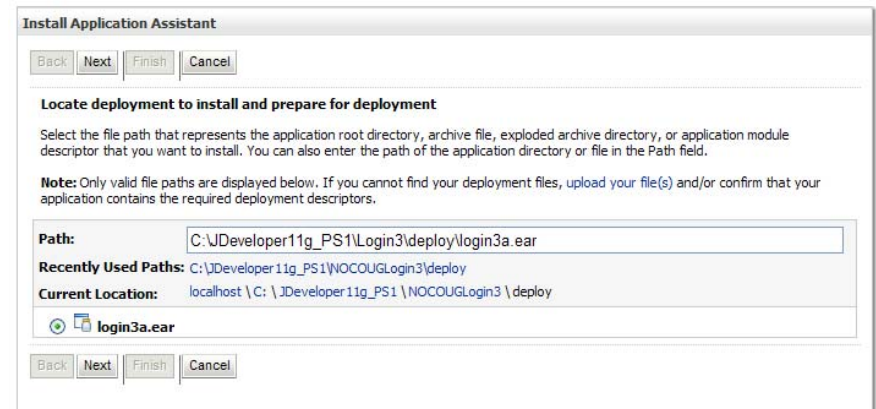
Deployments Page



QUOVERA

49

Follow the Screens

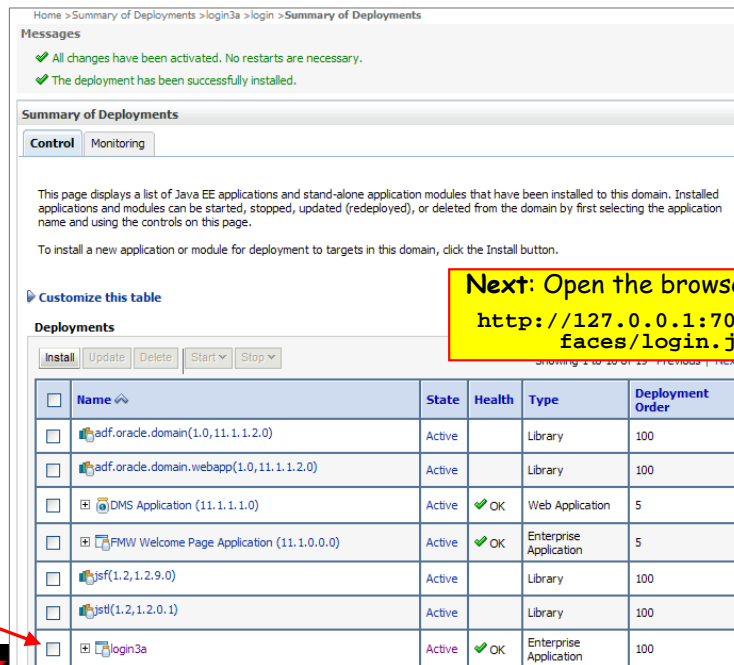


- Find the file in the file system
 - Select it. Click Next

QUOVERA

50

Success!



Next: Open the browser and test.
<http://127.0.0.1:7001/login/faces/login.jspx>

QUOVERA

51

We're Not Done Yet?

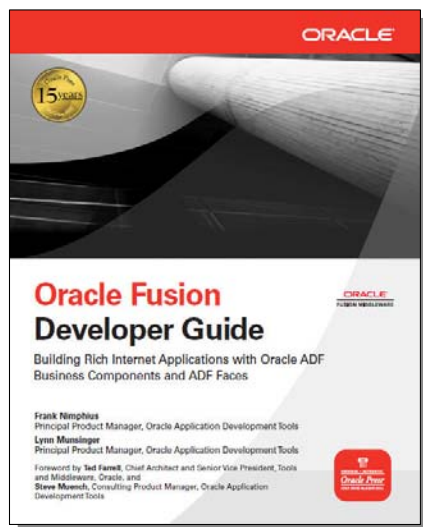
This is not the end.
 It is not even the beginning
 of the end.
 But it is, perhaps,
 the end of the beginning.
 — Winston Churchill (1874-1965)

QUOVERA

52

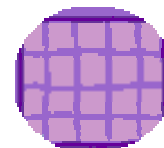
Other Resources

- *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework 11g Release 1 (11.1.1)*
 - PDF at OTN, online in JDev
 - Chapter 32
- *Oracle Fusion Middleware Security Guide 11g Release 1*
 - PDF at OTN
- *Oracle Fusion Developer Guide*
 - Nimphius and Munsinger, McGraw-Hill Professional, Oracle Press (2010)
 - Brief mention in Chapter 21



Summary

- Java EE web applications are deployed to the app server using EAR files
- EAR files contain WAR files
- WAR files contain application files and all required libraries
- JDeveloper's deployment profiles help gather the files (project level and application level)
- Run a utility to create the EAR file and deploy to the app server
- Alternatively, use WLS Console
- Testing deployments is a best practice



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



<http://www.quovera.com>

- Founded in 1995 as Millennia Vision Corp.
- Profitable 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

