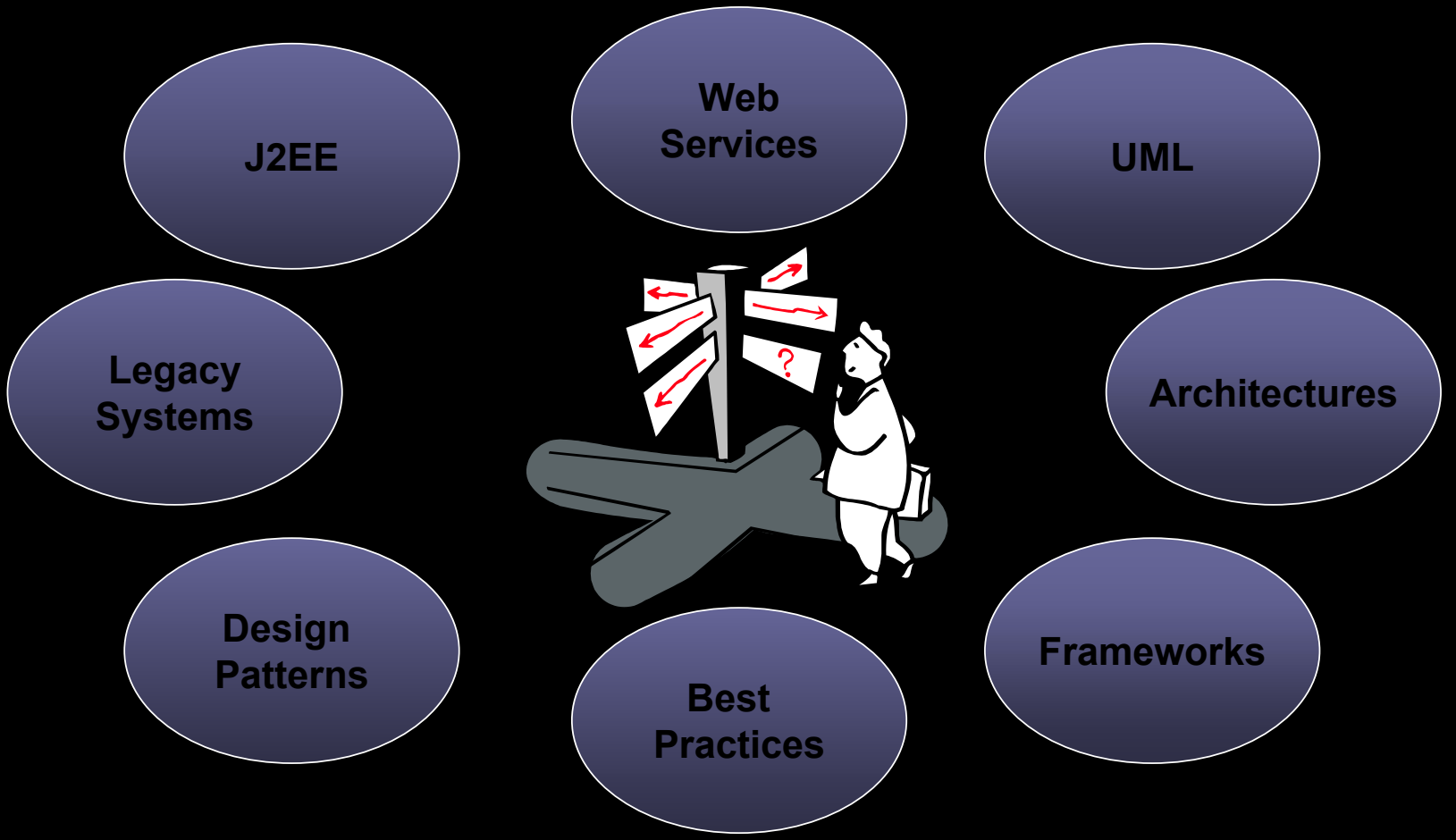
Tool Support for Developing Scalable J2EE Web Service Architectures

Guus Ramackers
Application Development Tools
Oracle Corporation

guus.ramackers@oracle.com
www.oracle.com

ORACLE

Using All This in Real Life …



Confused?

Compare with:

Elevator Sign in Hyatt Burlingame:

In the event of a malfunction :

1. Do not become alarmed
2. Press the red button marked “alarm”

Why web services?

- Enabling Virtual Organizations
  - Eliminating the “stove pipe syndrone” within a company
    - E.g. telecoms company with .Net and J2EE focused divisions (based on pre-takeover culture)
  - Cross company collaborative business processes
    - E.g. doing what Dell does, without dictating implementation
- Ubiquitous Enterprise Application Integration
  - “Cheap and cheerful”, not heavyweight
  - Utilize the internet for inter-connection
  - No more “screen scraping”

Start from Scratch?

- Web Service development does *not* require
  - A “whole new way of developing”
  - A “whole new architectural paradigm”
  - A “whole new component model”
- Web Services are simply a different way of exposing an *interface* to existing architectures
  - “WSDL is the IDL for the internet”
  - As with IDL, care must be taken to design the right interfaces
  - A reusable architecture is a prerequisite

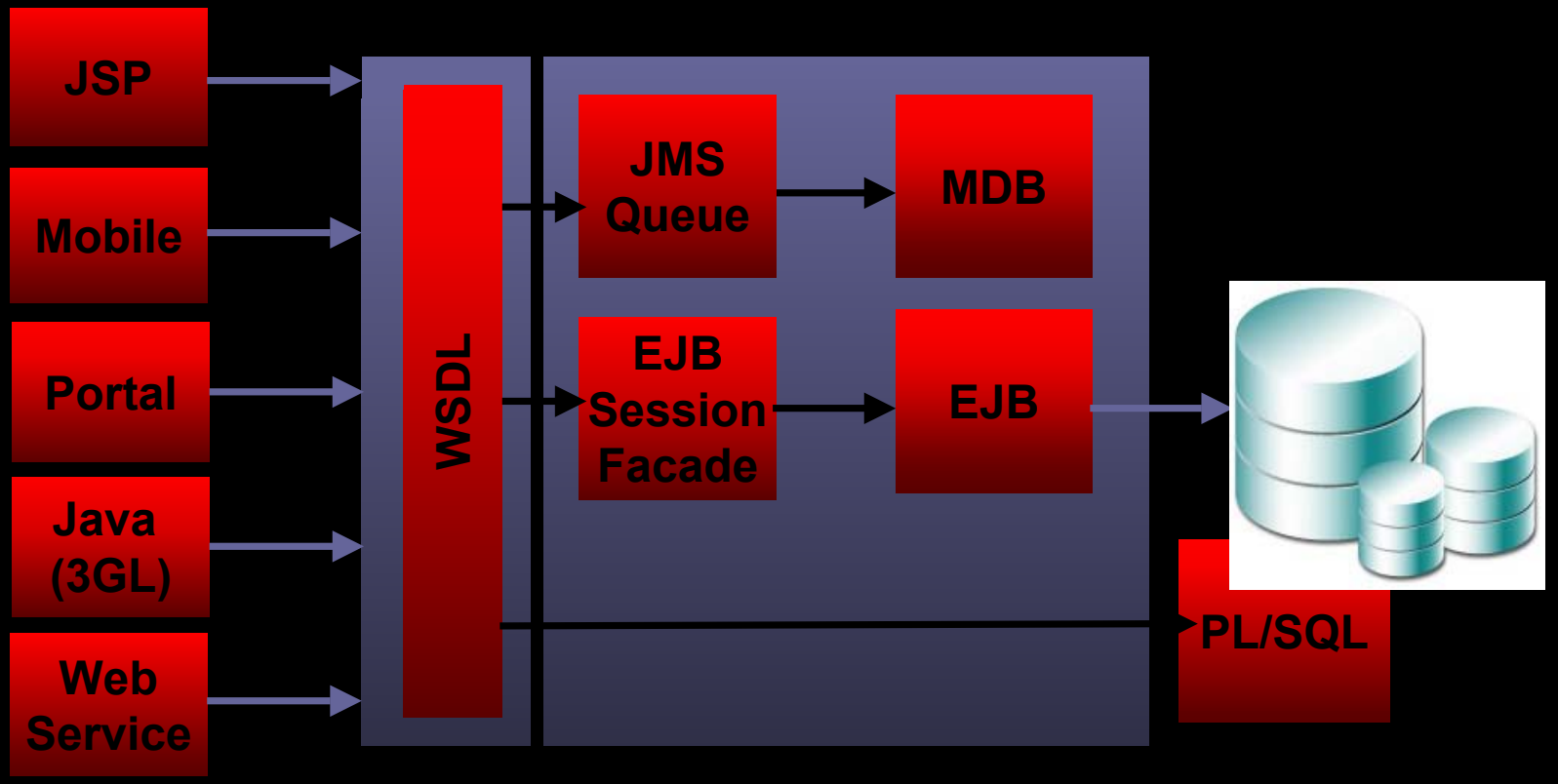
End-to-end J2EE Application Architecture

- Component Standards
  - Container provides standardized run-time, e.g. EJB, Servlets
- Design Patterns
  - Solve common design issues & provide common vocabulary
  - E.g. Session Façade, Value Object, Paged Lists
- Development Frameworks
  - Optimize interaction between multiple patterns
  - Focused on client, web, server, or database tier
  - E.g. Struts, Oracle BC4J
- Application Architecture
  - Combine frameworks to form end-to-end application architecture
  - Typically *Model-View-Controller* architecture in J2EE context
  - E.g. JSP + Struts + BC4J + JDBC
  - Reuse the Model (e.g. Mobile + WS + BC4J + JDBC)

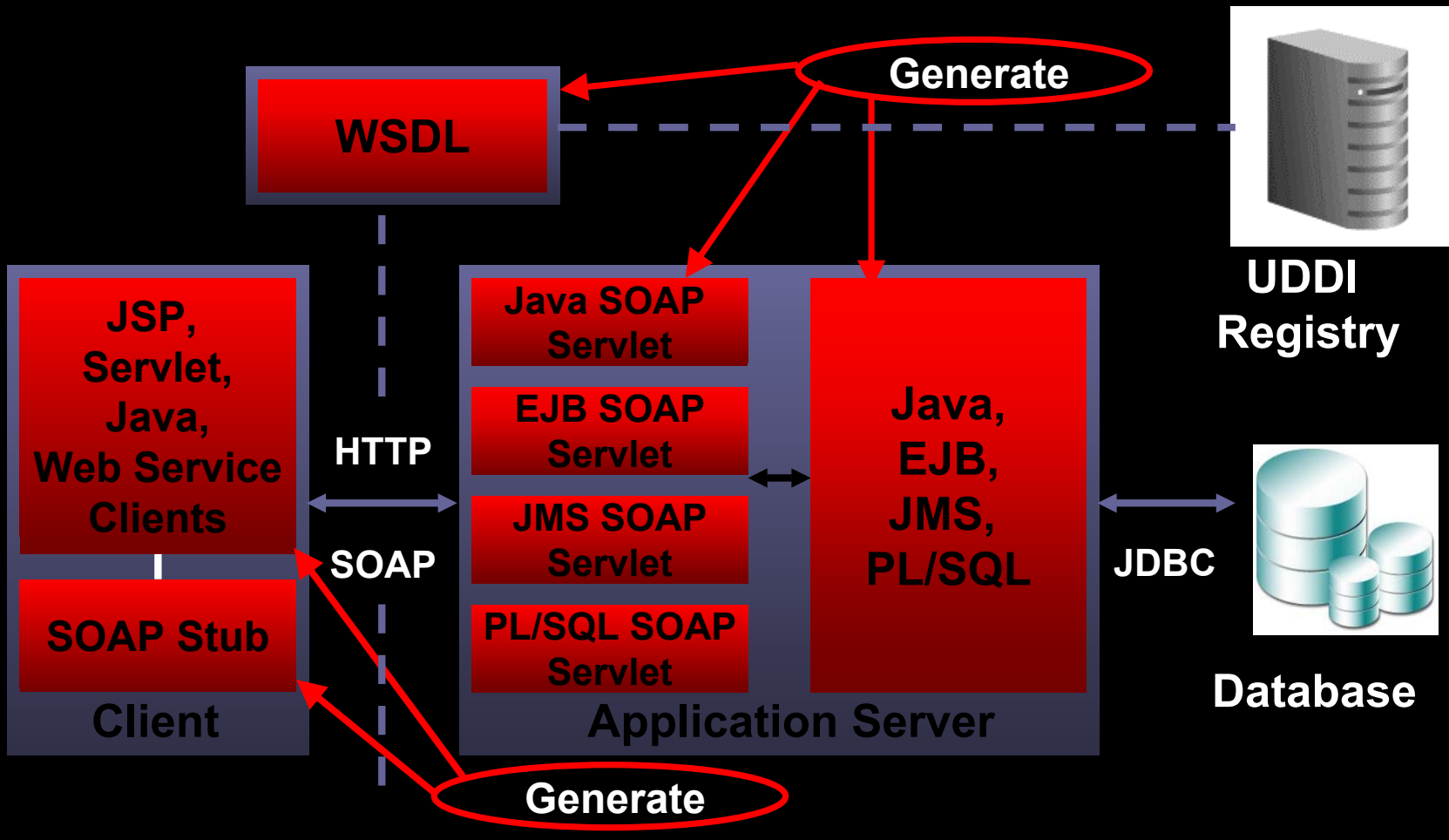
Scalability of WS Interfaces

- Performance of the protocol
  - SOAP versus RMI
- Binding between systems
  - Enterprise systems can't wait for each other
    - Asynchronous client
    - Asynchronous transport
    - Asynchronous server
- Granularity of requests
  - Batch up records
  - Batch up changes

Expose Coarse Grain Services

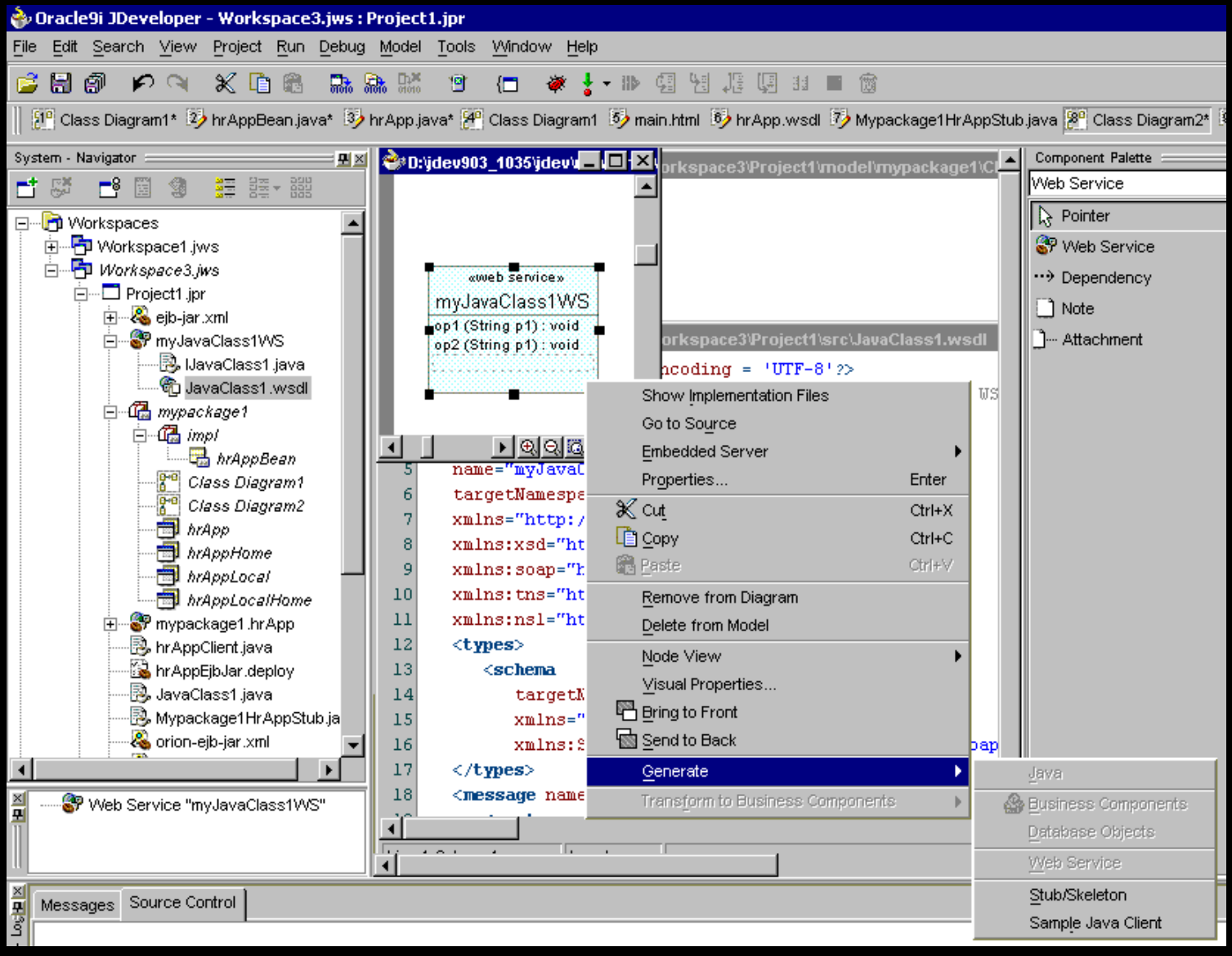


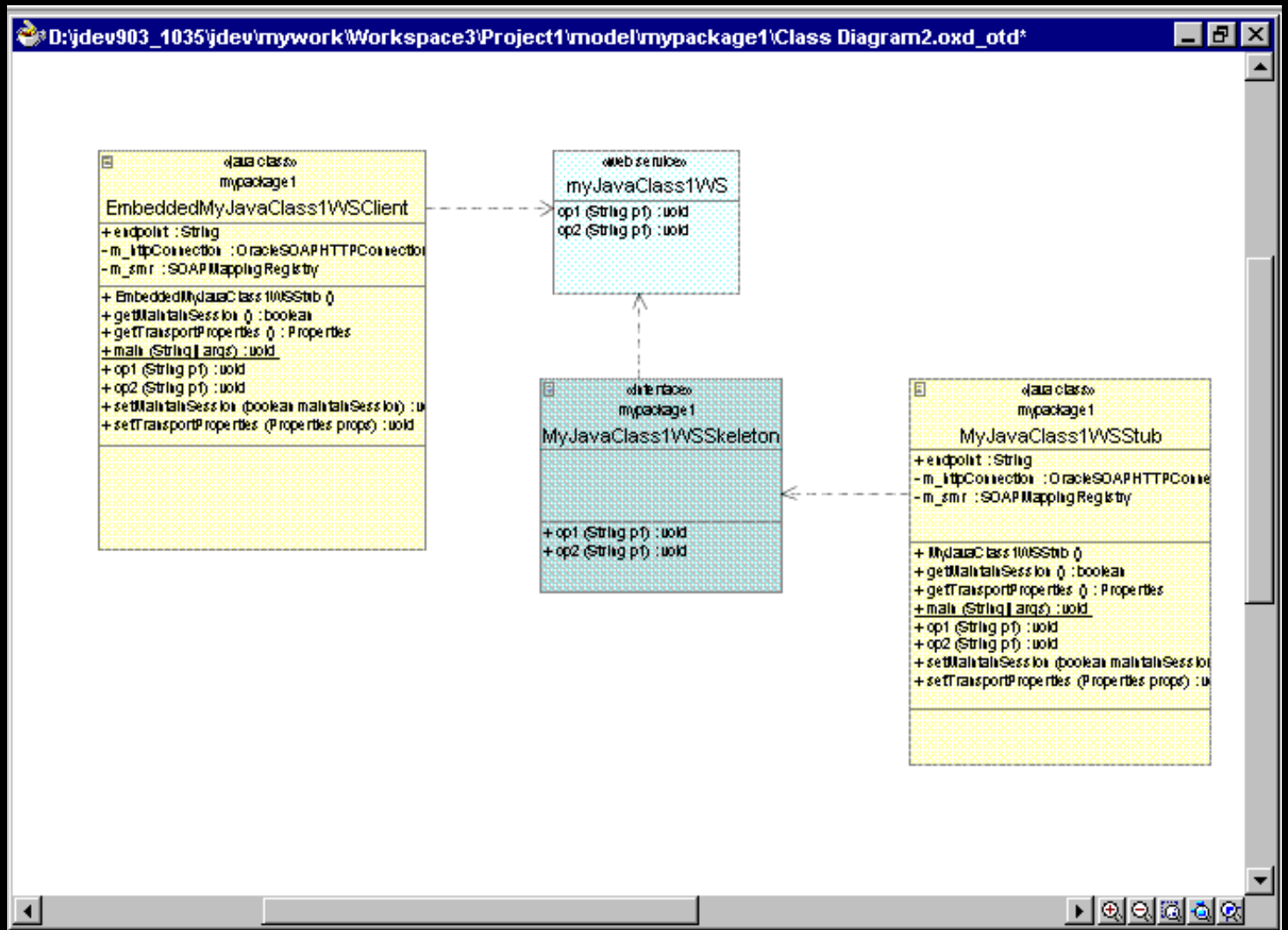
Tool Support for J2EE WS



Contract Driven Development

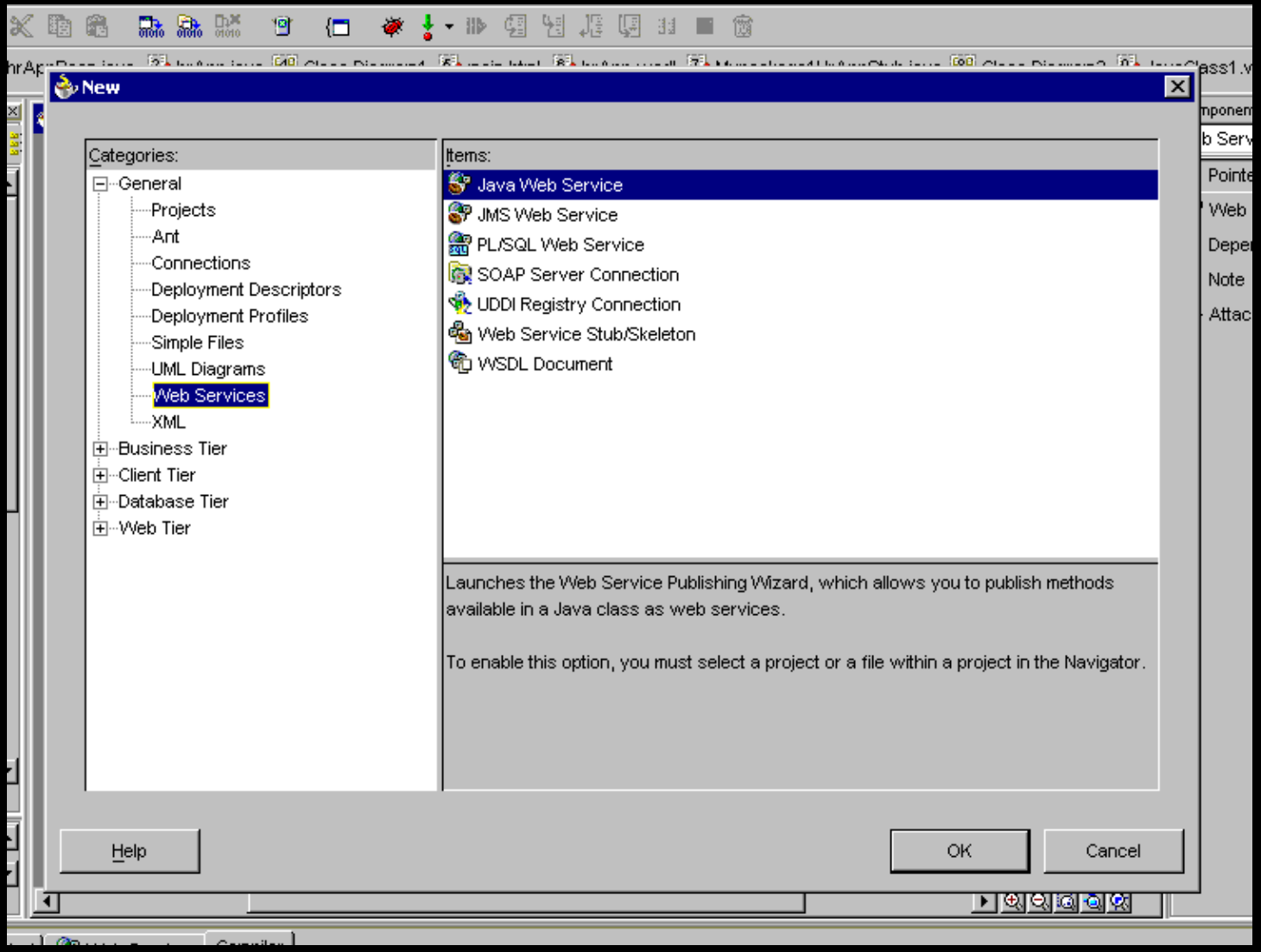
- Define UML web service PIM (service and operations)
- Select implementation target and generate PSM and Artifacts
    - WSDL
    - Client stub
    - (Sample) Client skeleton
    - Server stub
    - Server implementation skeleton
- Add client and server method implementations in framework context

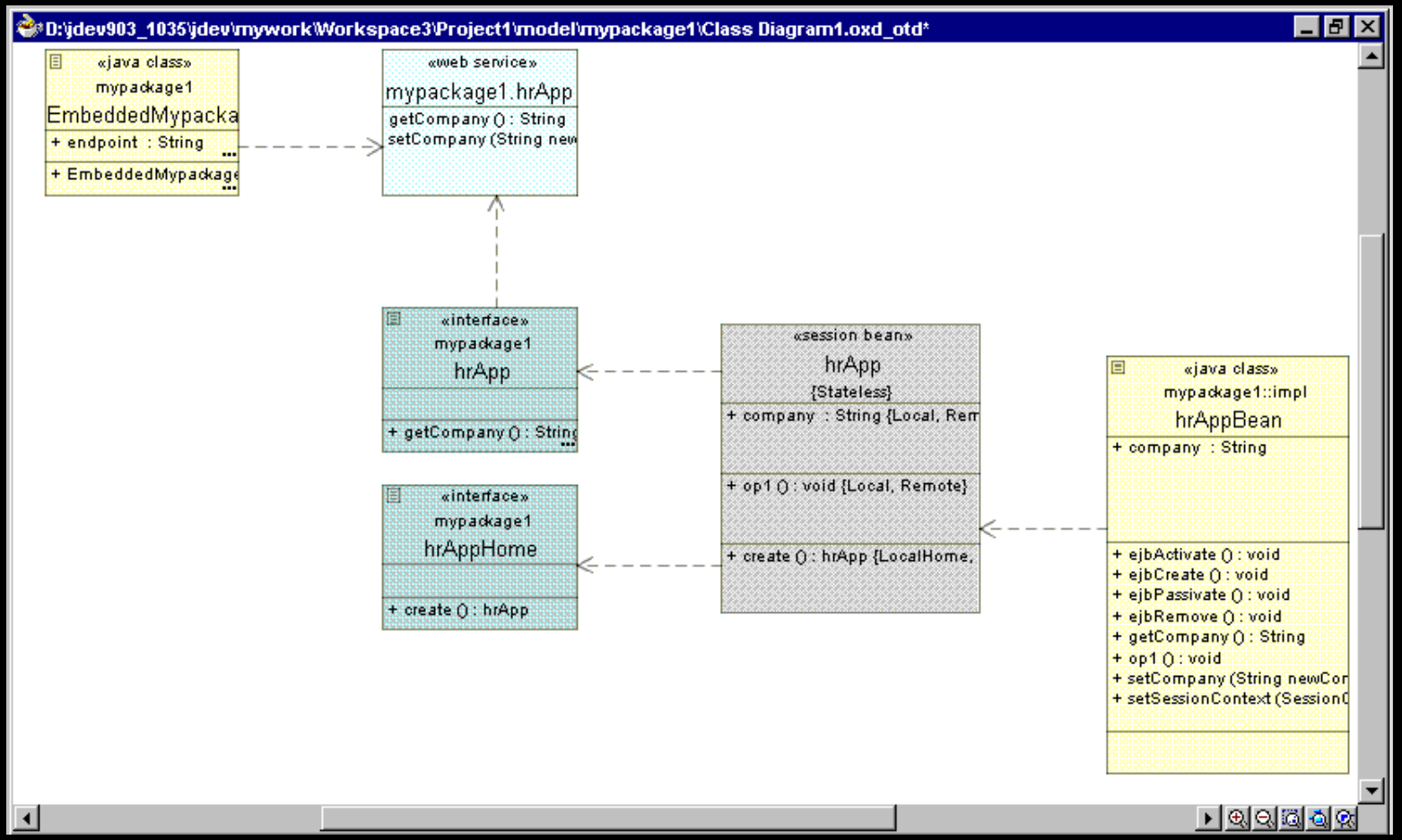

Oracle9i JDeveloper - Workspace3.jws : Project1.jpr
File Edit Search View Project Run Debug Model Tools Window Help
Class Diagram1* hrAppBean.java* hrApp.java* Class Diagram1 main.html hrApp.wsdl Mypackage1HrAppStub.java Class Diagram2*
System - Navigator
Workspaces
Workspace1.jws
Workspace3.jws
Project1.jpr
ejb-jar.xml
myJavaClass1WS
IJavaClass1.java
JavaClass1.wsdl
mypackage1
impl
hrAppBean
Class Diagram1
Class Diagram2
hrApp
hrAppHome
hrAppLocal
hrAppLocalHome
mypackage1.hrApp
hrAppClient.java
hrAppEjbJar.deploy
JavaClass1.java
Mypackage1HrAppStub.ja
orion-ejb-jar.xml
Web Service "myJavaClass1WS"
«web service»
myJavaClass1WS
op1 (String p1) : void
op2 (String p1) : void
Show Implementation Files
Go to Source
Embedded Server
Properties... Enter
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Remove from Diagram
Delete from Model
Node View
Visual Properties...
Bring to Front
Send to Back
Generate
Transform to Business Components
Java
Business Components
Database Objects
Web Service
Stub/Skeleton
Sample Java Client
Component Palette
Web Service
Pointer
Web Service
Dependency
Note
Attachment
Messages
Source Control


D:\jdev903_1035\jdev\mywork\Workspace3\Project1\model\mypackage1\Class Diagram2.oxd_otd*
«java class»
mypackage1
EmbeddedMyJavaClass1WSClient
+ endpoint : String
- m_httpConnection : OracleSOAPHTTPConnection
- m_smr : SOAPMappingRegistry
+ EmbeddedMyJavaClass1WSStub ()
+ getMaintainSession () : boolean
+ getTransportProperties () : Properties
+ main (String[] args) : void
+ op1 (String p1) : void
+ op2 (String p1) : void
+ setMaintainSession (boolean maintainSession) : v
+ setTransportProperties (Properties props) : void
«web service»
myJavaClass1WS
op1 (String p1) : void
op2 (String p1) : void
«interface»
mypackage1
MyJavaClass1WSSkeleton
+ op1 (String p1) : void
+ op2 (String p1) : void
«java class»
mypackage1
MyJavaClass1WSStub
+ endpoint : String
- m_httpConnection : OracleSOAPHTTPConne
- m_smr : SOAPMappingRegistry
+ MyJavaClass1WSStub ()
+ getMaintainSession () : boolean
+ getTransportProperties () : Properties
+ main (String[] args) : void
+ op1 (String p1) : void
+ op2 (String p1) : void
+ setMaintainSession (boolean maintainSessio
+ setTransportProperties (Properties props) : v

Implementation Driven Development

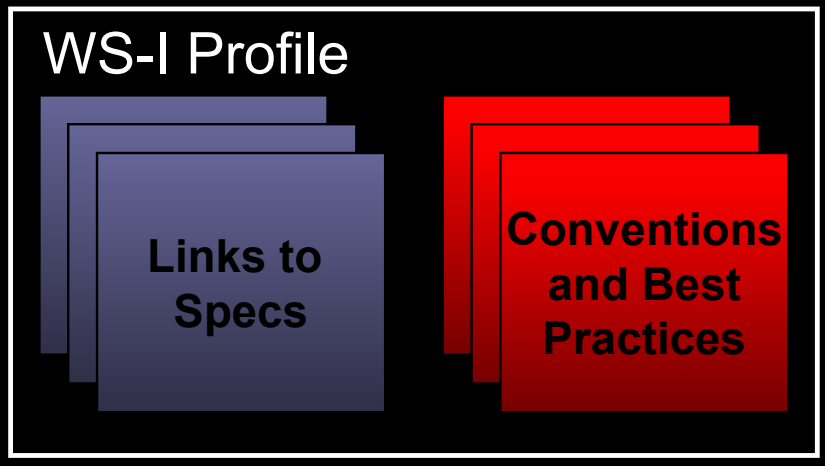
- Reverse Engineer UML web service PIM, by selecting methods to be exposed from existing (framework or legacy) implementation
- Tool generation of PSM and Artifacts
  - WSDL
  - Client stub
  - (Sample) Client skeleton
  - Server stub
- Refine client implementation


New
Categories:
General
Projects
Ant
Connections
Deployment Descriptors
Deployment Profiles
Simple Files
UML Diagrams
Web Services
XML
Business Tier
Client Tier
Database Tier
Web Tier
Items:
Java Web Service
JMS Web Service
PL/SQL Web Service
SOAP Server Connection
UDDI Registry Connection
Web Service Stub/Skeleton
WSDL Document
Launches the Web Service Publishing Wizard, which allows you to publish methods available in a Java class as web services.
To enable this option, you must select a project or a file within a project in the Navigator.
Help
OK
Cancel


D:\jdev903_1035\jdev\mywork\Workspace3\Project1\model\mypackage1\Class Diagram1.oxd_otd*
«java class»
mypackage1
EmbeddedMypacka
+ endpoint : String
+ EmbeddedMypackage
«web service»
mypackage1.hrApp
getCompany () : String
setCompany (String new
«interface»
mypackage1
hrApp
+ getCompany () : String
«interface»
mypackage1
hrAppHome
+ create () : hrApp
«session bean»
hrApp
{Stateless}
+ company : String {Local, Rem
+ op1 () : void {Local, Remote}
+ create () : hrApp {LocalHome,
«java class»
mypackage1::impl
hrAppBean
+ company : String
+ ejbActivate () : void
+ ejbCreate () : void
+ ejbPassivate () : void
+ ejbRemove () : void
+ getCompany () : String
+ op1 () : void
+ setCompany (String newCor
+ setSessionContext (Session

Standards are essential

- JAX-RPC, JSR-109
- Interoperability : WS-I
  - Profiles
  - Best practices
  - Monitoring, Logging, Analyzing tools



- Customer quote:
  "there are standards, and there are standards"
  - Royalty-free versus licensing required?
  - Centralized control versus community process?
  - Overlapping content versus complementary?
  - Historical competence versus competitive body?

Download Oracle 9*i* Application Server and JDeveloper for Free

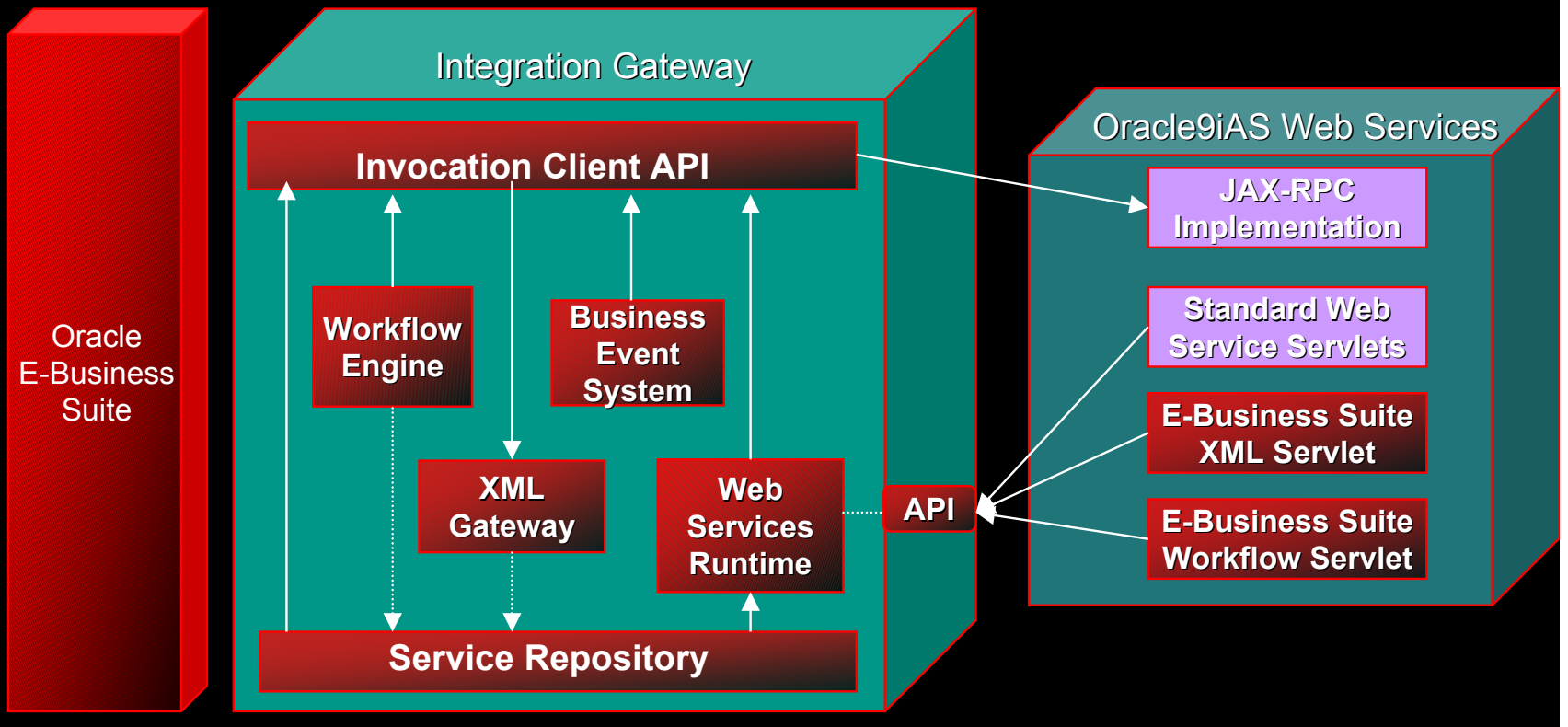
http://otn.oracle.com/products/

- Full functionality software
- License fee payable on deployment of commercial application
- Many “How-To’s” and Examples available

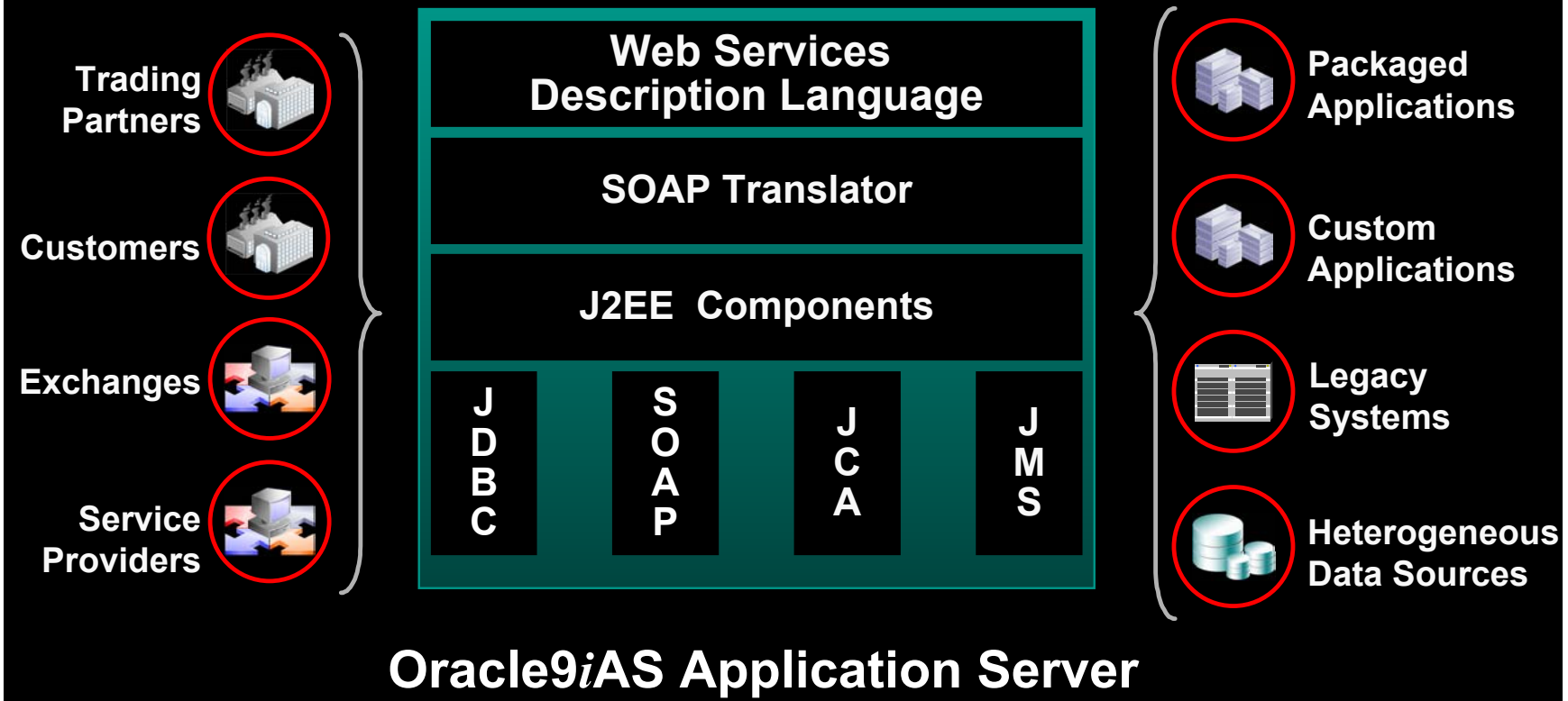

QUESTIONS
&
ANSWERS
ORACLE

e-businesssuite

Integration Gateway



Oracle9*i*AS: Application Server



Oracle9*i*AS Application Server

Oracle9*i* JDeveloper: Building Web services

- Built on standards
  - SOAP, WSDL, UDDI
  - Java XML Pack
  - Apache SOAP
- Complete lifecycle
  - Model, build, deploy, debug, profile
- Integrated
  - Publish J2EE components
  - Publish DB stored procedures
  - Consume external services
- UDDI lookup and discovery