

Session #: E1111

# Creating Web Services with WebSphere Studio Application Developer

Speaker: Rick Weaver

Title: IBM Worldwide Technical Sales Support  
WebSphere Development Tools

May 7 - 10, 2002



# Objectives



- Understand the Value of Web Services
- Understand the tasks necessary to develop and deploy a Web Service
- Understand the usage of Web Services in WebSphere



- **Brief Introduction to Web Services**
  - ▶ Value of Web Services
- Developing Web Services with WebSphere Studio Application Developer
- Deploying Web Services to WebSphere
- Enterprise Services



*Web services are self-contained, modular applications that can be described, published, located, and invoked over a network.*

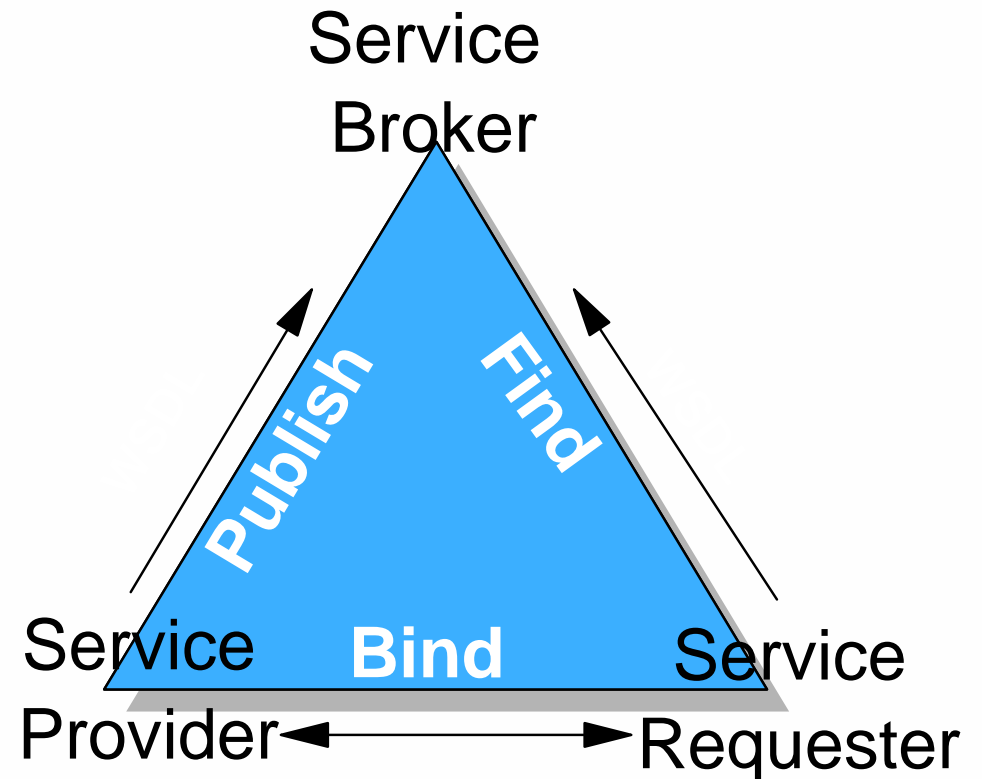
- Represent a shift towards 'service-oriented' applications
- Any application can be exposed as a Web Service
  - ▶ EJB, COM Object, JavaBean
- Web service paradigm is about interoperability and platform-neutral communications



- SOAP - Simple Object Access Protocol
  - ▶ Network-neutral service access protocol
- WSDL - Web Service Descriptive Language
  - ▶ XML-based interface definition that describes a service and its implementation
- UDDI - Universal Description, Discovery and Integration
  - ▶ A way to find out about available services

- **Service Broker**

- ▶ A searchable repository of service descriptions
- ▶ Service providers publish their services
- ▶ Service requesters locate and invoke services



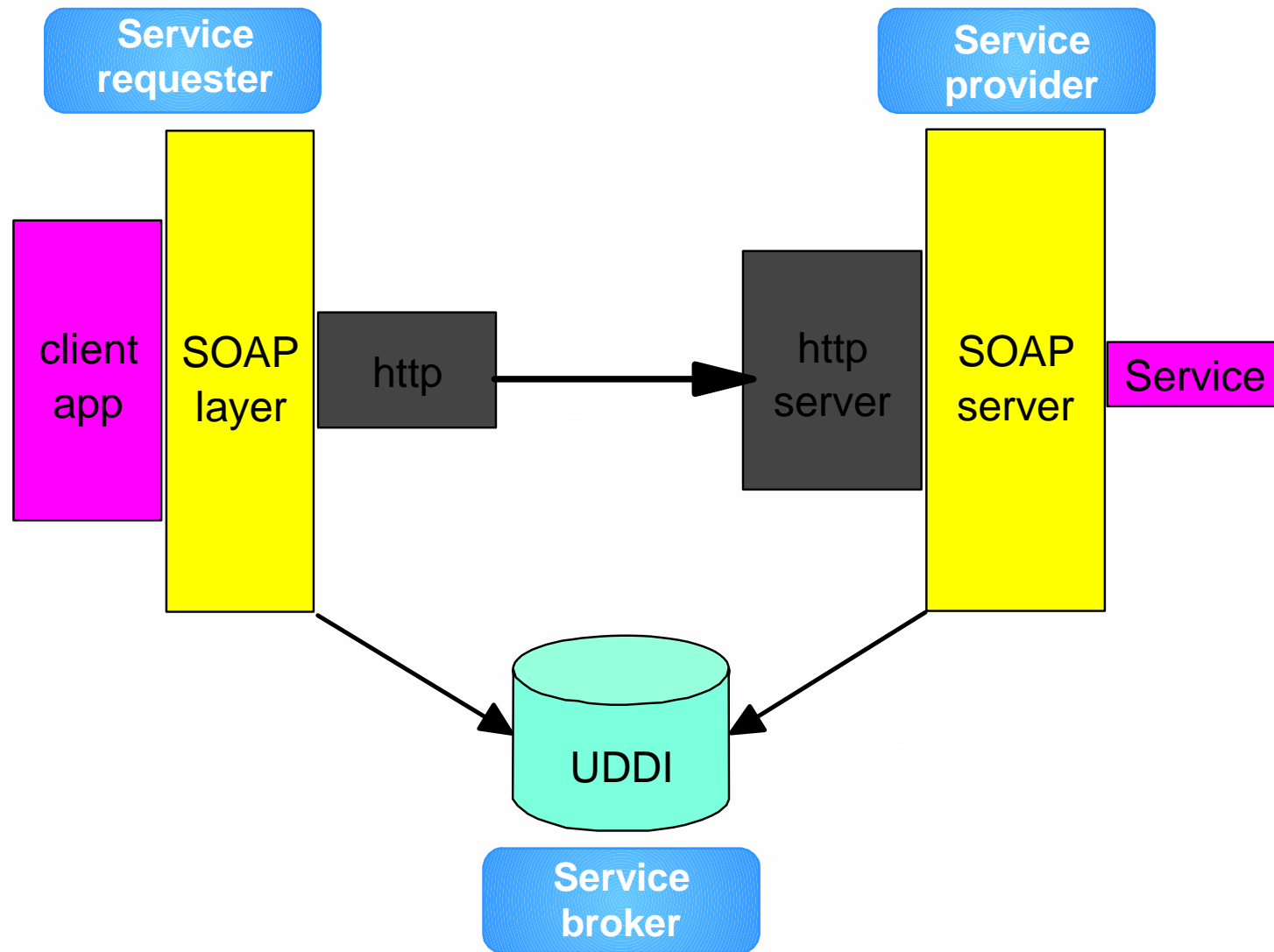
- **Service Provider**

- ▶ Provide applications as a Web Services
- ▶ Publish their services

- **Service Requester**

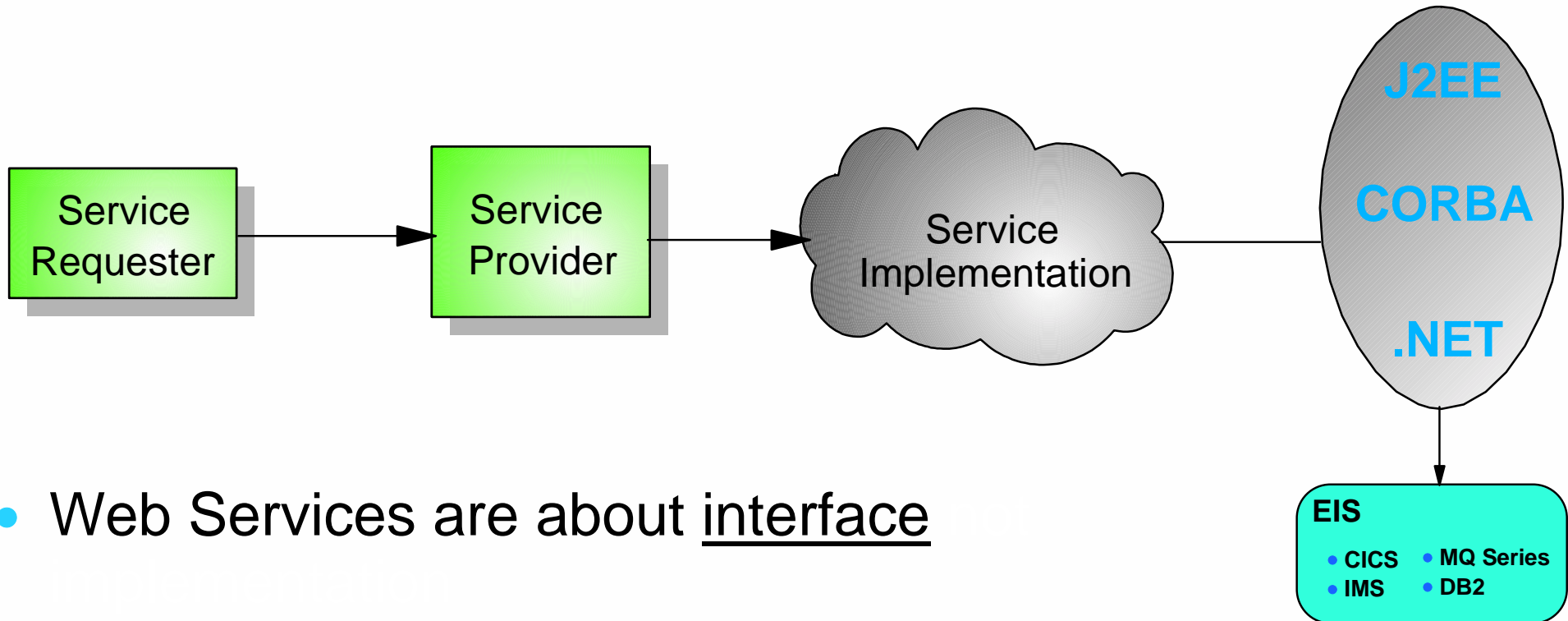
- ▶ Client that invokes a Web Service

# Web Service Model





# Web Services are about Interface



- Web Services are about interface
- J2EE is about implementation
  - ▶ One possible implementation of a Web Service
  - ▶ Programming model supported by WebSphere





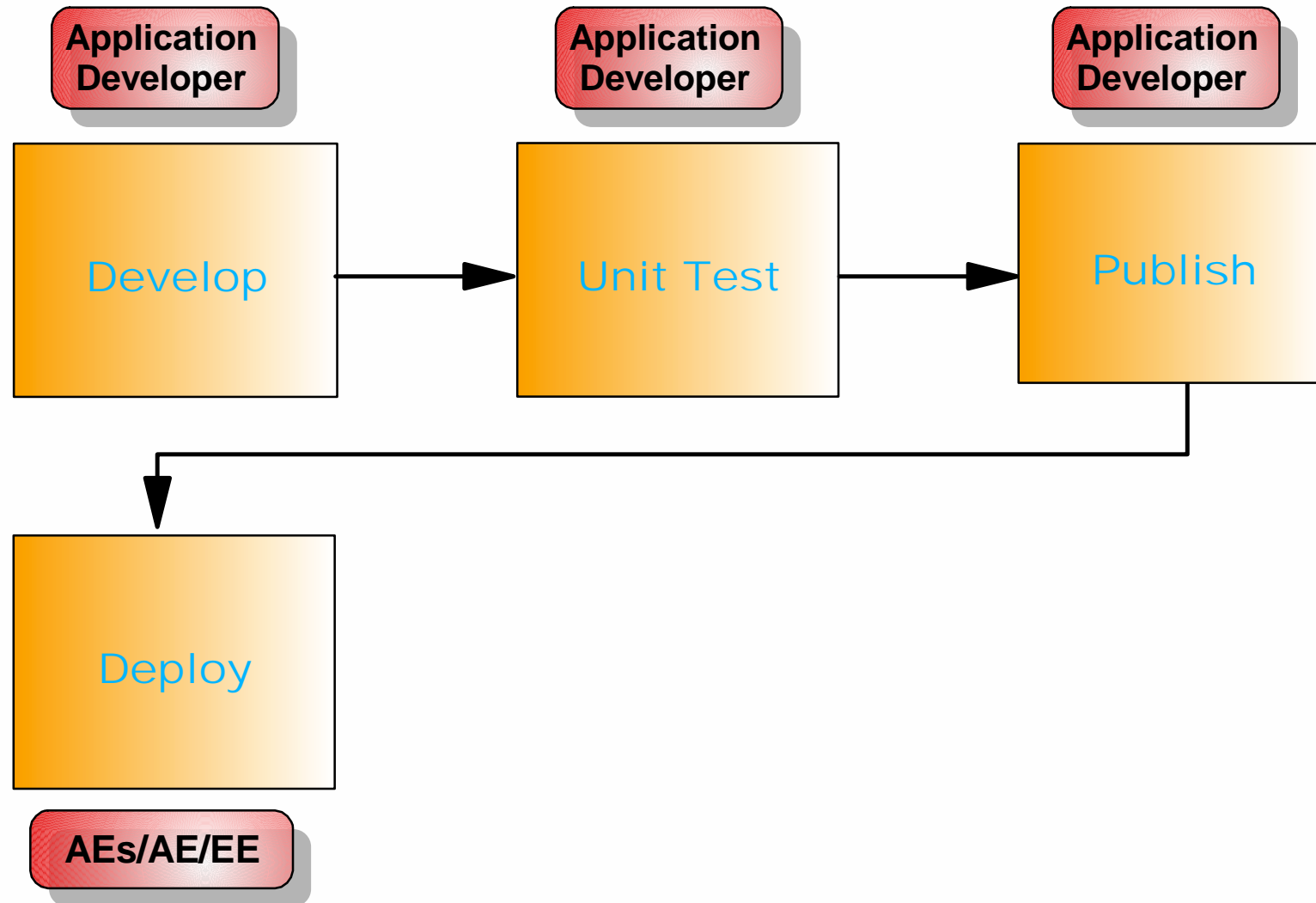
- Business Value: Business Integration made easy!
  - ▶ Based on open standards (HTTP, SOAP, XML)
- Development Value: Web Services decouple client (interface) from server (implementation)



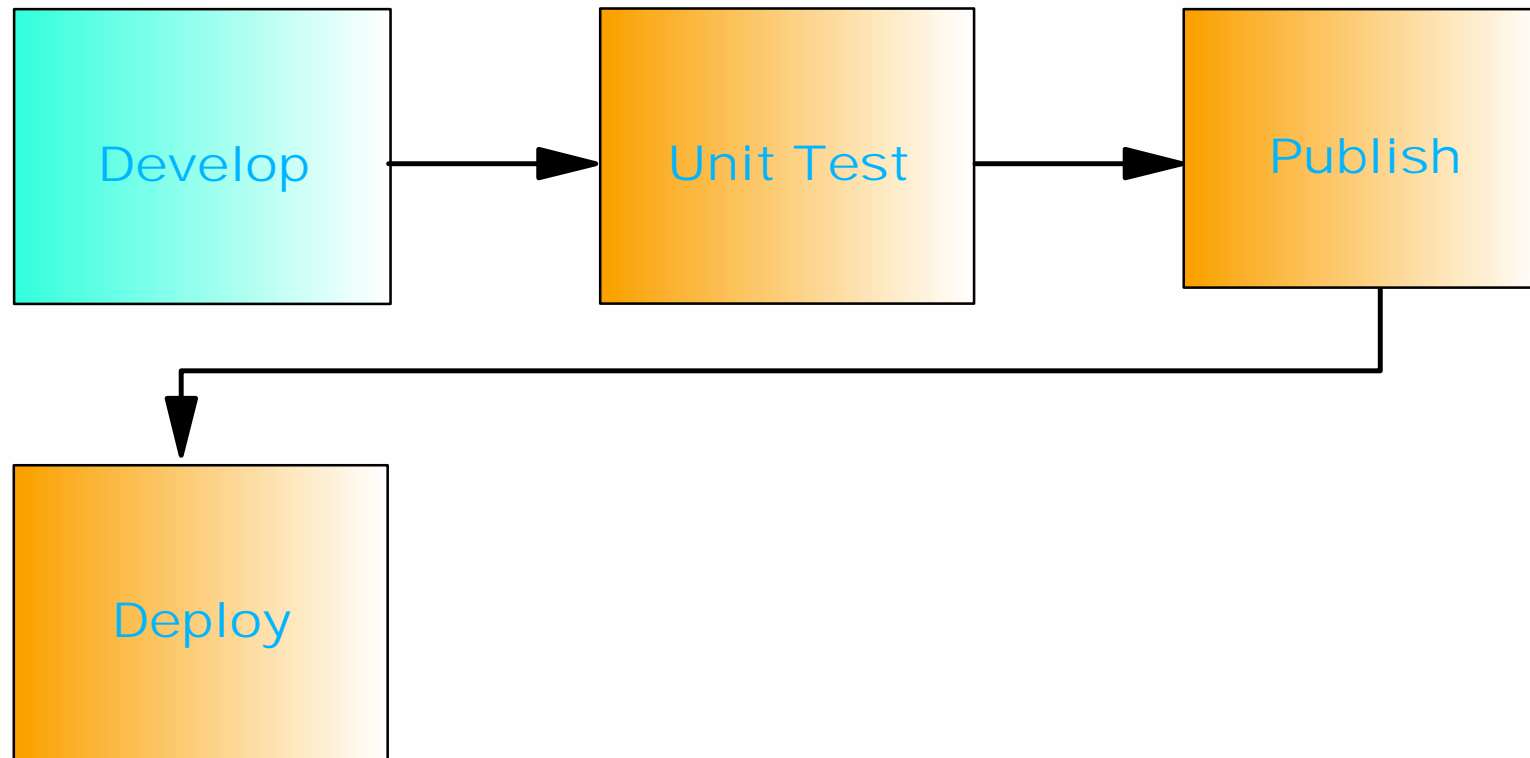
- Brief Introduction to Web Services
  - ▶ Value of Web Services
- **Developing Web Services with WebSphere Studio Application Developer**
- Deploying Web Services to WebSphere
- Enterprise Services



# Web Service Development Lifecycle



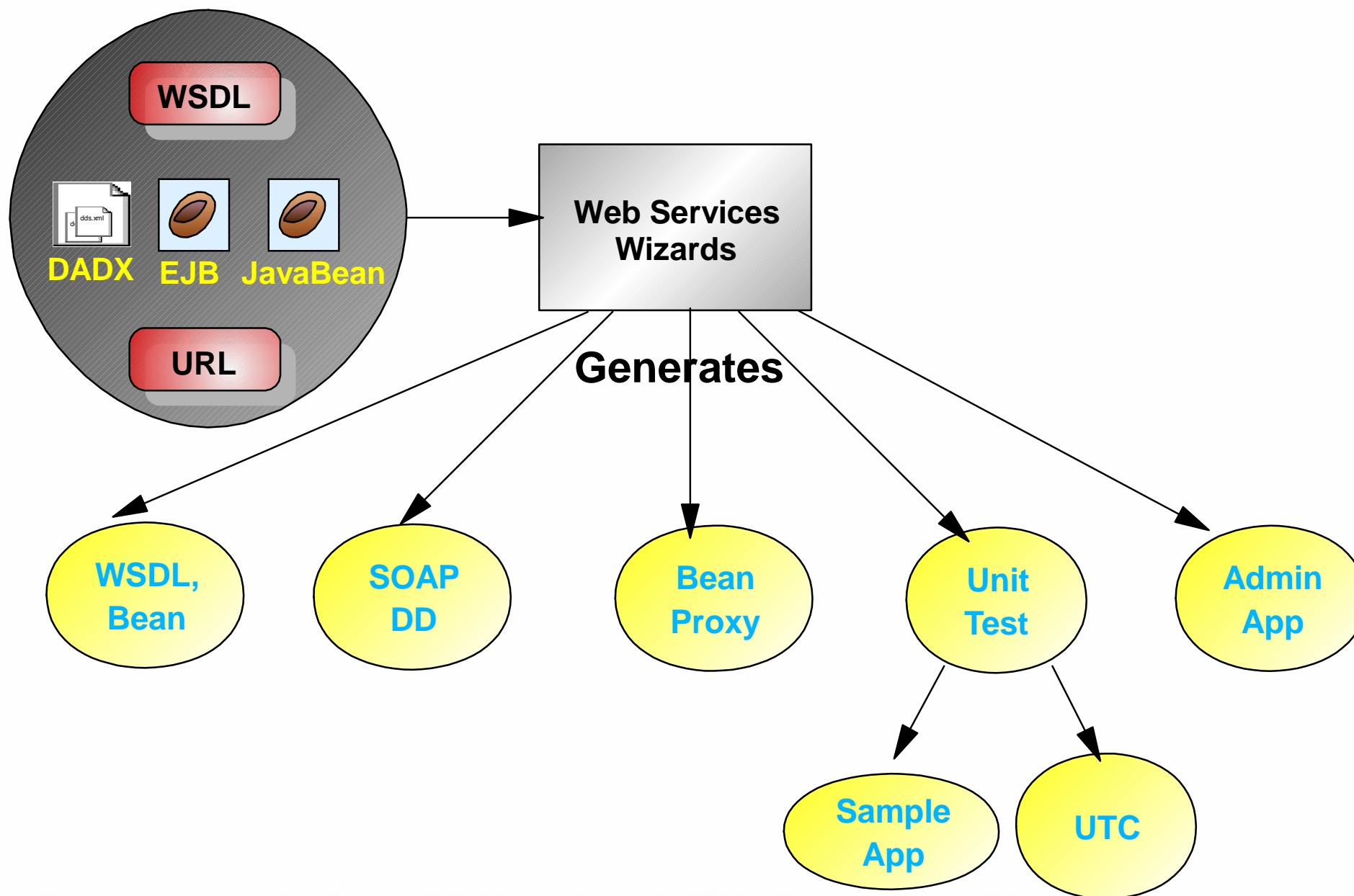
# Web Service Development Lifecycle



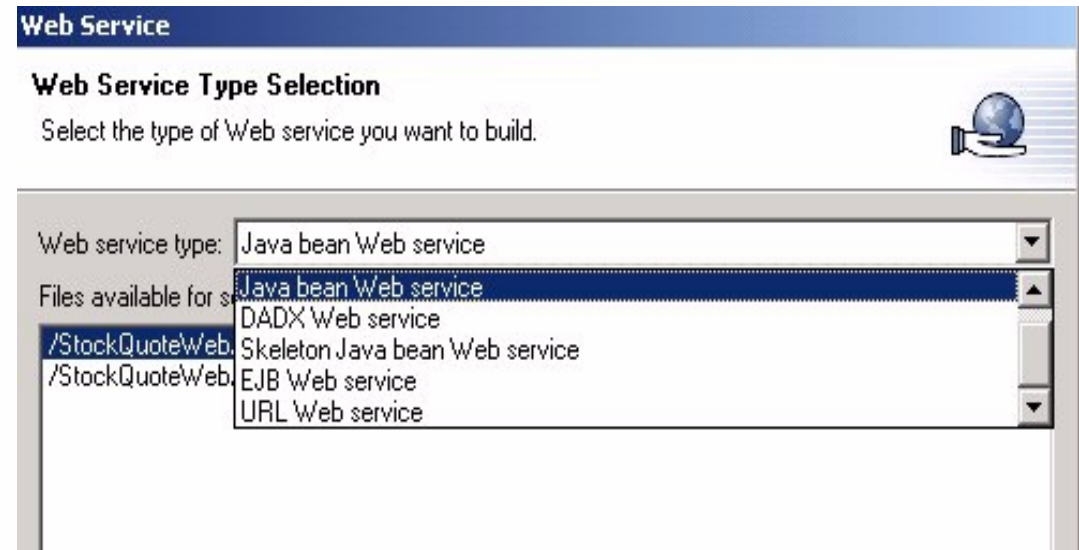
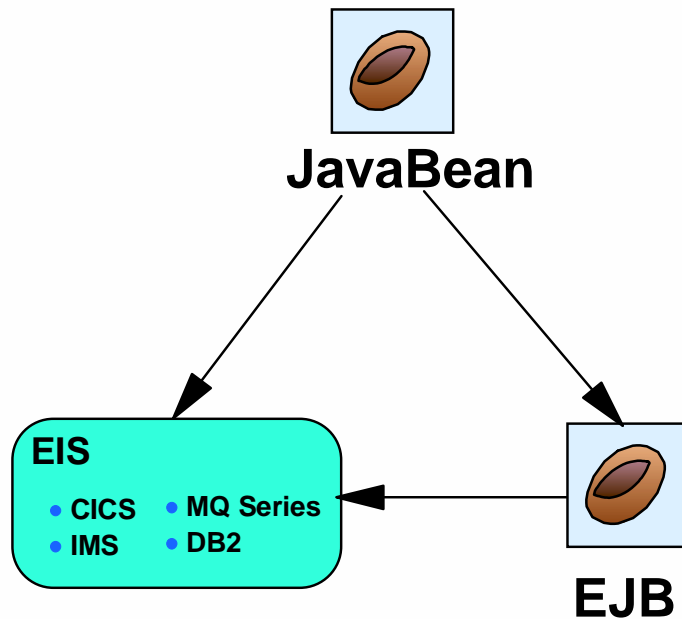
- Complex protocols - SOAP, UDDI
- Complex hand-coding of XML-based SOAP deployment descriptors and WSDL
- Without proper tooling, low-level Java coding skills required

WebSphere Application  
Developer simplifies Web  
Services development

# Creating Web Services



# Web Services JavaBean Wizard

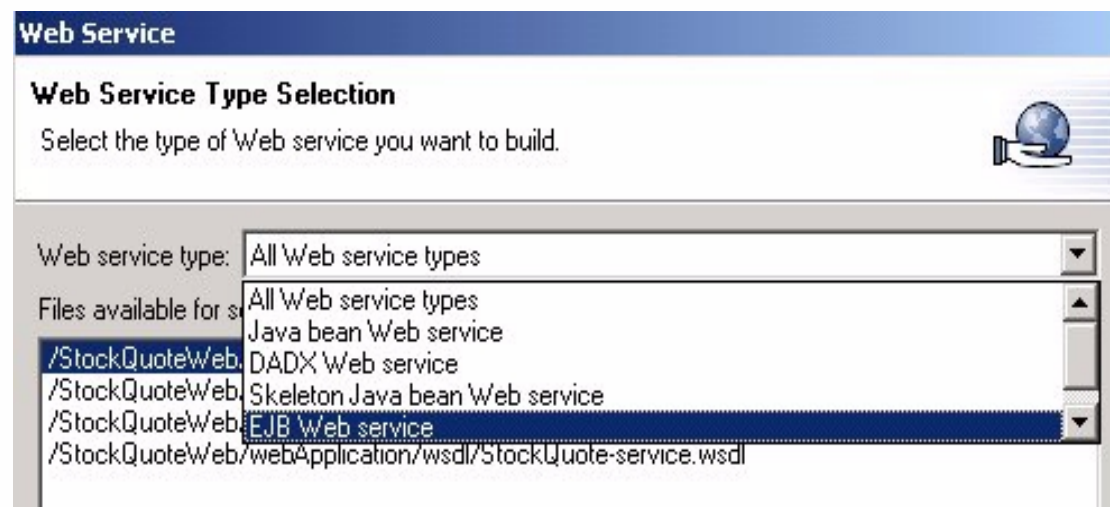
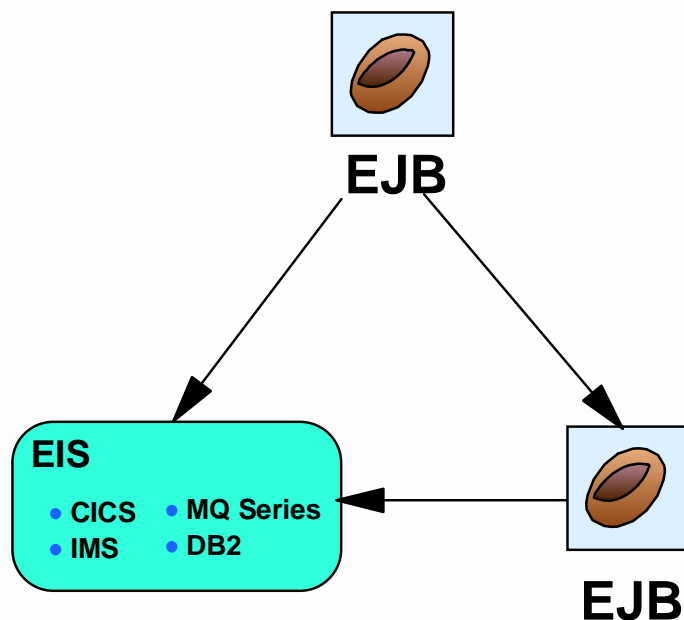


- Using an existing JavaBean you specify which methods you want to expose as a Web Service
- Generates WSDL representing the Web Service
- Bottom-up approach





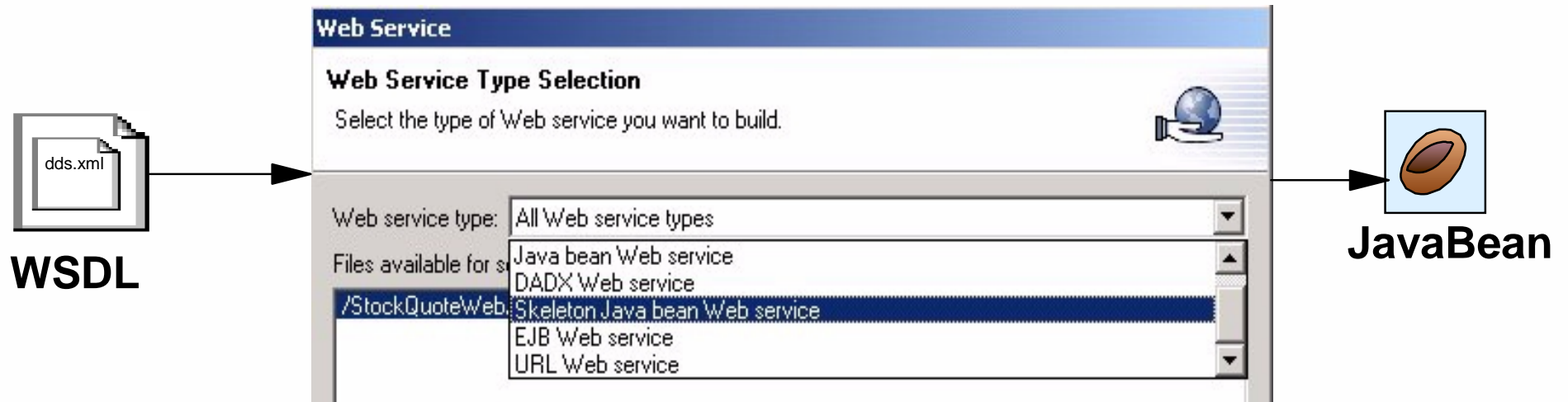
# Web Services EJB Wizard



- Using an existing EJB you specify which methods you want to expose as a Web Service
  - ▶ Generates WSDL representing the Web Service
  - ▶ Bottom-up approach

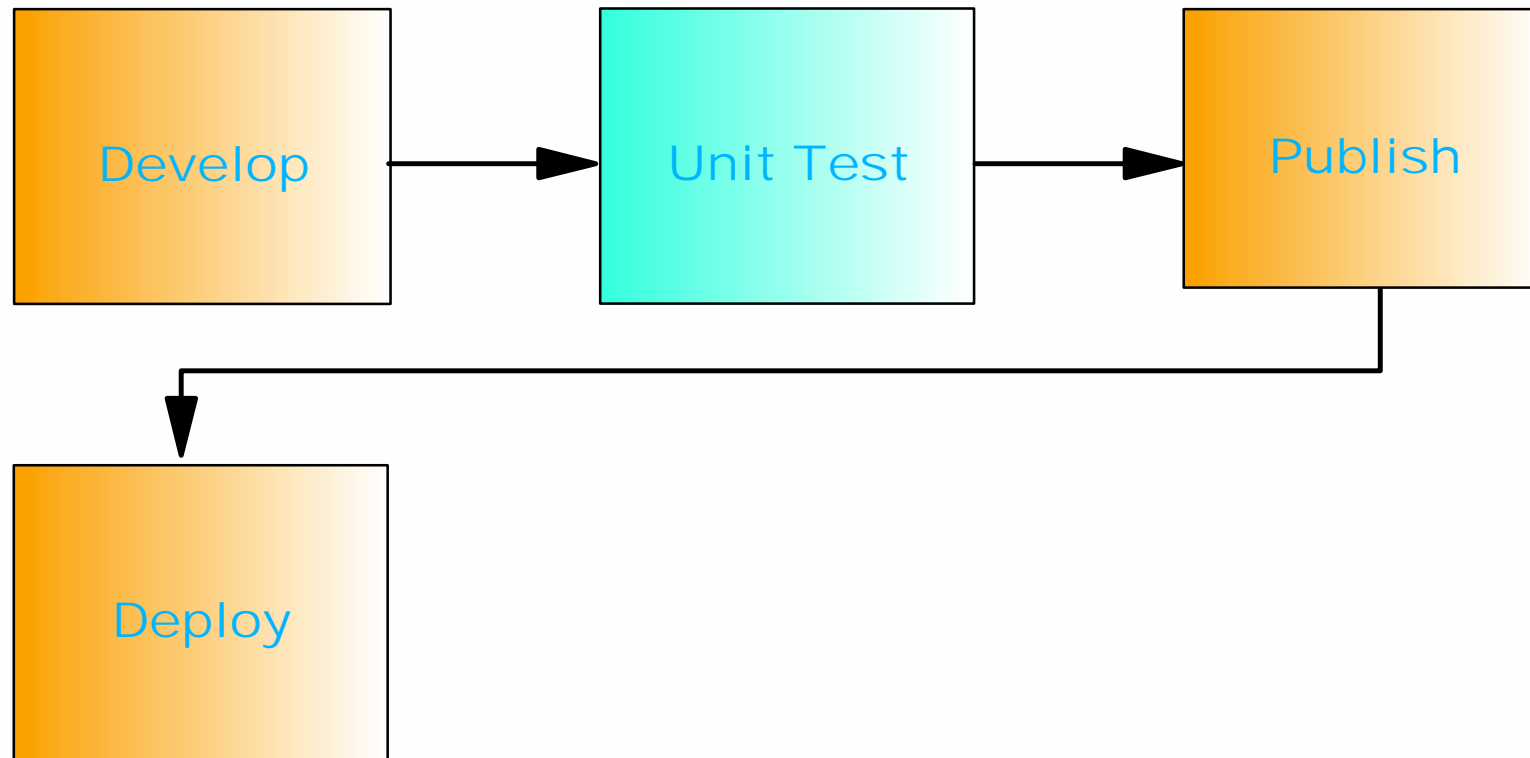


# Web Services WSDL Wizard

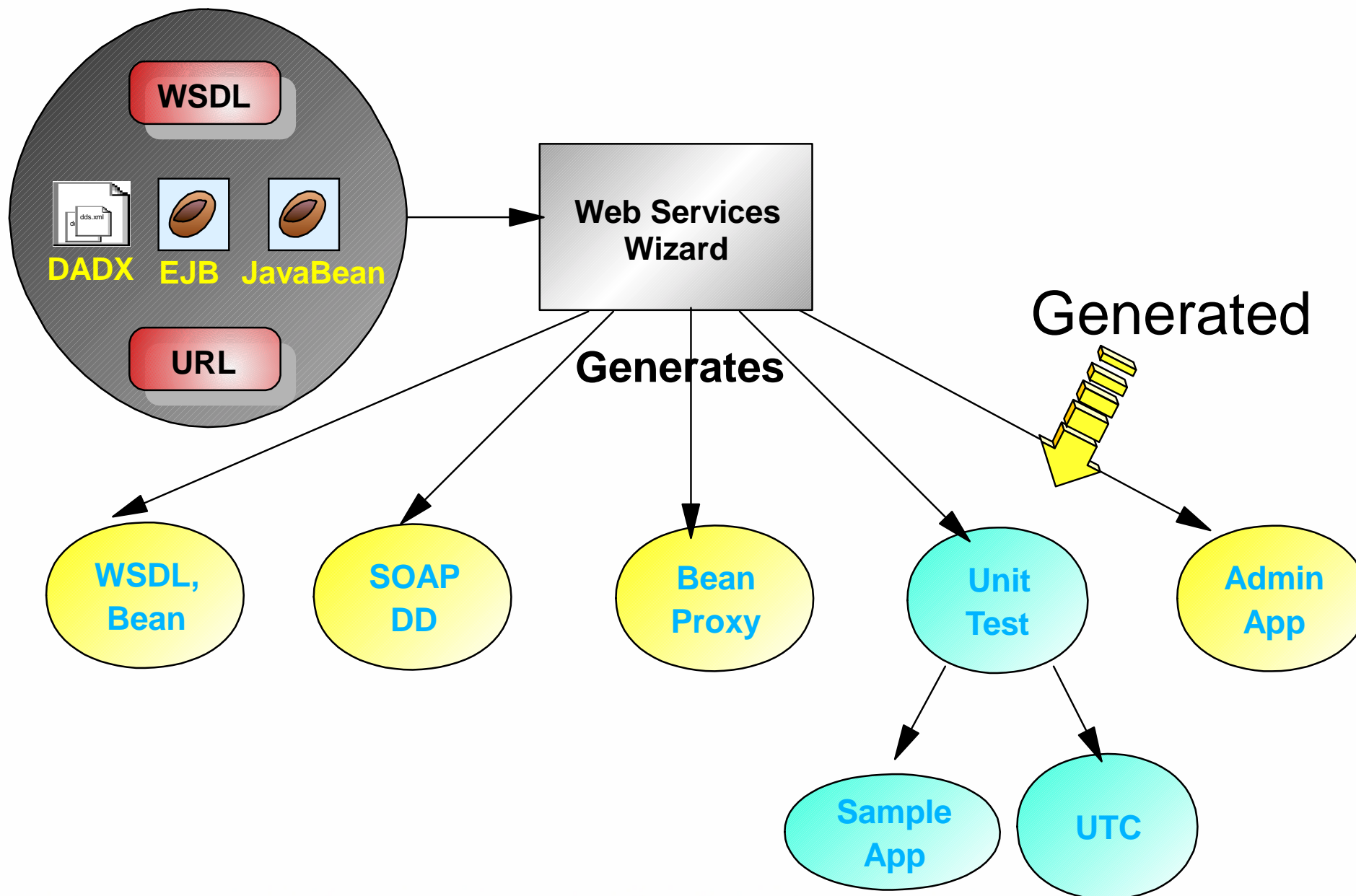


- Using an existing WSDL file a skeleton JavaBean is generated
  - ▶ represents starting point for Java developer to add implementation logic
- Top-down approach

# Web Service Development Lifecycle



# Unit Testing Your Web Service



# Sample Application



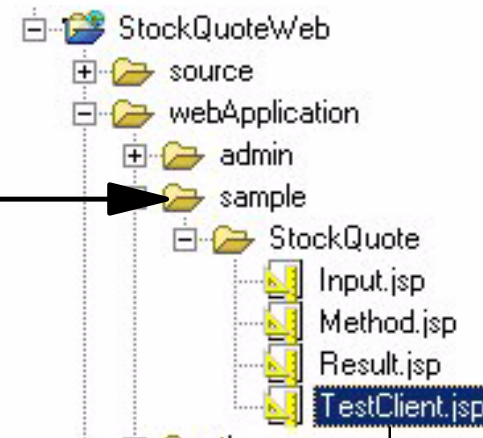
**Web Service**

**Web Service Sample Generation**

Do you want to generate a sample Web application?

☒ Generate a sample

generates



run

**Run on Server**

Validate HTML Syntax

Team ▶

Compare With ▶

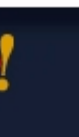
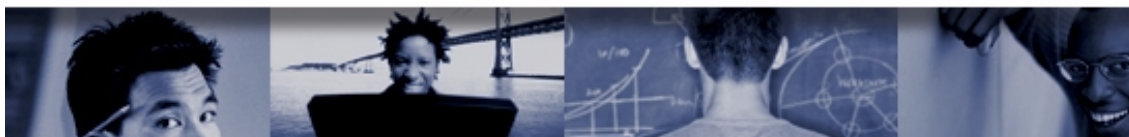
Replace With ▶

Properties

- Generated Web application to unit test your Web Service
- Contains code to the use the Web service proxy
- Useful template for 'real' Web Service client

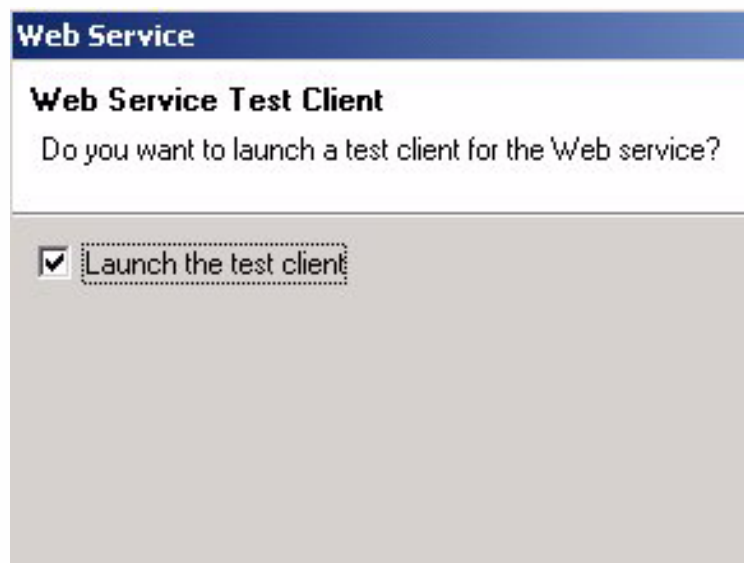
http://localhost:8080/StockQuoteWeb/sample/StockQuote/TestClient.jsp

Methods	Inputs
<ul style="list-style-type: none"><li>• <a href="#">setEndPoint</a></li><li>• <a href="#">getEndPoint</a></li><li>• <a href="#">getQuote</a></li></ul>	symbol: <input type="text" value="IBM"/>  <input type="button" value="Invoke"/> <input type="button" value="Clear"/>
<b>Result</b>  123.45	

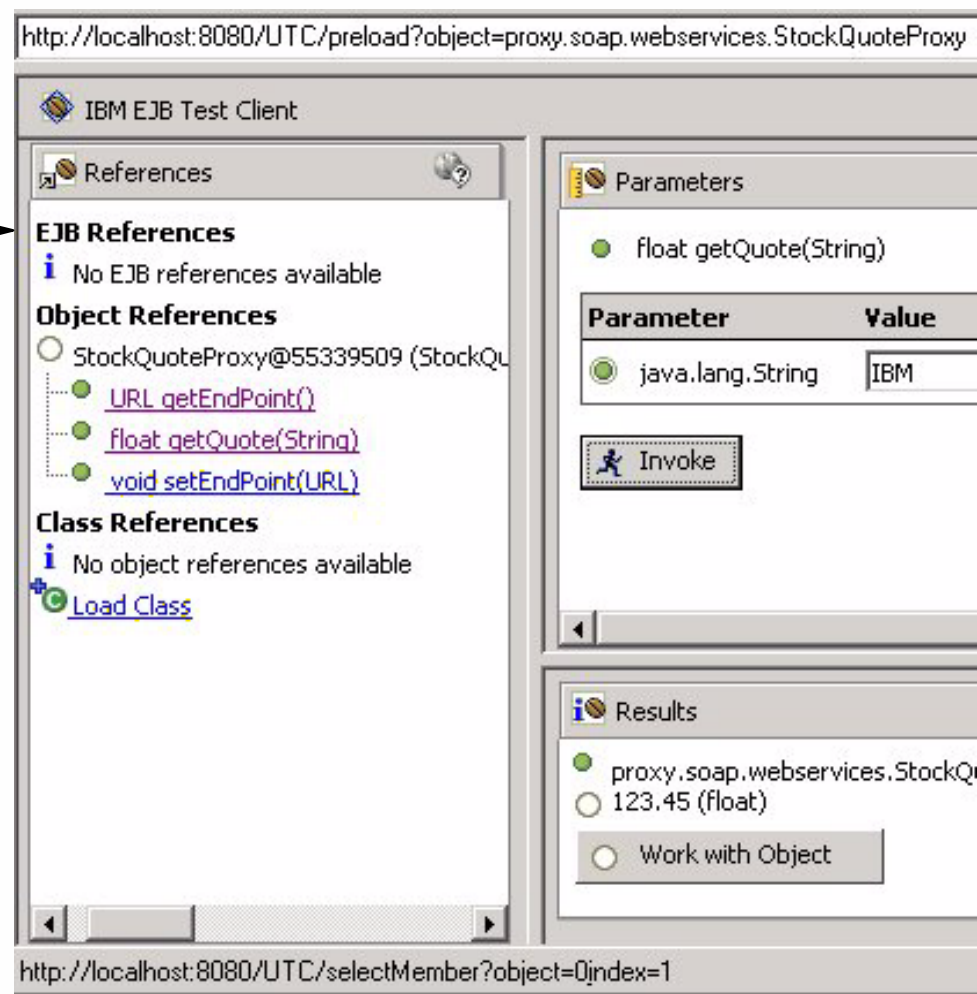




# Using the Universal Test Client



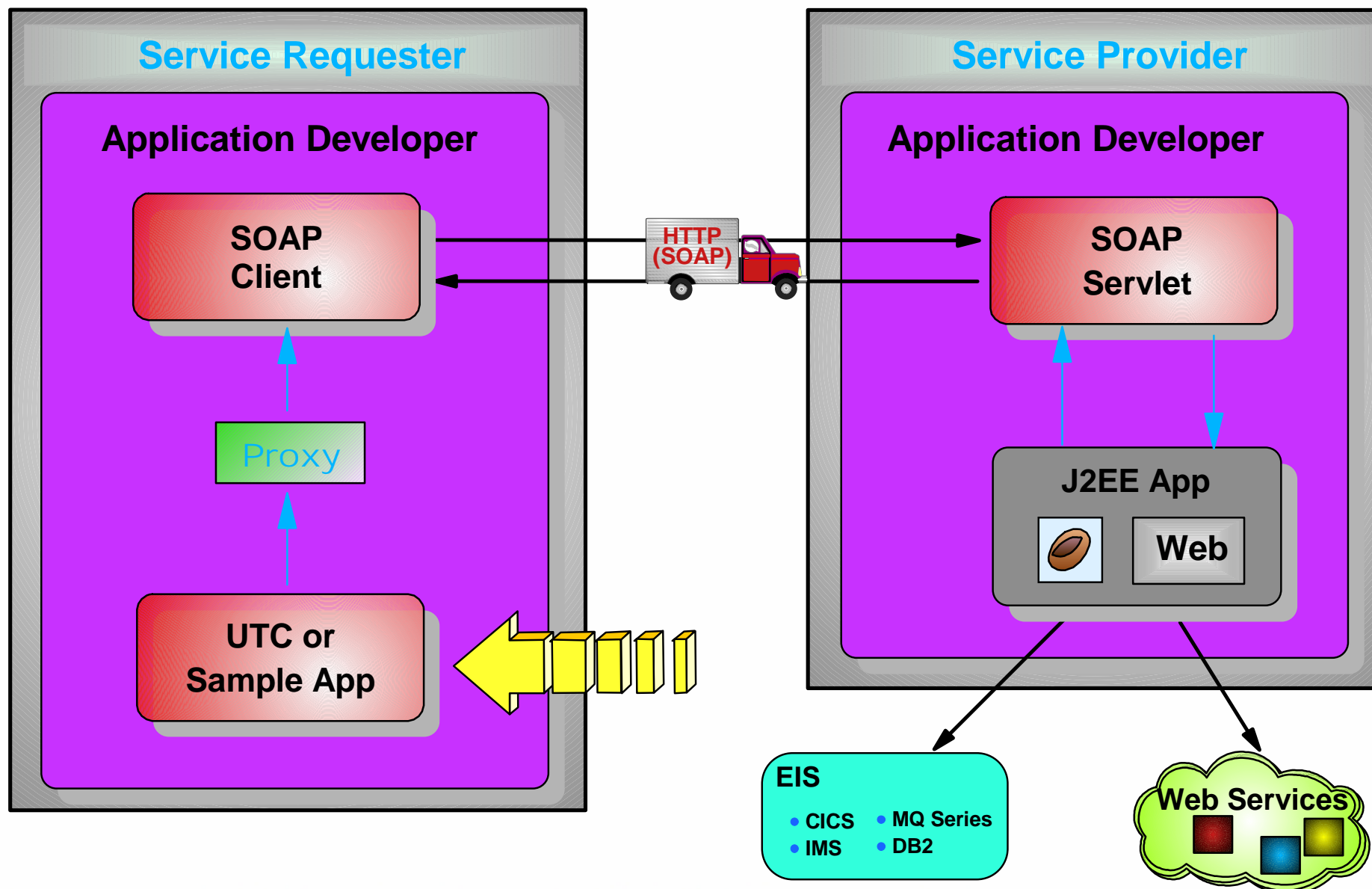
launch



- UTC launched to test your Web Service
- Ideal for quick unit testing
- Doesn't generate any code

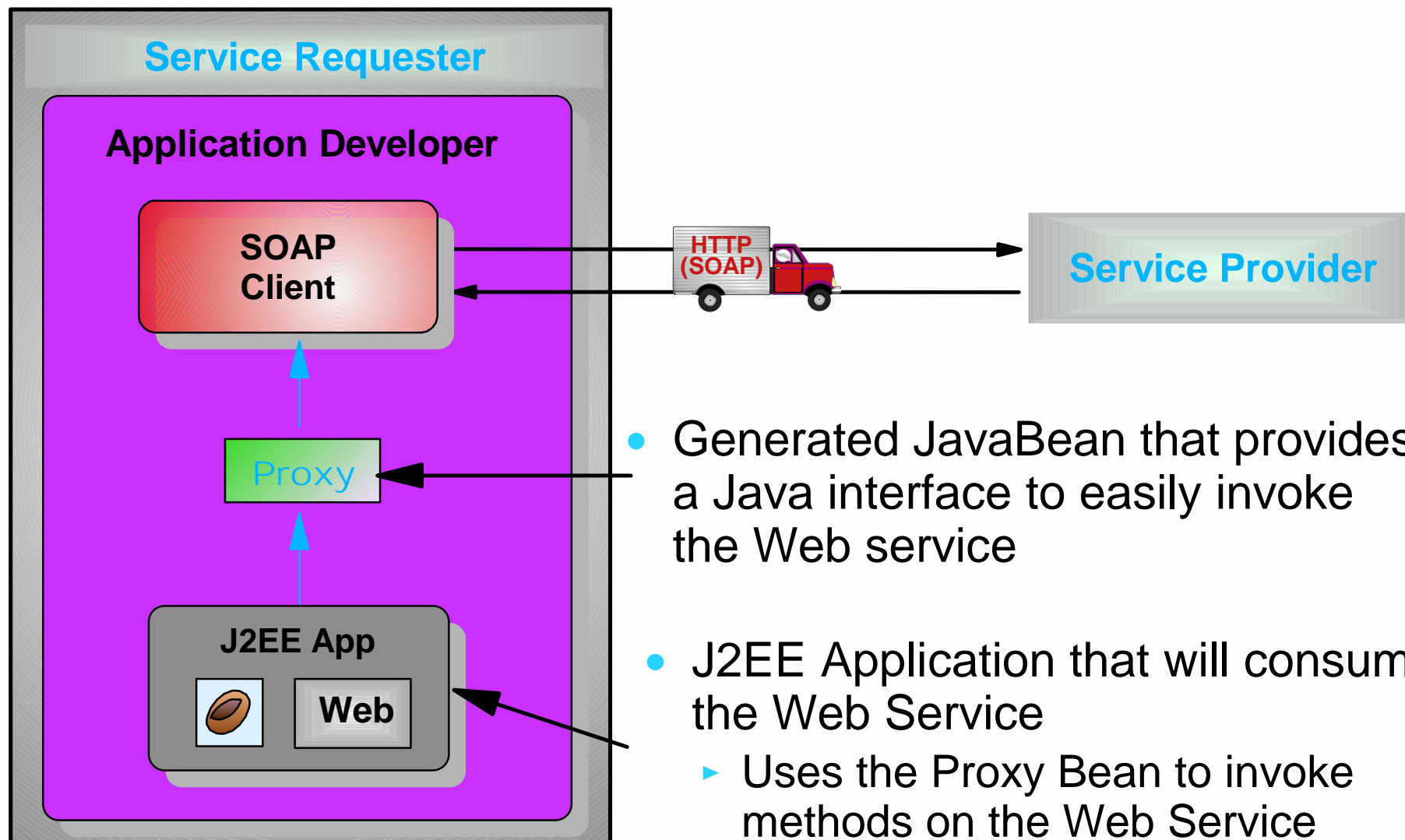


# Unit Testing Your Web Service





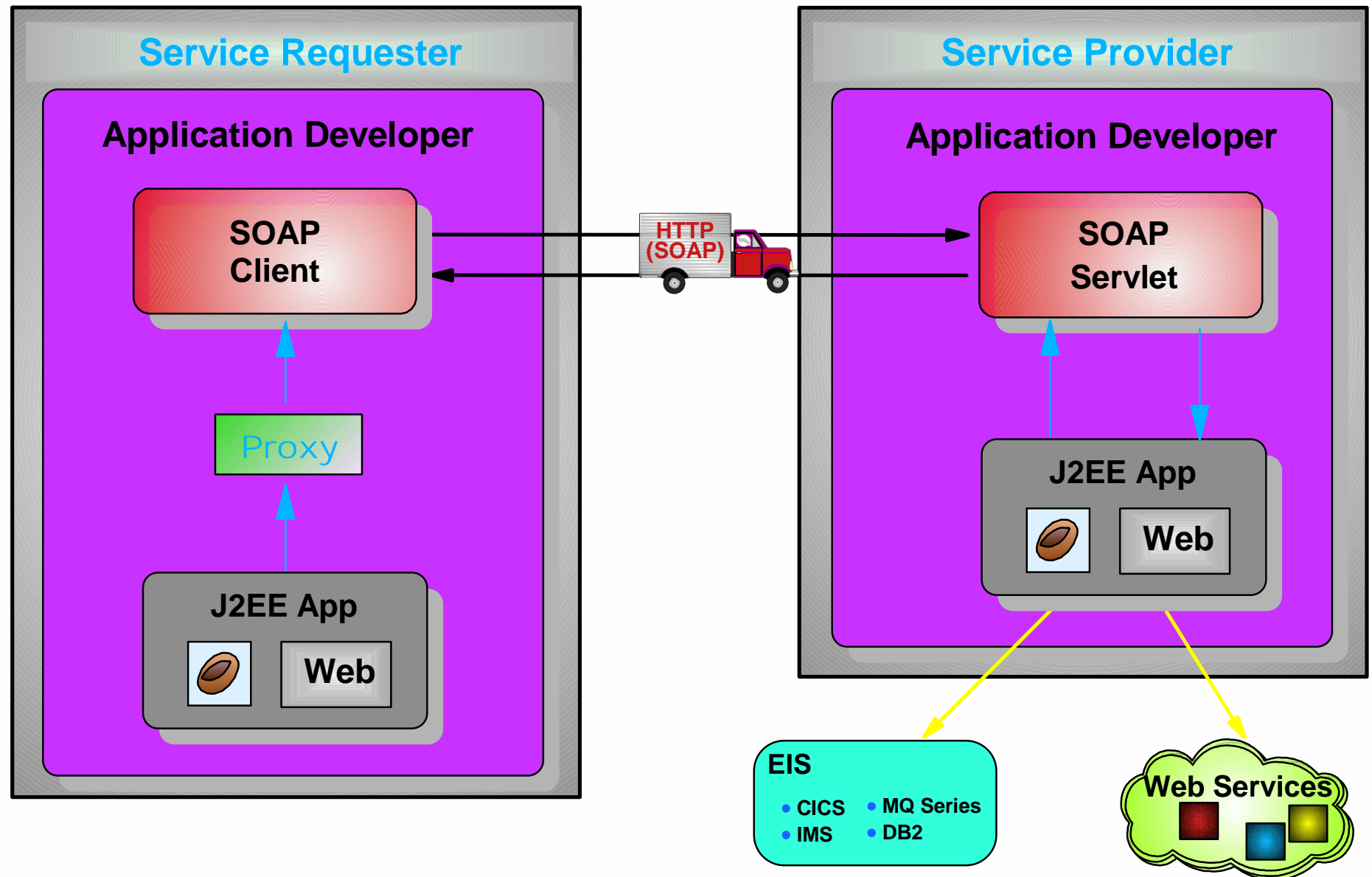
# Developing A Web Service Client



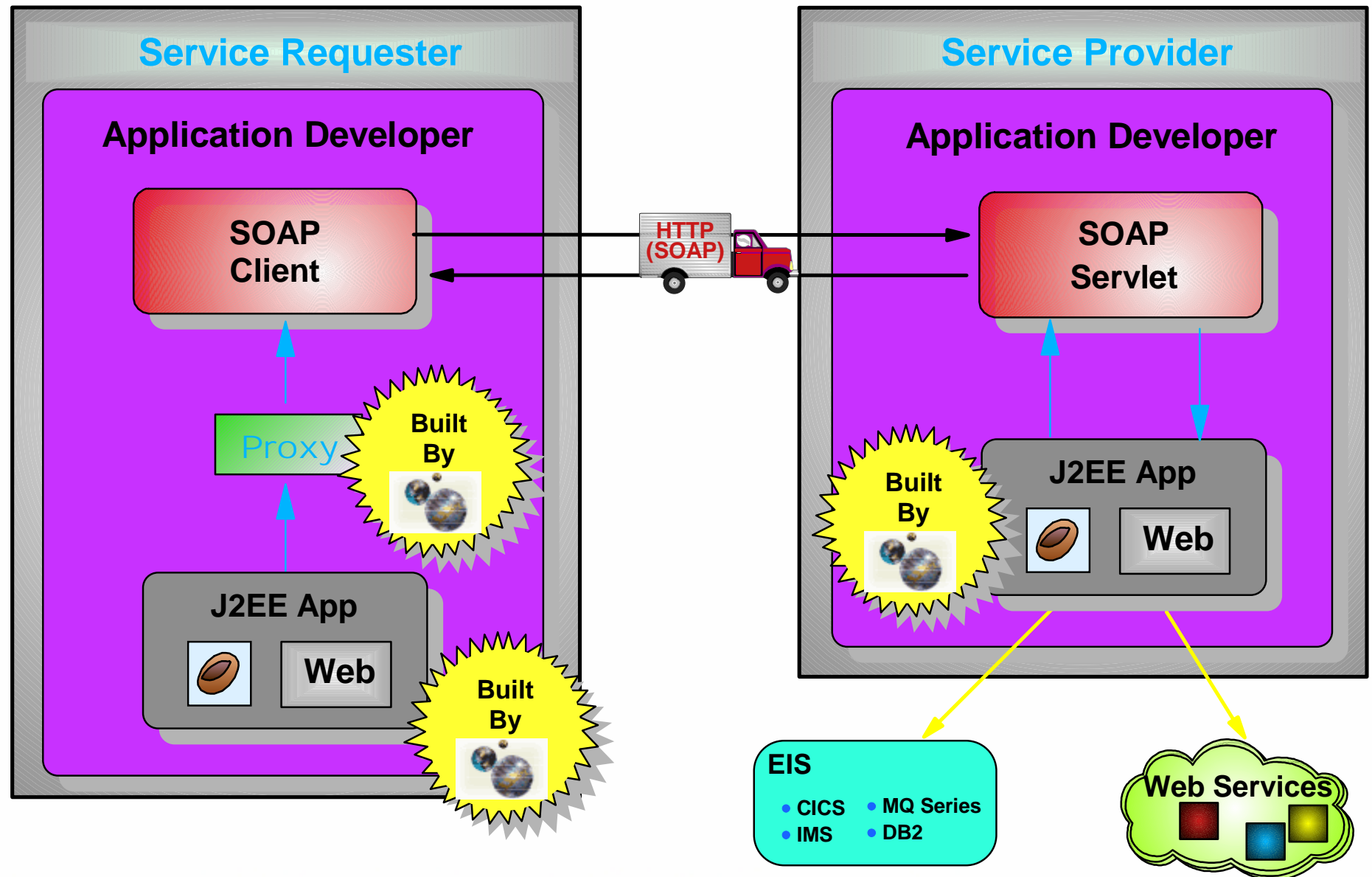
- Generated JavaBean that provides a Java interface to easily invoke the Web service
- J2EE Application that will consume the Web Service
  - ▶ Uses the Proxy Bean to invoke methods on the Web Service



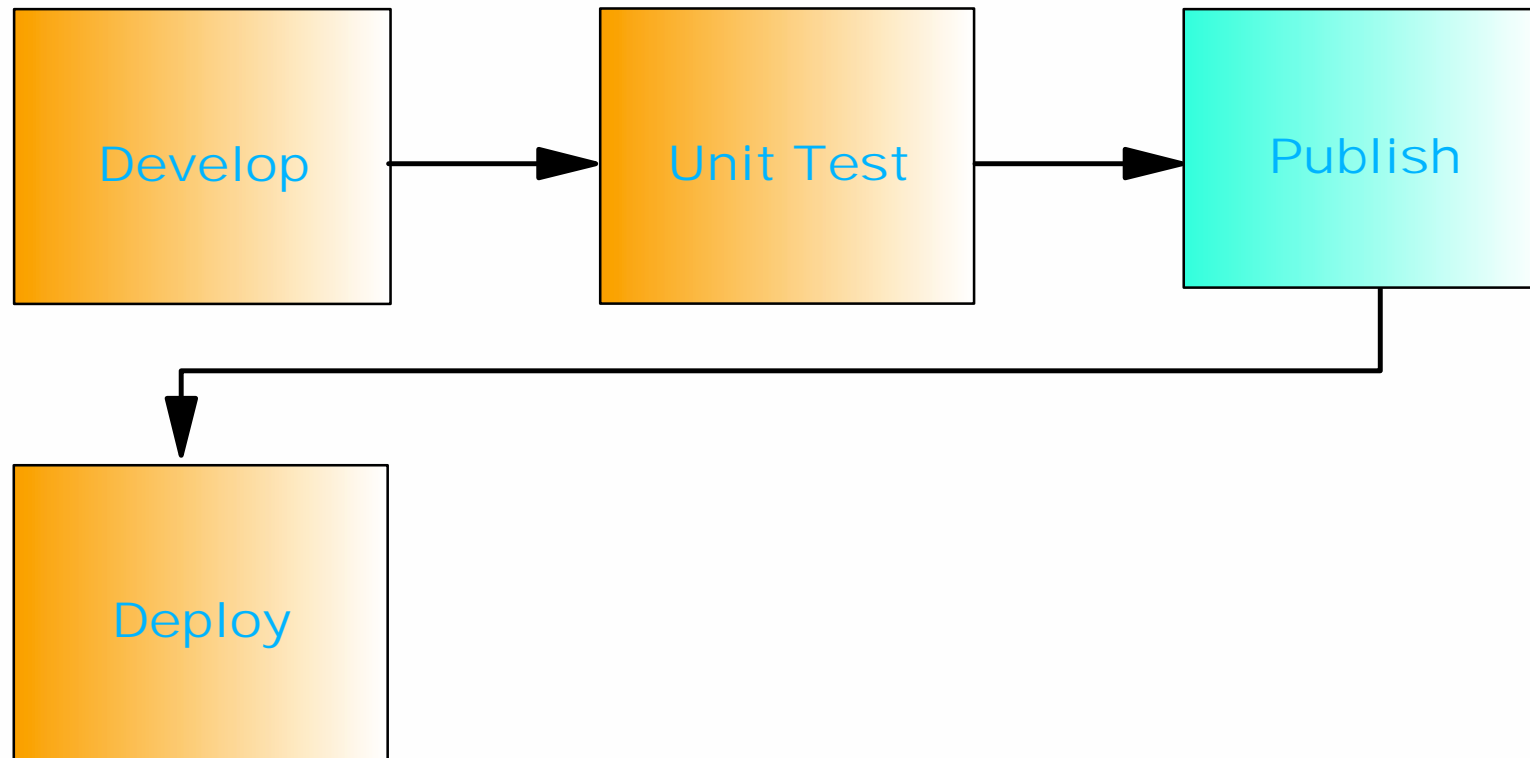
# Runtime View



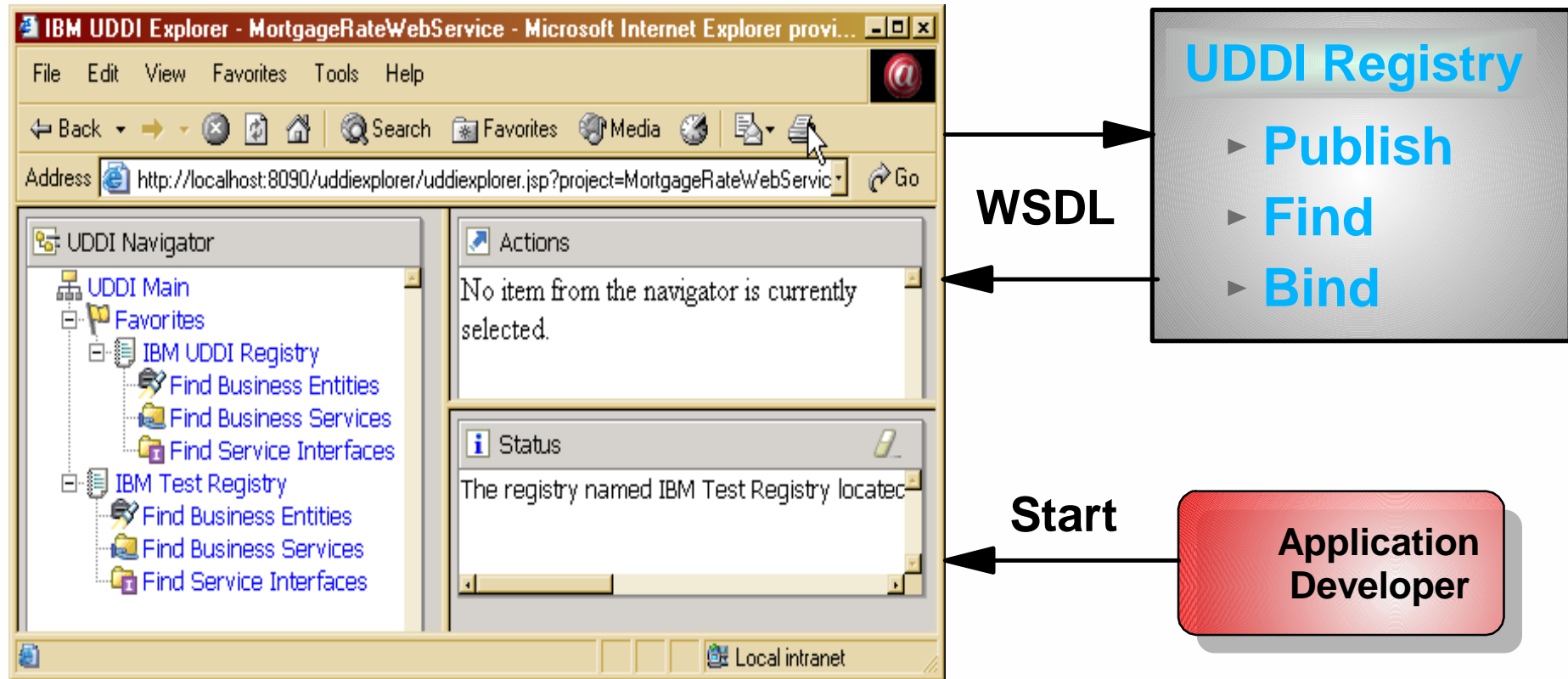
# Runtime View



# Web Service Development Lifecycle



## UDDI Explorer



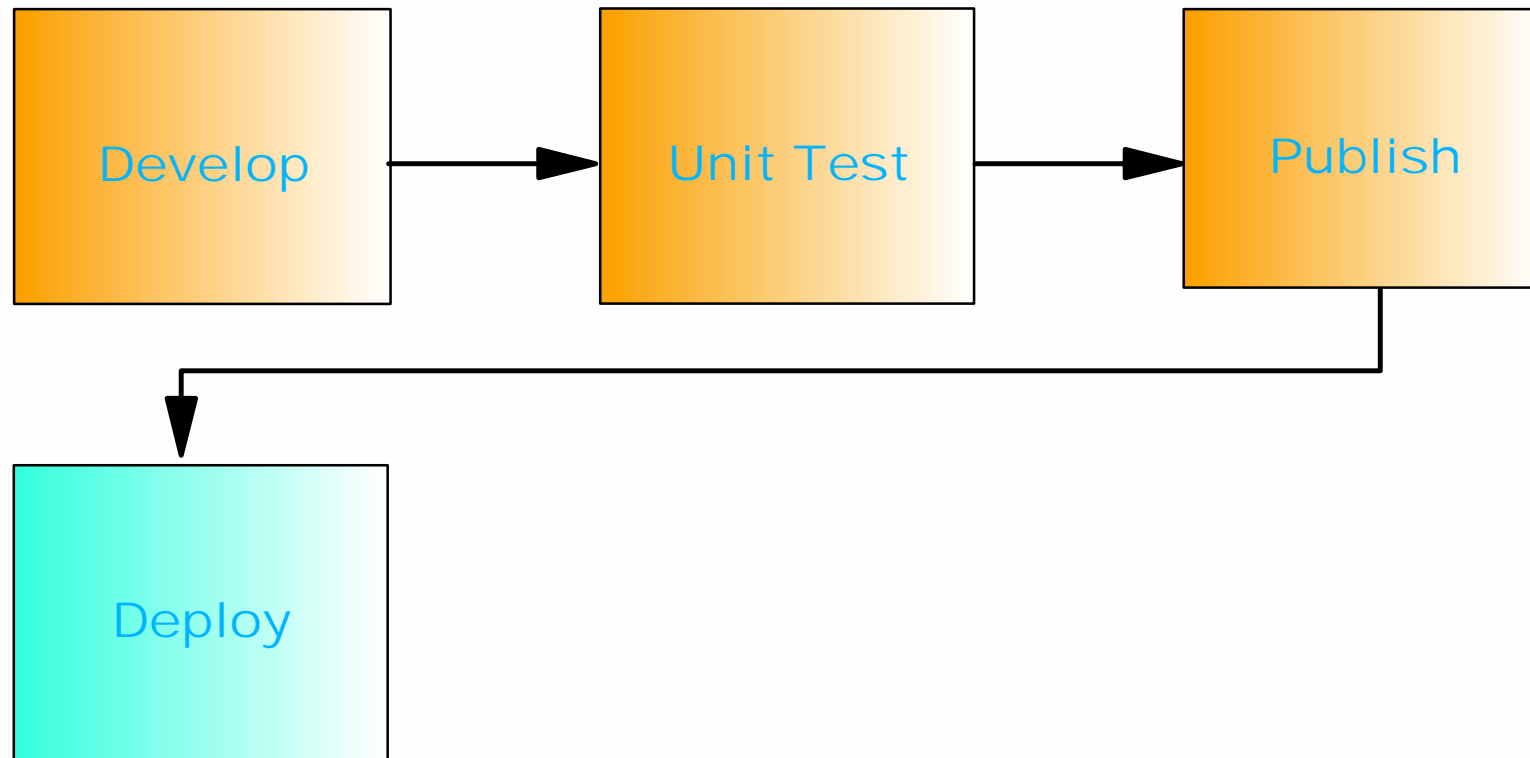
- Web Application that interfaces with UDDI Registry
- Launched from Application Developer



- Brief Introduction to Web Services
  - ▶ Value of Web Services
- Developing Web Services with WebSphere Studio Application Developer
- **Deploying Web Services to WebSphere**
- Enterprise Services



# Web Service Development Lifecycle





# WebSphere Support for Web Services



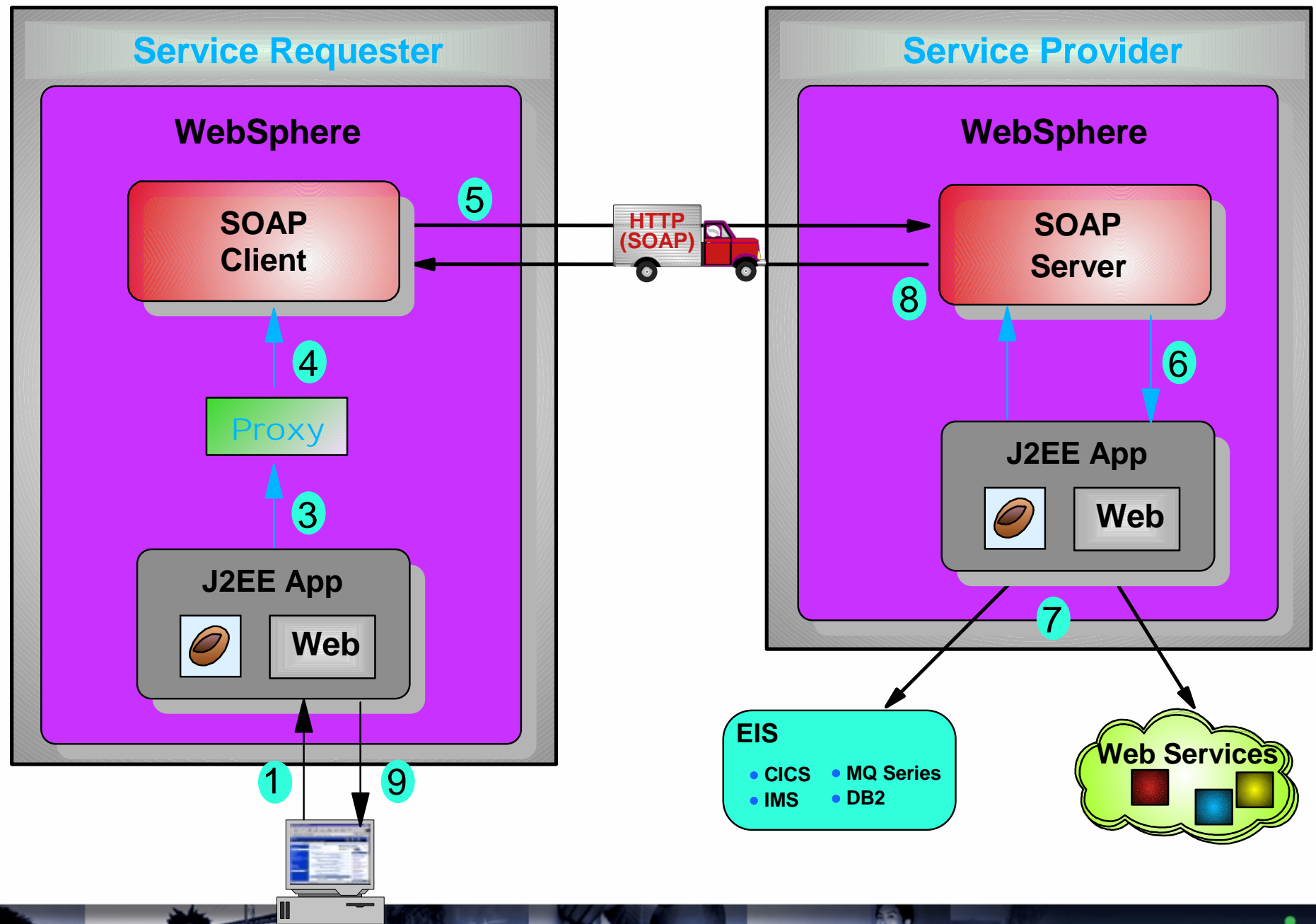
- Apache SOAP Version 2.2
  - ▶ Java-based implementation of SOAP 1.1
- UDDI4J 1.0.3 Open Source UDDI Client
  - ▶ client side of UDDI (everything your application needs to publish, find, and bind a Web service)
- XML Parsers
  - ▶ XML4J Version 3.1, Apache Xerces 1.2.1



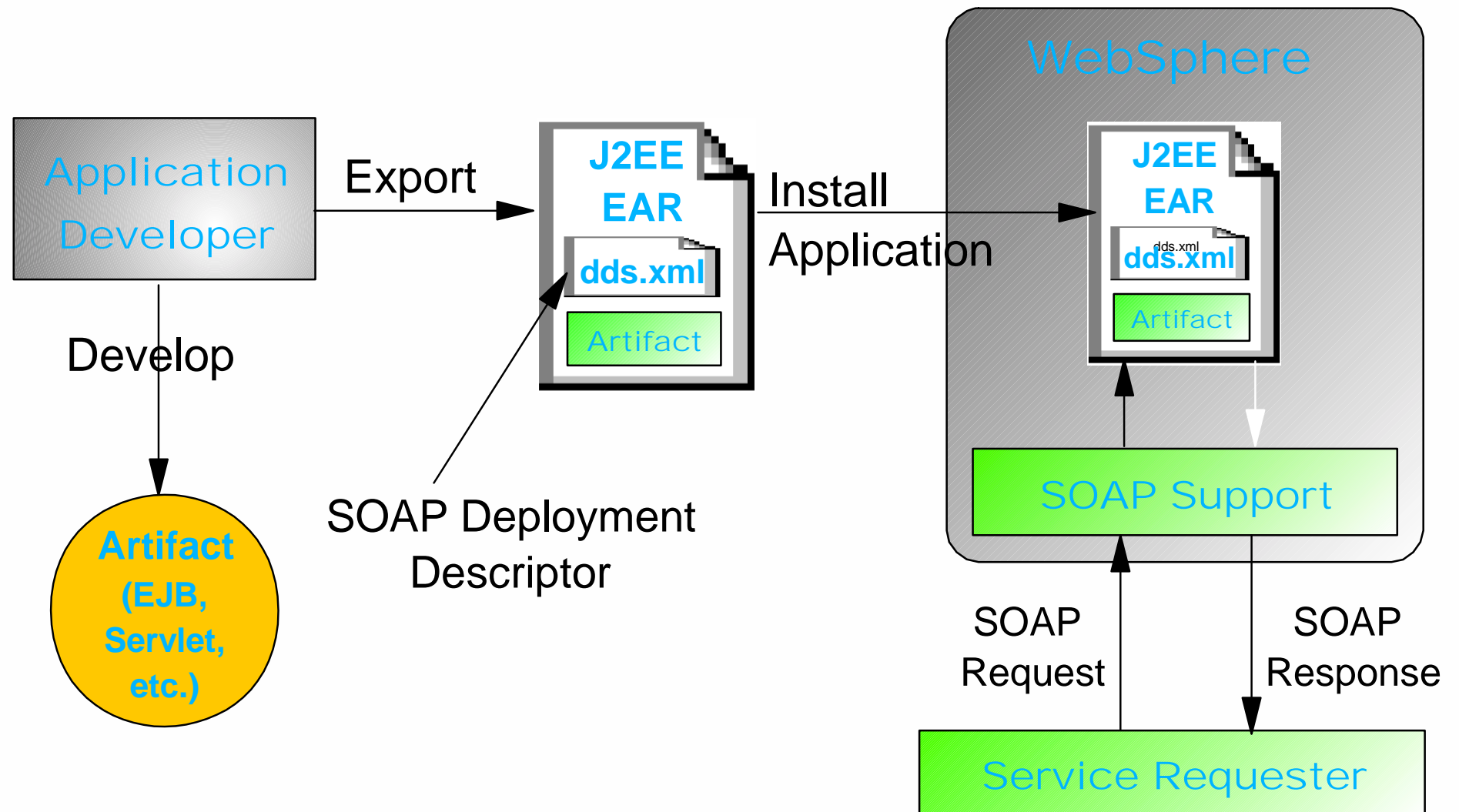
- Tools for deployment of Web (SOAP) services
  - ▶ SOAPEarEnabler
    - Java application that enables a set of SOAP services within an EAR
    - Automatically adds SOAP support (a separate web app with the rpcrouter servlet) to the EAR file
    - Not required if using Application Developer



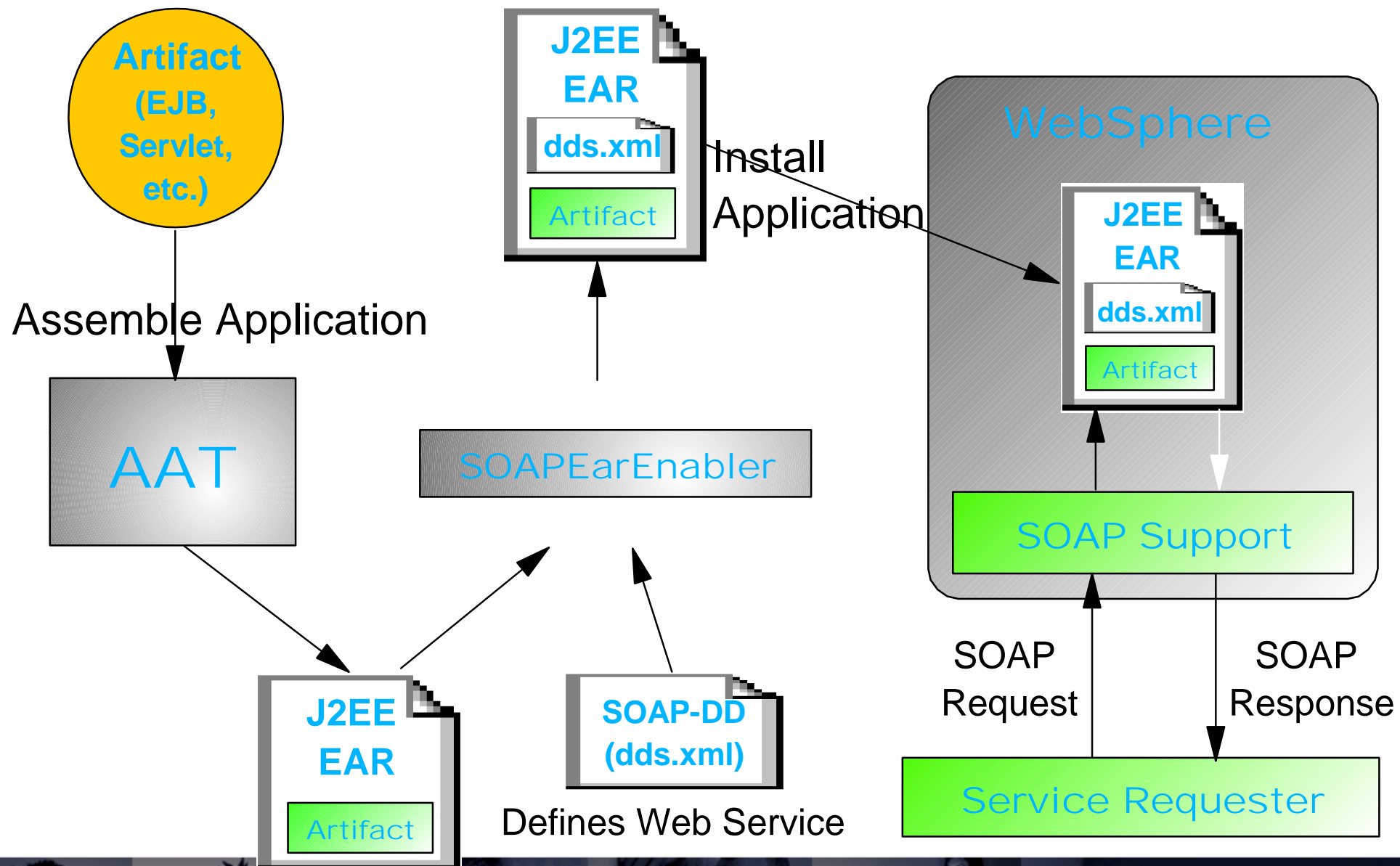
# WebSphere Runtime View



# Deployment to WebSphere from Application Developer



# Web Services Deployment to WebSphere Without Application Developer



- Brief Introduction to Web Services
  - ▶ Value of Web Services
- Developing Web Services with WebSphere Studio Application Developer
- **Deploying Web Services to WebSphere**
- **Enterprise Services**

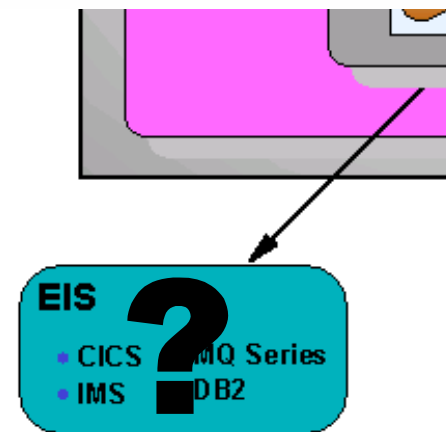




# Enterprise Services



- What if you want to expose a backend system (e.g. CICS, IMS, etc) as a service?
- Enterprise Services can be used to expose backend systems as a service
  - ▶ J2C Resource Adapters provide connectivity to backend system
  - ▶ Enterprise Services can be deployed as a web service
- ▶ WebSphere Studio Application Developer **Integration Edition** provides the tooling support



## Backend Systems

- CICS
- IMS
- HOD
- Other J2C enabled backend





- Web Service represent a movement towards service-oriented applications
- Application Developer represents a productive tool for development of Web Services
- WebSphere represents a scalable, robust, high performing runtime for Web Services

Runtime+Tools=Successful  
Web Services



- Apache SOAP Documentation Users Guide
  - ▶ <http://xml.apache.org/soap/docs/index.html>
- Web Services Description Language
  - ▶ <http://www.w3.org/TR/wsdl>
- UDDI
  - ▶ <http://www.uddi.org/>
- Developer Works
  - ▶ <http://www.ibm.com/developerworks/webservices>



- Redpieces
  - ▶ SG246292 'Web Services Wizardry with WebSphere Studio Application Developer'
  - ▶ SG246176 'IBM WebSphere V4.0 Advanced Edition Handbook'
- IBM UDDI Test Registry
  - ▶ <http://www.ibm.com/services/uddi/>
- Alphaworks
  - ▶ <http://www.alphaworks.ibm.com>
- WebSphere InfoCenter
  - ▶ Article 4.8 'Web Services'
- AXIS
  - ▶ <http://xml.apache.org/axis/>

