E18

Web Enabling IMS Transactions with IMS Connector for Java

Judith Hill

St. Louis, MO **Sept. 30 - Oct. 3, 2002**

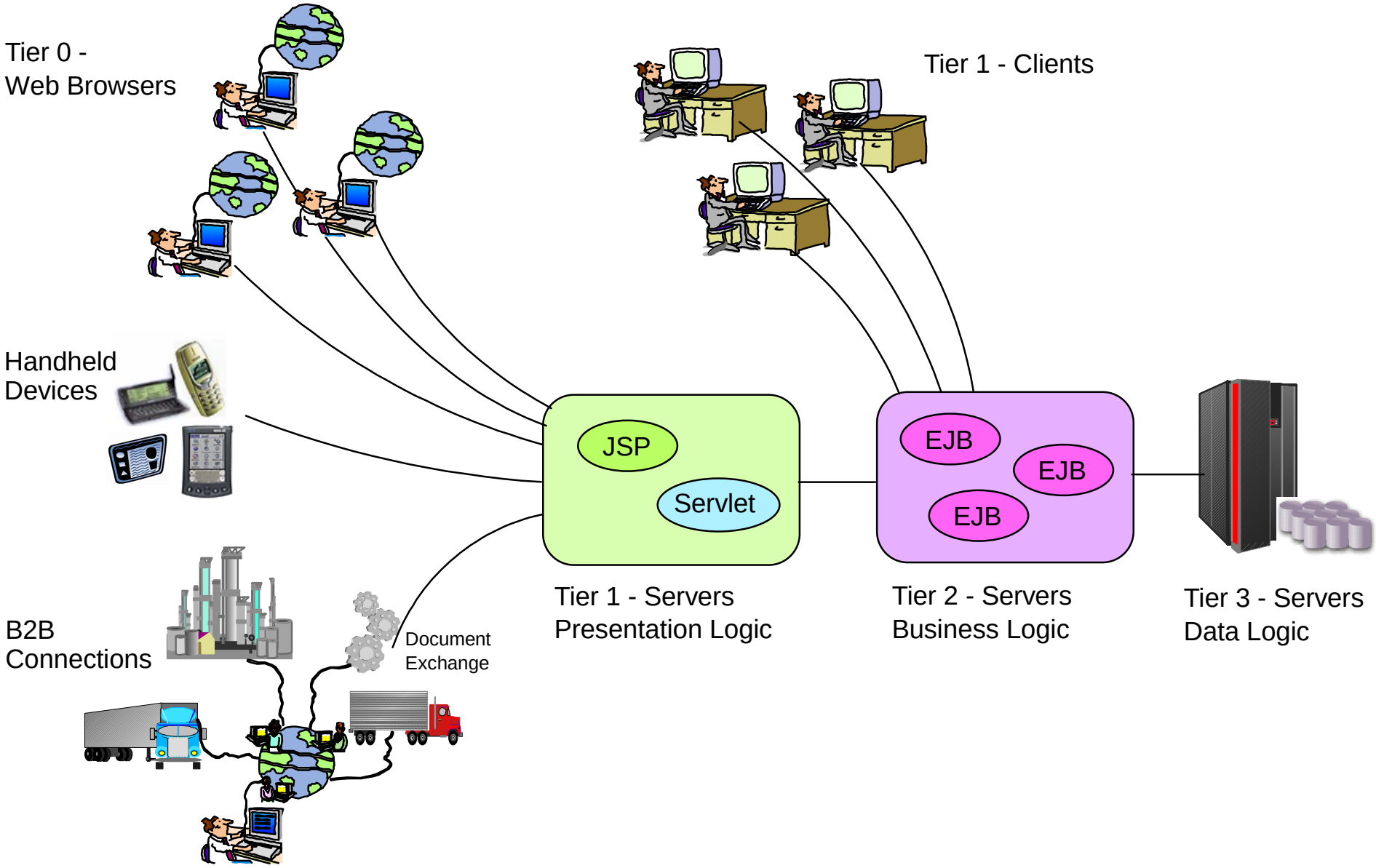
© IBM Corporation 2002

RETURN TO INDEX

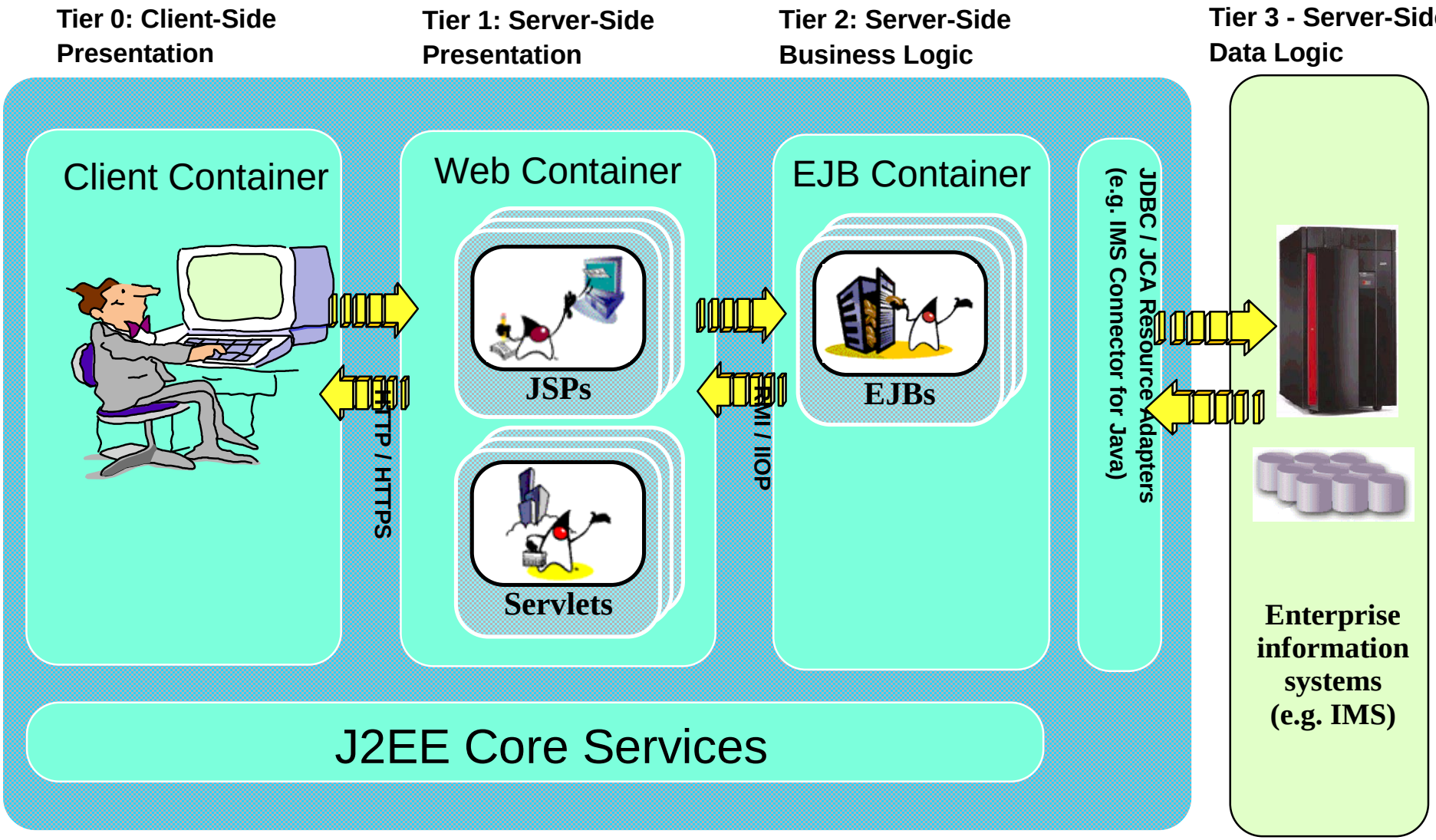
Agenda

- J2EE Connector Architecture
- IMS Connector for Java Overview
- Recent IMS Connector for Java Enhancements
  - Global Transaction (two-phase commit) support
  - Container-Managed Security
  - Services programming model
  - WebSphere Studio Application Developer Integration Edition
    - Demo

Multi-tier Application Environments



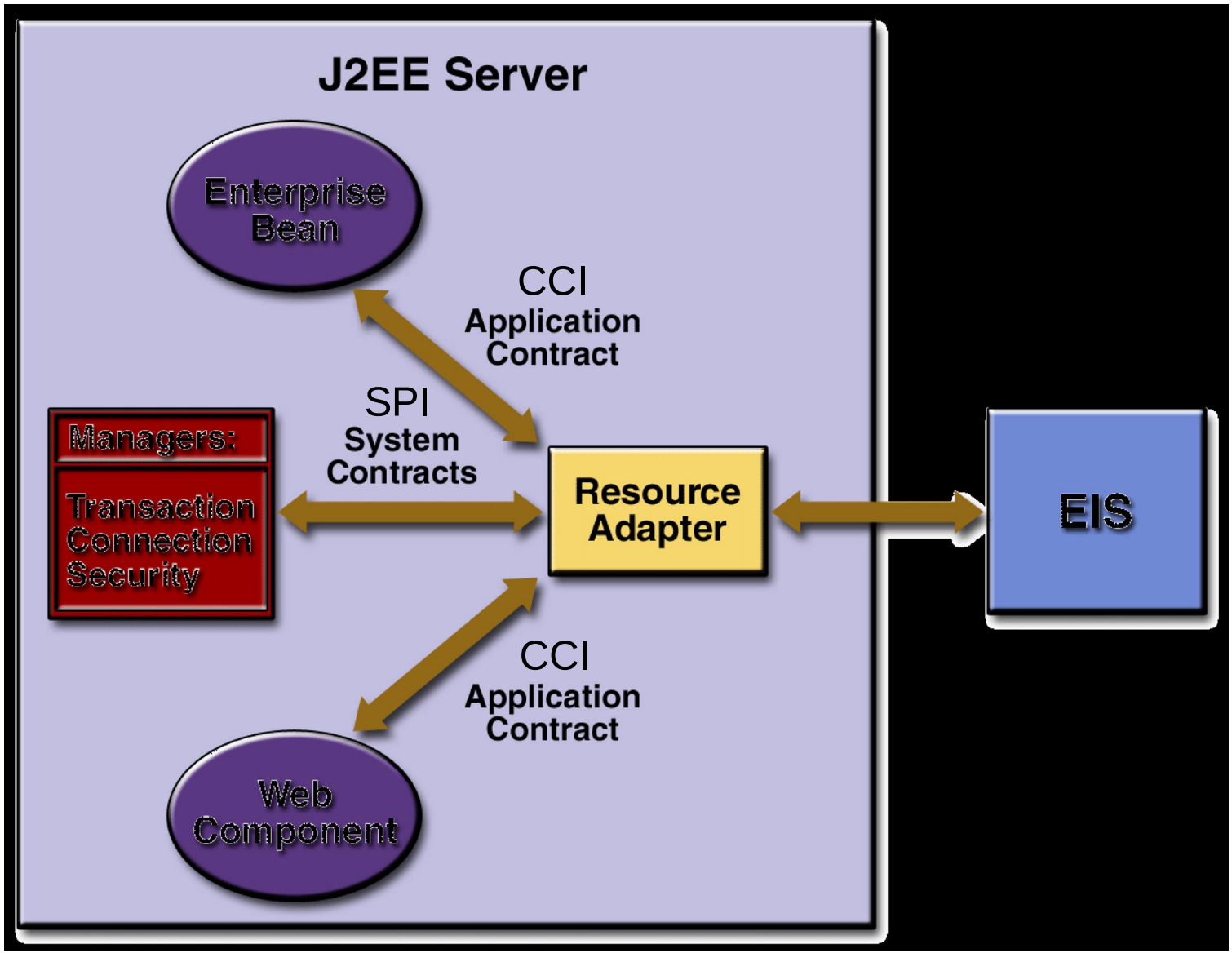
J2EE Server Model



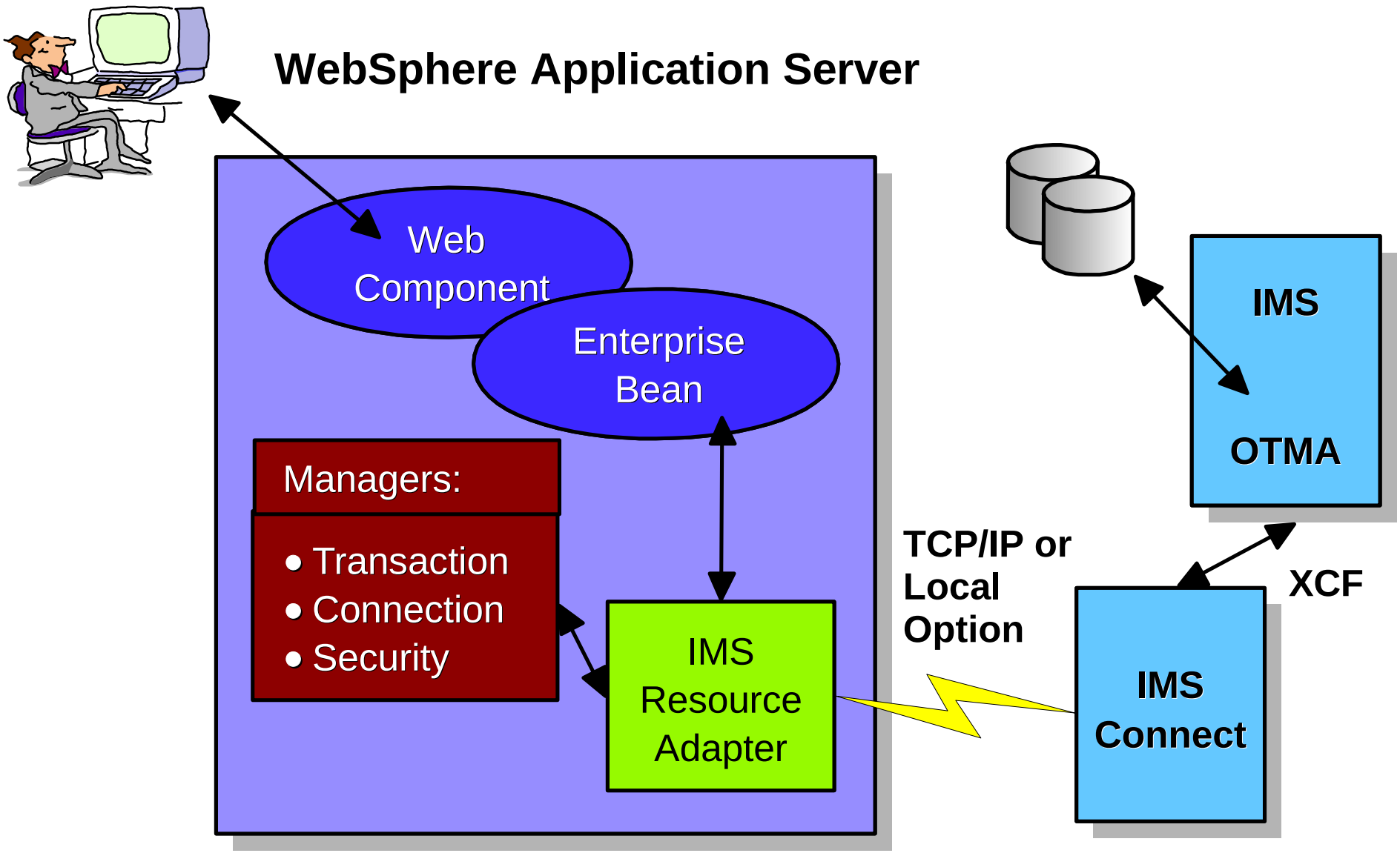
J2EE Connector Architecture

- A standard architecture for Enterprise Information Systems (EIS) integration
- Connects the J2EE platform to heterogeneous EISs
- Defines a common set of interfaces (SPI and CCI) that standardizes interactions between
  - Resource adapters
  - Application servers
  - Application components
- Simplifies application component development
- No longer need to customize a connector for each application server, and vice versa

J2EE Connector Architecture

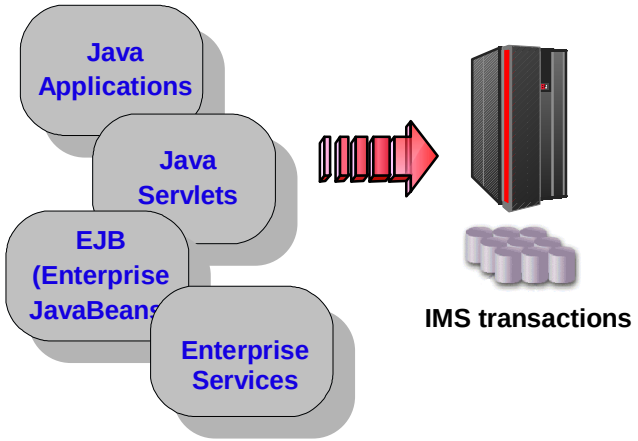


IMS Connector for Java



IMS Connector for Java

Helping IMS Users make the transition to e-business easier



- Offers a highly scalable and flexible topology
- A J2EE Connector Architecture compliant resource adapter
- Used to develop and run J2EE applications that access IMS transactions via **IMS Connect**
- Supports rapid application development with **WebSphere Studio Application Developer Integration Edition**
- Runs in **WebSphere Application Server** on both z/OS and distributed platforms

Recent Releases

Version	Description	Development Environment	Runtime Environment
1.2.0/1.2.1(NLV)	• Adds J2EE Connector Architecture support for distributed platforms	VisualAge for Java 4.0	WAS 4.0 for Windows, AIX, Solaris, Linux for z/OS
1.2.2/1.2.3 (NLV)	• Adds J2EE Connector Architecture support for z/OS platform • Adds Two-Phase Commit Support for z/OS platform (via Local Option)	VisualAge for Java 4.0	WAS 4.0.1 for z/OS
1.2.4/1.2.5 (NLV)	• Adds web services support for distributed platforms	WSAD IE 4.1	WAS EE 4.1 for Windows, AIX, Solaris

Recent IMS Connector Enhancements

- Global Transaction (two-phase commit) support
- Container-Managed Security
- Services programming model
- A new development environment:
  - *WebSphere Studio Application Developer Integration Edition*

What is a Transaction ?

- A transaction is a group of changes that are either fully completed or fully rolled back.
  - For example, a financial program might transfer funds from a checking account to a savings account with the following steps:
    - Begin Transaction
      - Debit checking account
      - Credit savings account
    - Commit Transaction
  - Either both of these steps must complete, or none of them

Types of Transactions

- There are two types of transactions:
  - Global
    - One or more resource managers are involved in the group of changes
    - An external coordinator is used to coordinate the changes
    - If more than one resource manager is involved in a transaction, two phase commit processing is used to coordinate the changes
  - Local
    - Only one resource manager is involved in the group of changes
    - The resource manager is responsible for coordinating the changes
- J2EE applications
  - Use JTA (Java Transaction API) interfaces to demarcate Global transactions
  - Use CCI LocalTranasction to demarcate Local transactions

IMS Connector for Java Transaction Support

- Allows applications to use JTA interfaces to participate in global transaction and two-phase commit processing
- The initial transaction offering is available with IMS Connector for Java version 1.2.2
  - Exploits WebSphere Application Server on z/OS and RRS
  - Via Local Option connection only
  - IMS Connect, WebSphere Application Server and IMS must reside in the same MVS image

IMS Connector for Java Transaction Support

Transfer $100 from Checking to Savings

z/OS

WebSphere Application Server for z/OS

Web Component

Enterprise Bean

1,4

Transaction Manager

IMS Resource Adapter

1. Begin Transaction

2. Debit $100 from Checking

3. Credit $100 to Savings

4. Commit Transaction

2

3

Resource Recovery Services (RRS)

Local Option

IMS Connect

IMS

Checking Account DB

DB2

Savings Account DB

JCA Security Options

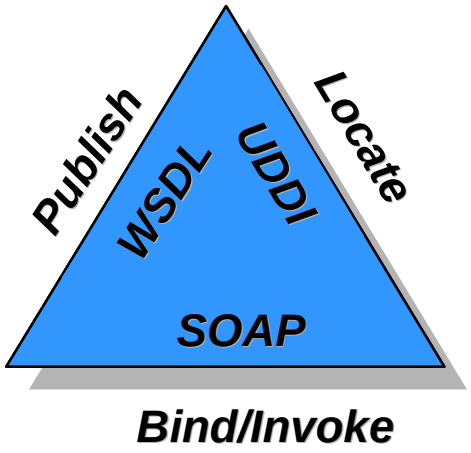
- When an application component requests a connection to an EIS resource, the EIS may require a security sign-on
- An application has two choices for EIS Sign-on
  - **Container-Managed**
    - The container (J2EE Server) provides the security information to the resource adapter for EIS Sign-on
    - The security information is provided to the J2EE server using the J2EE server's administrative/deployment tool
  - **Component-Managed**
    - The component (J2EE application) provides the security information to the resource adapter for EIS Sign-on
    - The security information is provided in the application code

IMS Connector for Java Container-Managed Security Support

- Allows the application to use Container-Managed Sign-on in WebSphere Application Server for z/OS
  - Easier to maintain security information
  - No need to recompile code when changing security information
- Available with IMS Connector for Java version 1.2.2
  - Via Local Option connection only

What are Web Services ?

- "Web services" is how businesses describe fuctionality (services) they want to externalize
- Web sevices allow businesses to
  - **Publish** services across the Web
  - **Locate** services across the Web
  - **Invoke** services across the Web

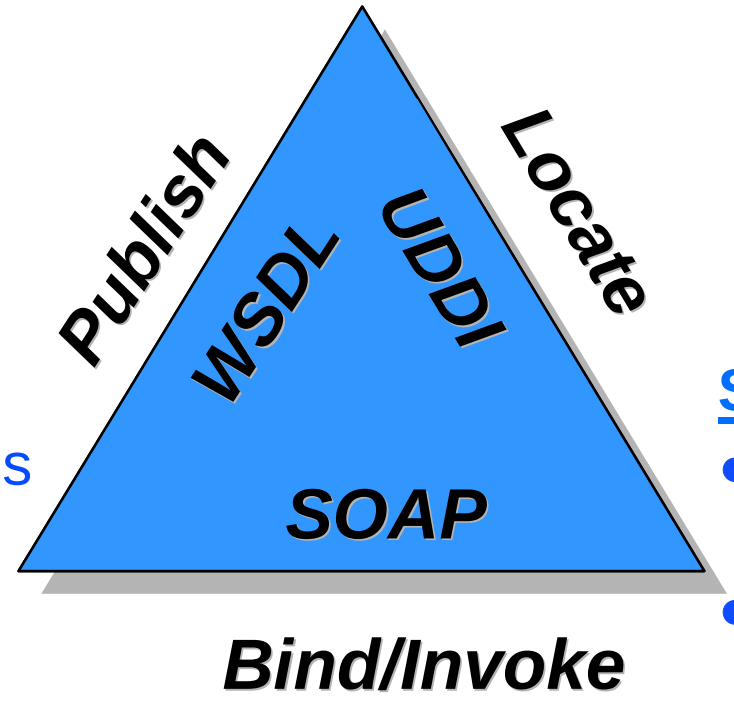


- Web services use simple, standard interfaces for
  - Publishing and discovering services
  - Communicating between service provider and service requester in a platform-independent way

Key Functions of Web Services

Service Broker

- A searchable repository of service descriptions
- Service Providers publish their services
- Service Consumers find services



Service Provider

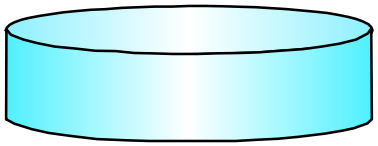
- Provides applications as Web Services
- Publishes their services

Service Consumer

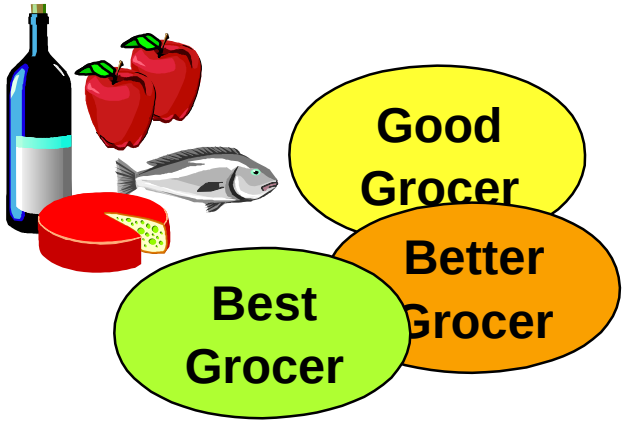
- A client that needs a service
- Publishes their needs

Why Web Services ?

Web Services allow applications that run on different platforms, and developed with different technologies to easily interact with each other in a loosely coupled integration



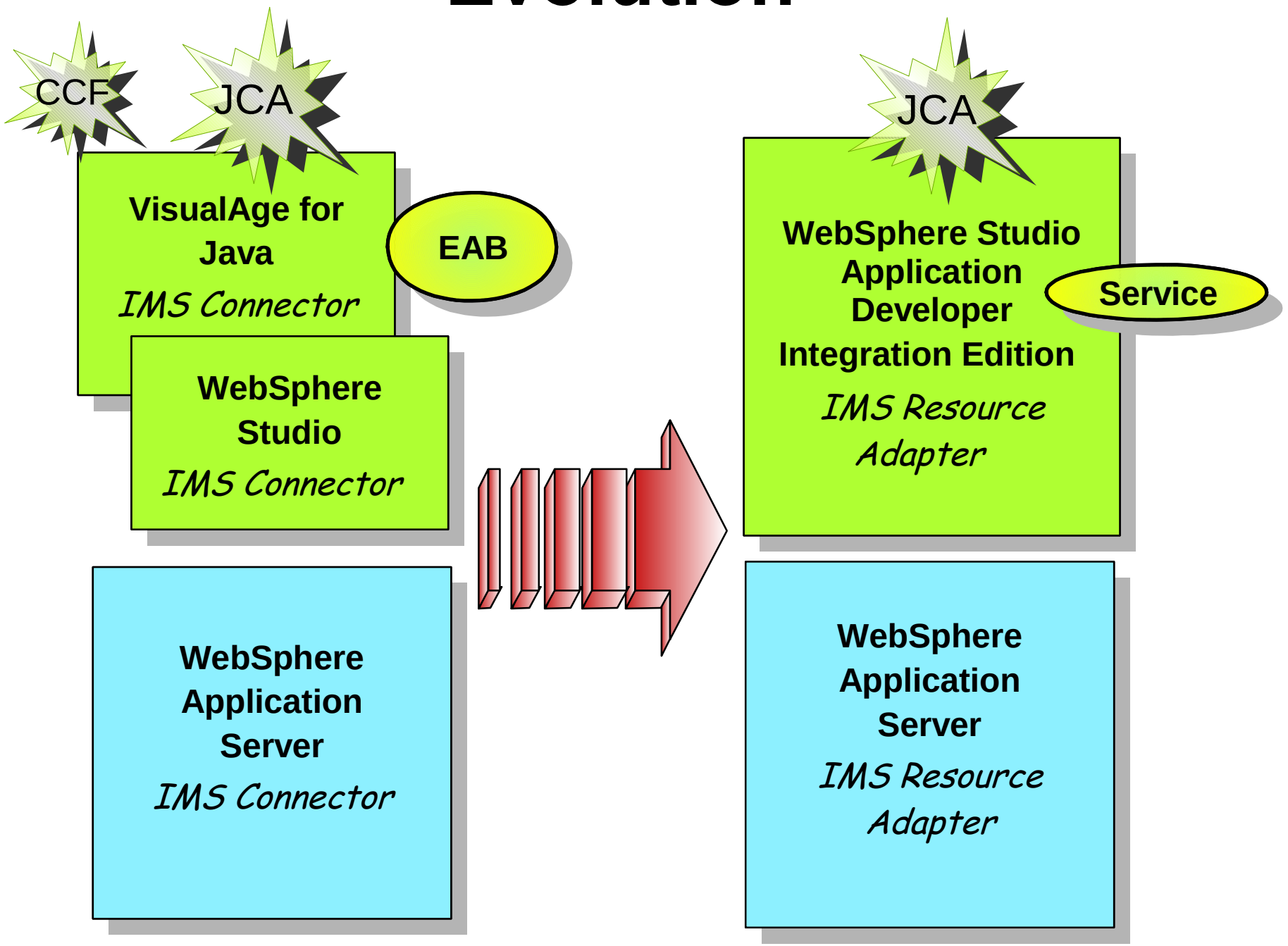
Service Broker



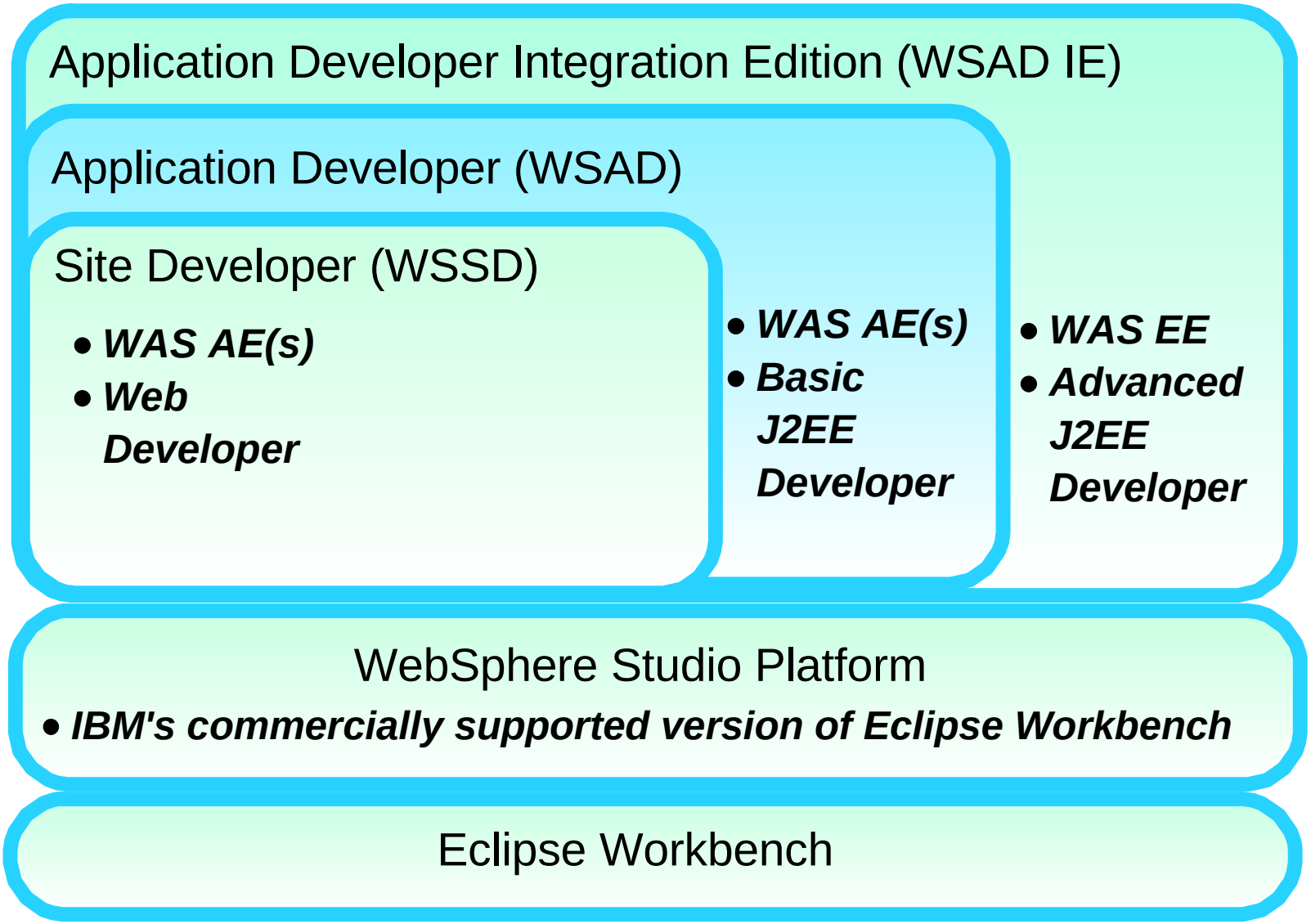
Service Provider

Service Consumer

Evolution

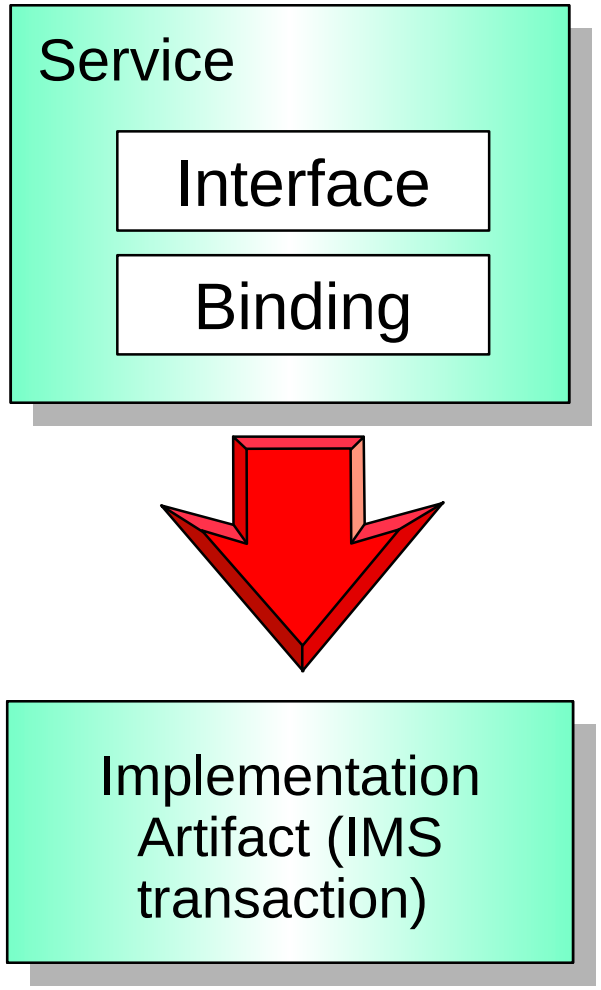

CCF
JCA
VisualAge for Java
IMS Connector
EAB
WebSphere Studio
IMS Connector
WebSphere Application Server
IMS Connector
JCA
WebSphere Studio Application Developer Integration Edition
IMS Resource Adapter
Service
WebSphere Application Server
IMS Resource Adapter

WebSphere Tooling

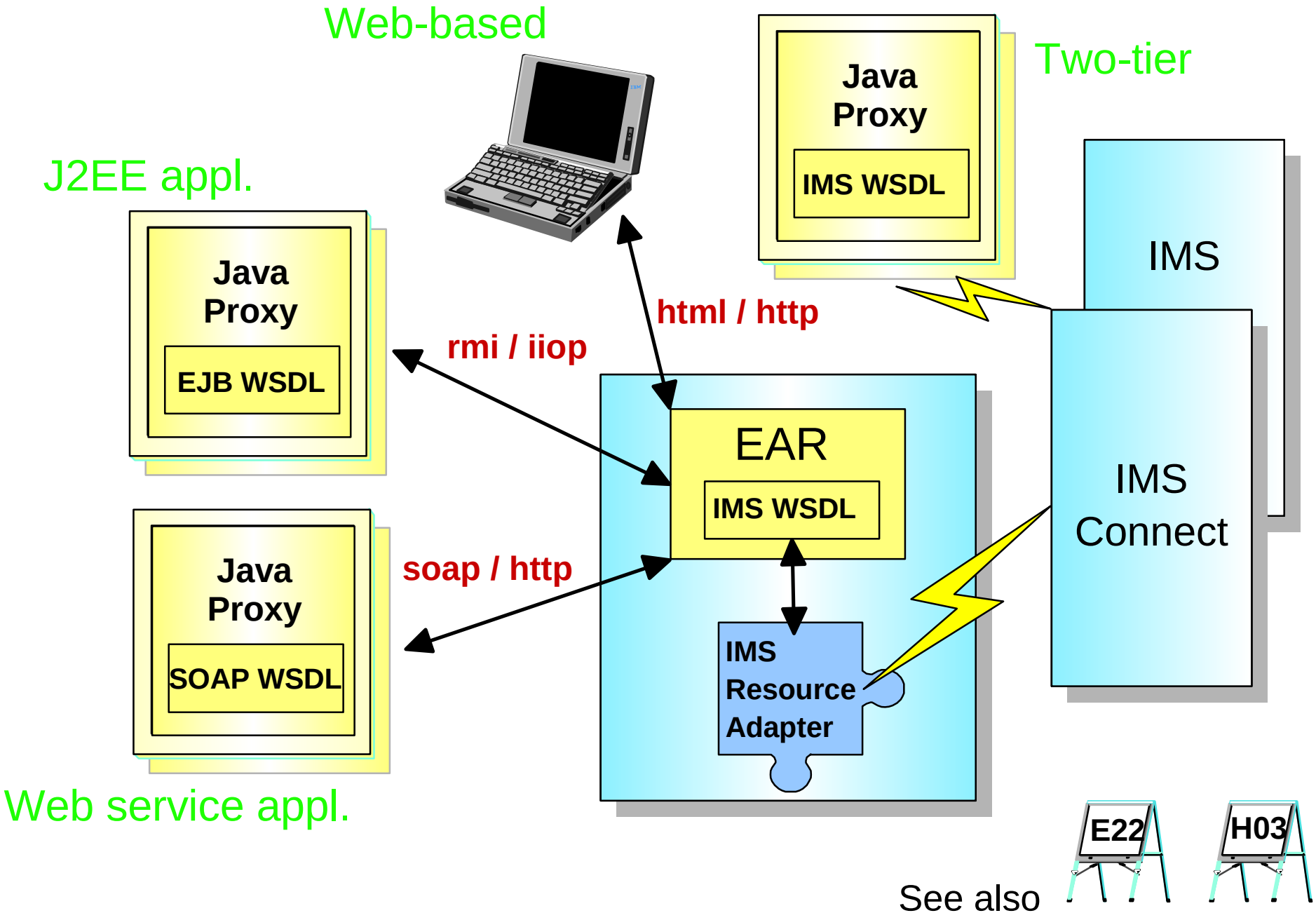


Enterprise Services

- WSAD IE is based on the concept of **Enterprise services**
- A service consists of three parts:
  - **Implementation** artifact
    - IMS transactions, EJBs, ...
  - **Interface**
    - Describes the available ***operations*** and the ***messages*** they exchange
  - **Binding**
    - Describes how to get to the implementation artifact
- Services are described by WSDL documents

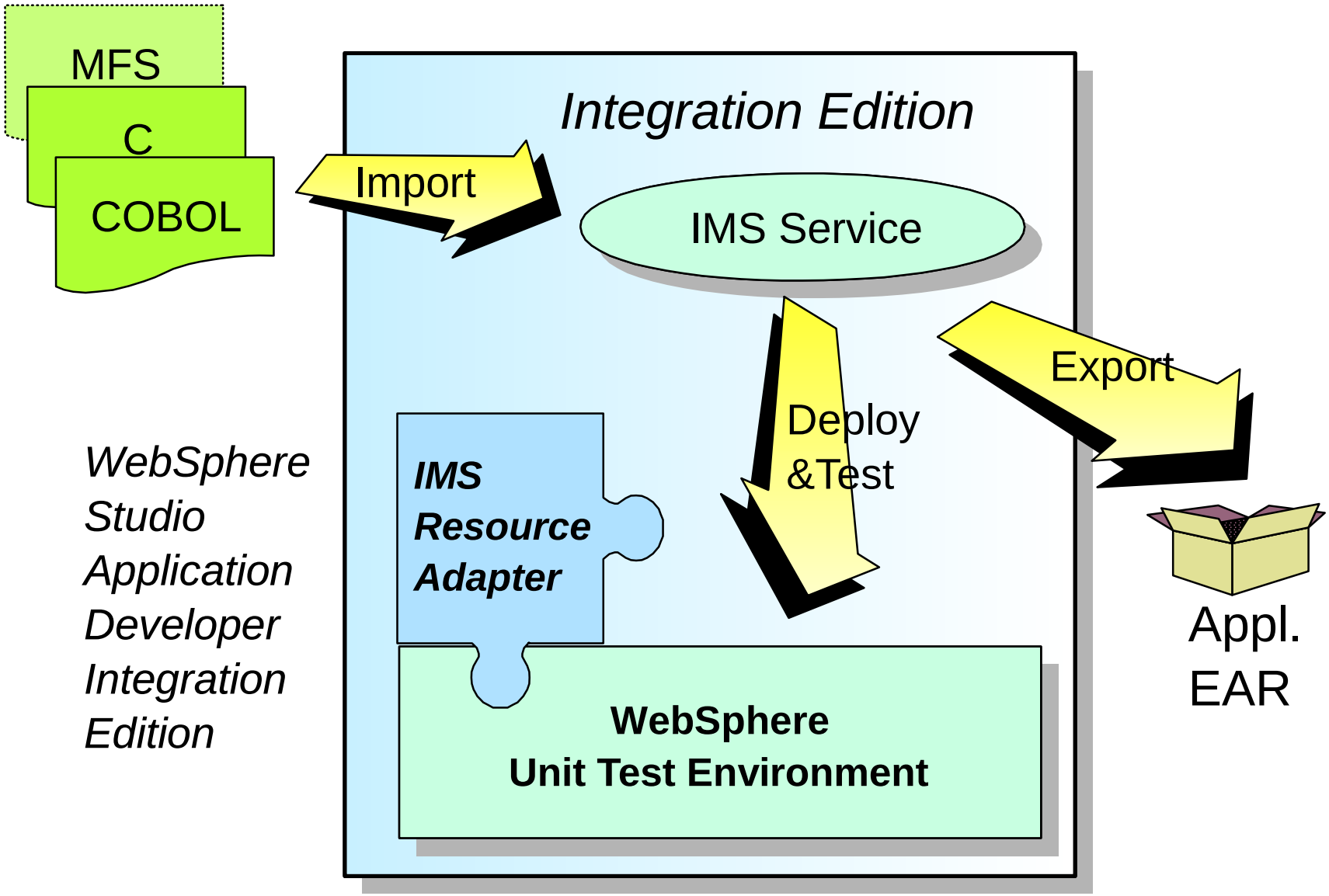


Application Models

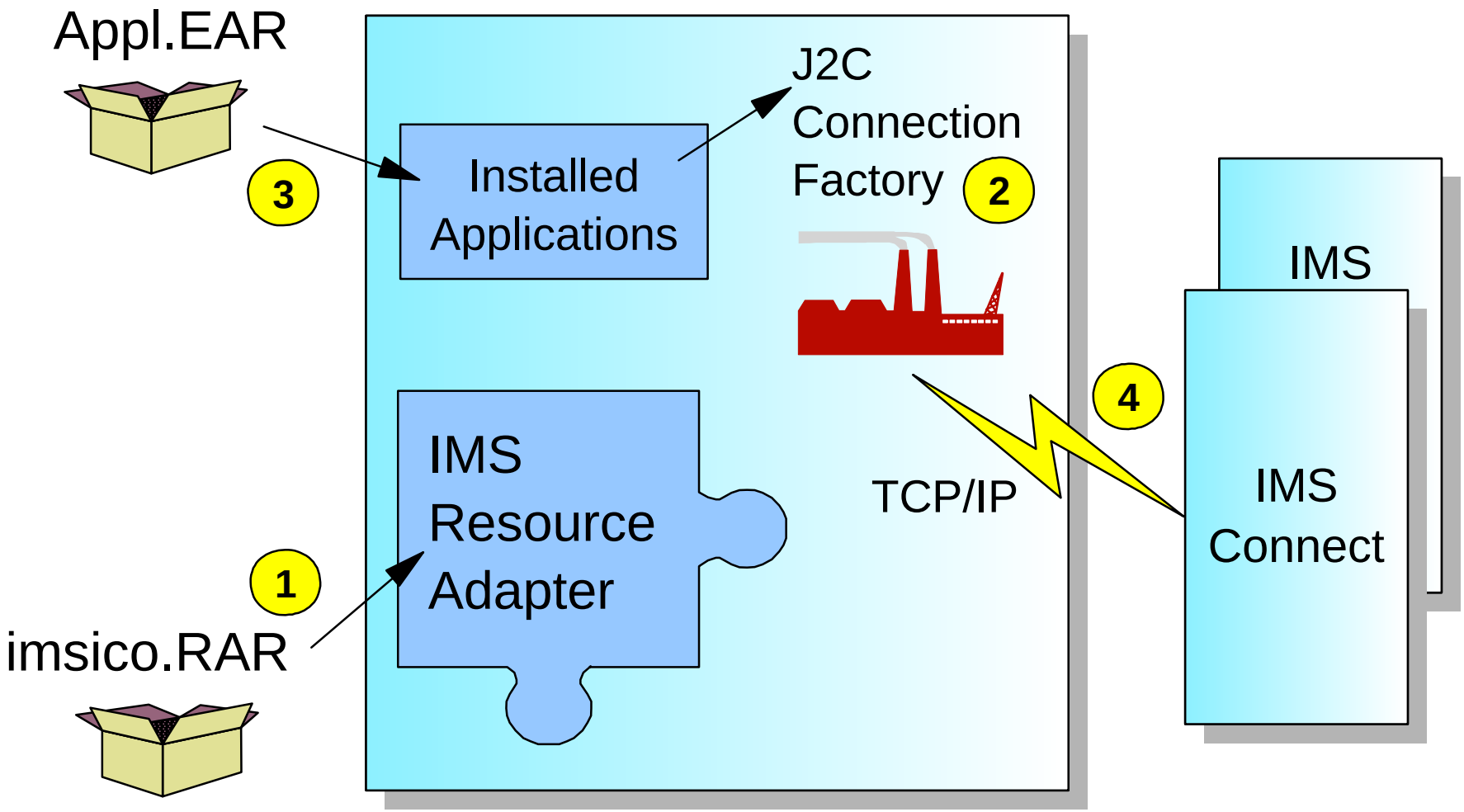


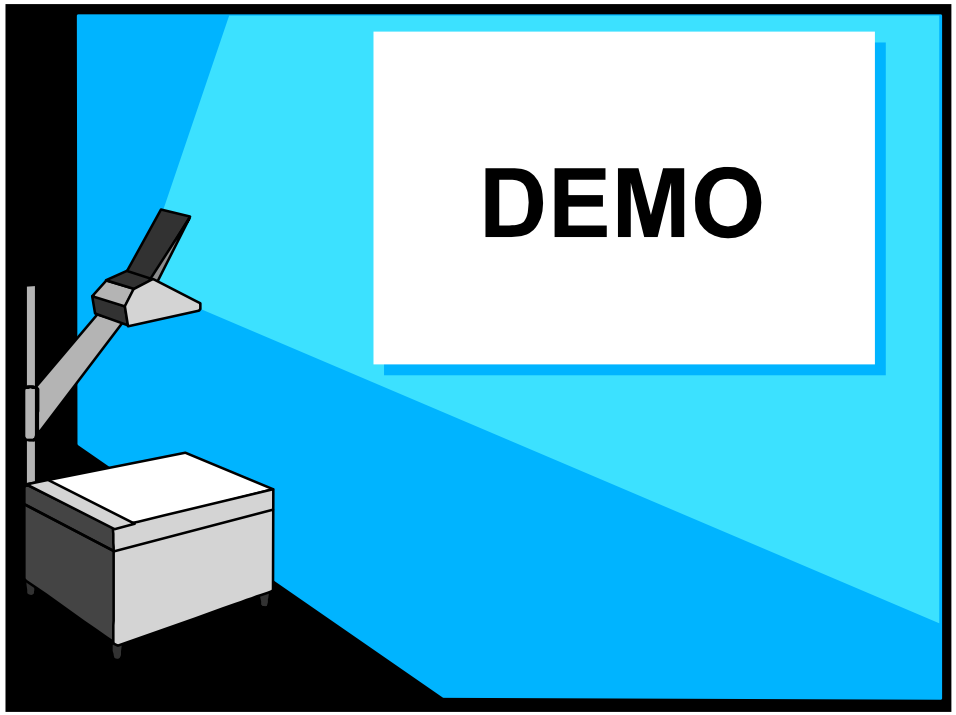
See also E22 H03

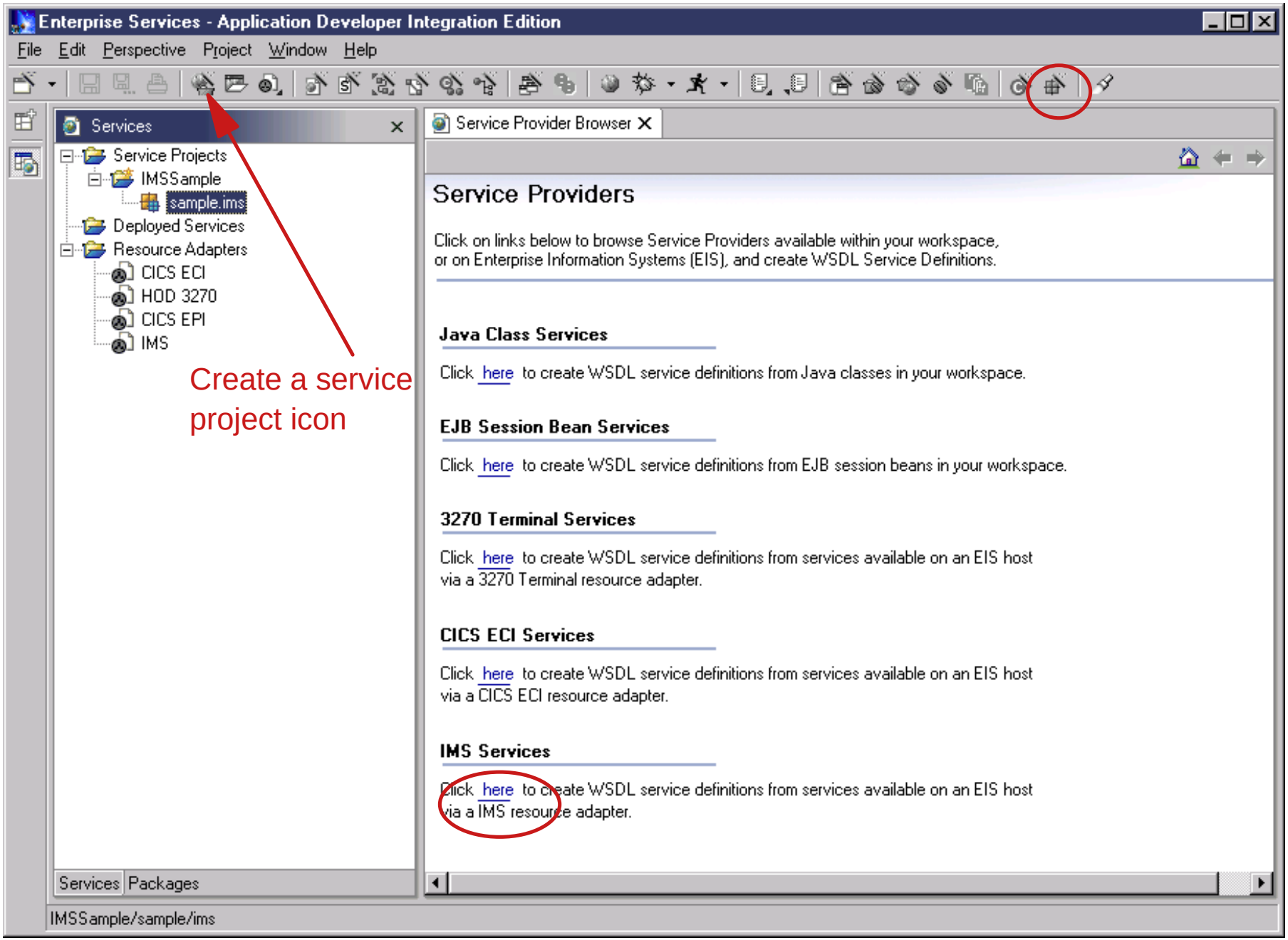
WSAD IE Development

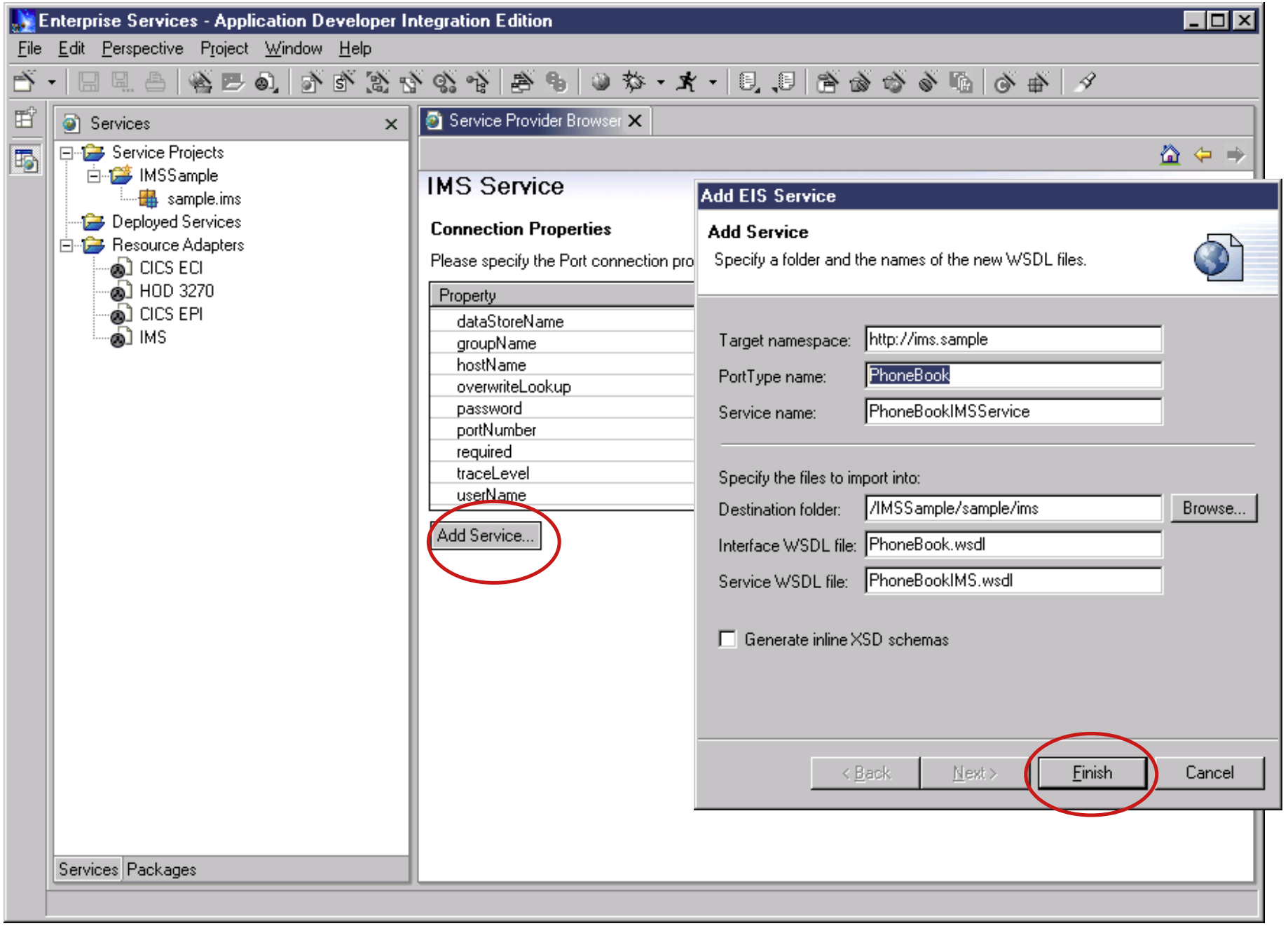


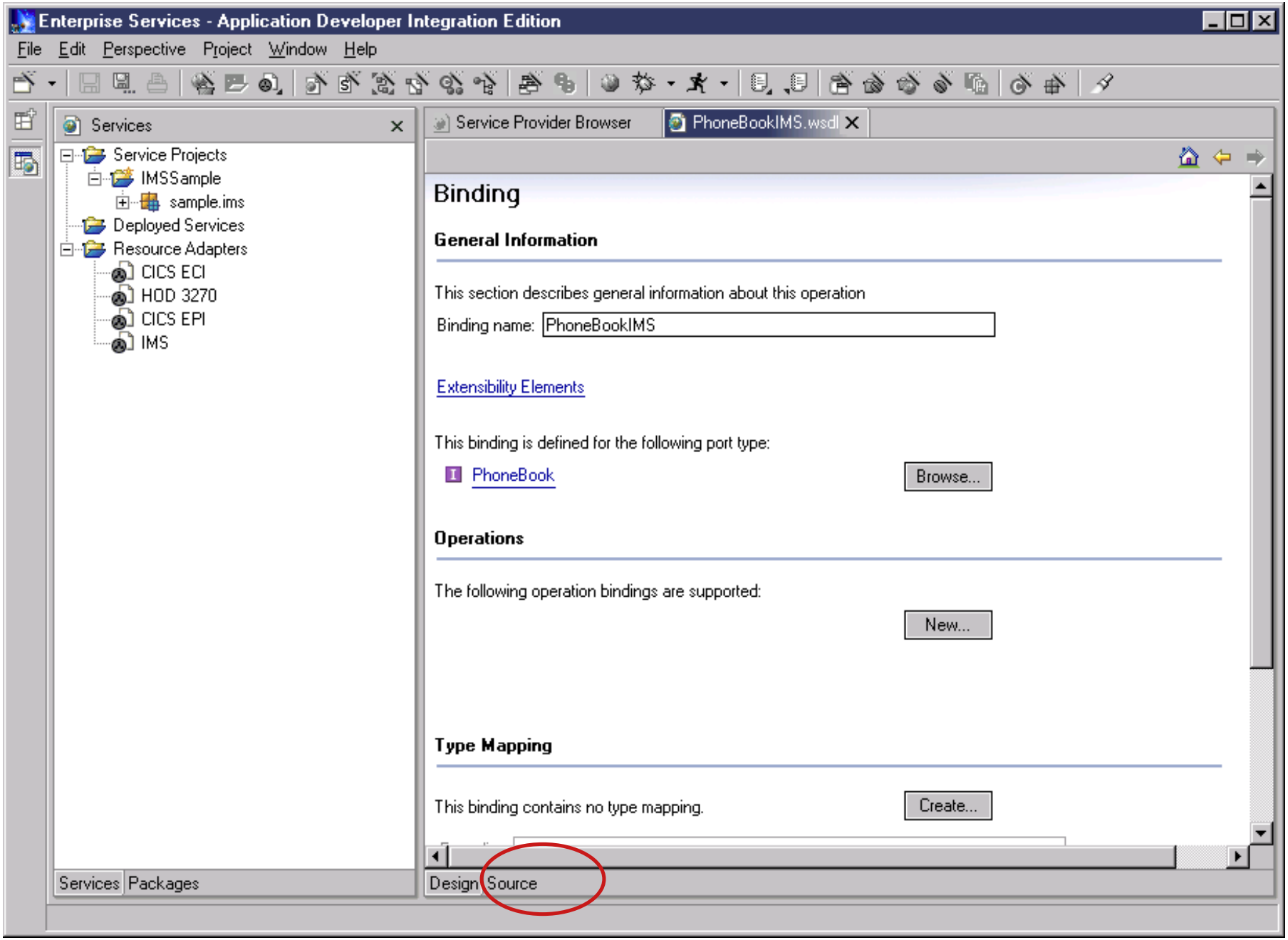
WebSphere Application Server Runtime

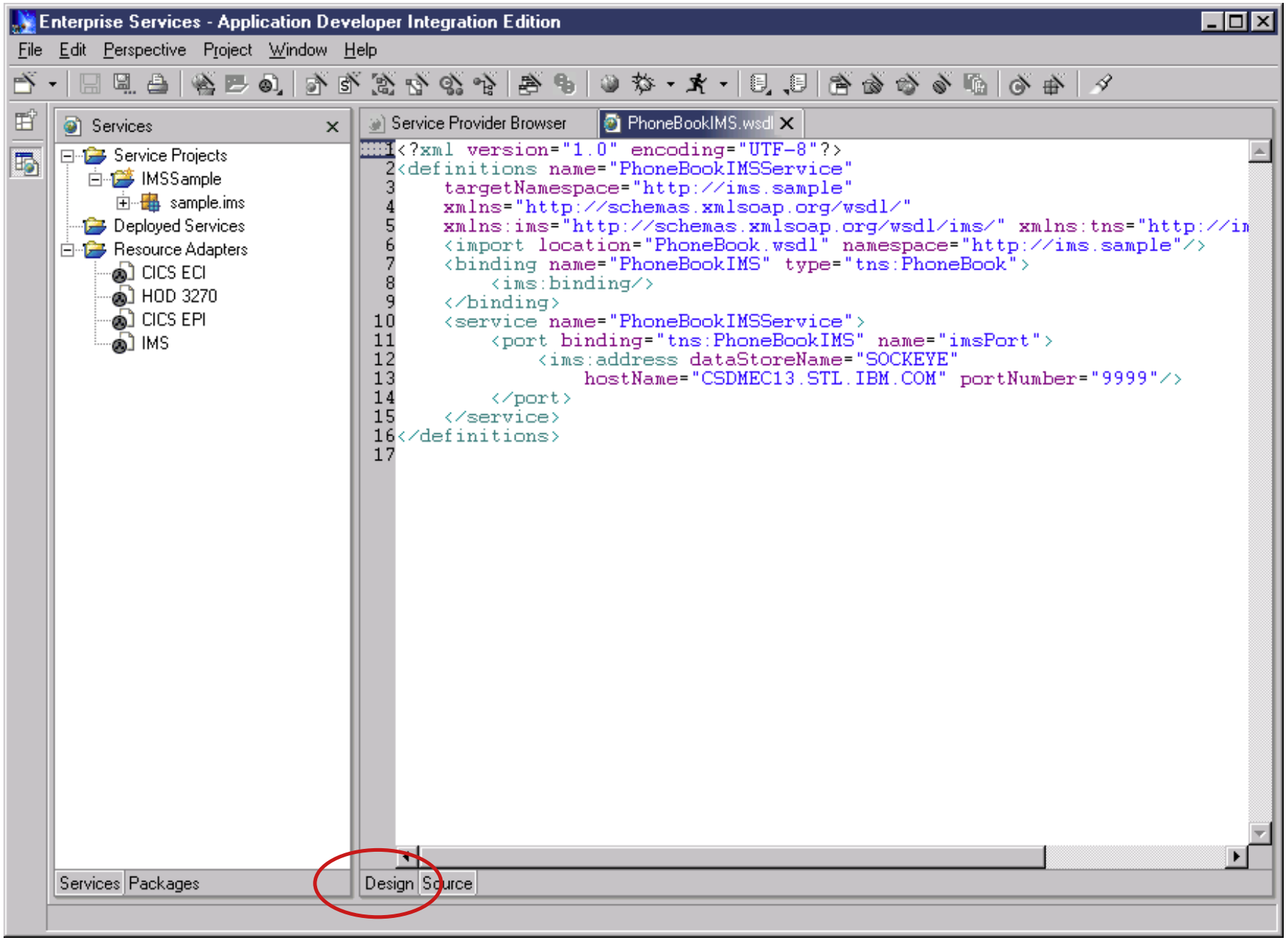


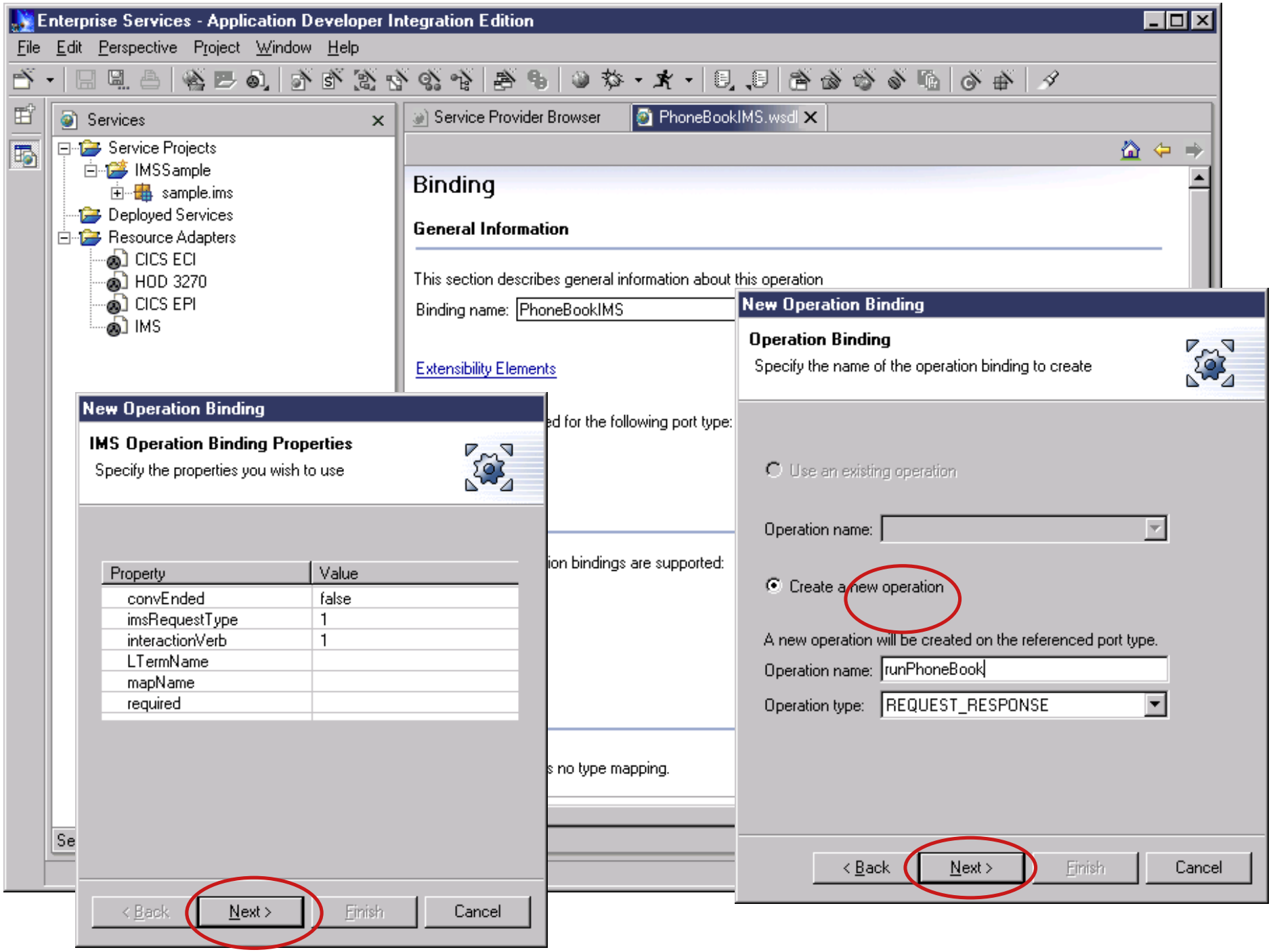

DEMO

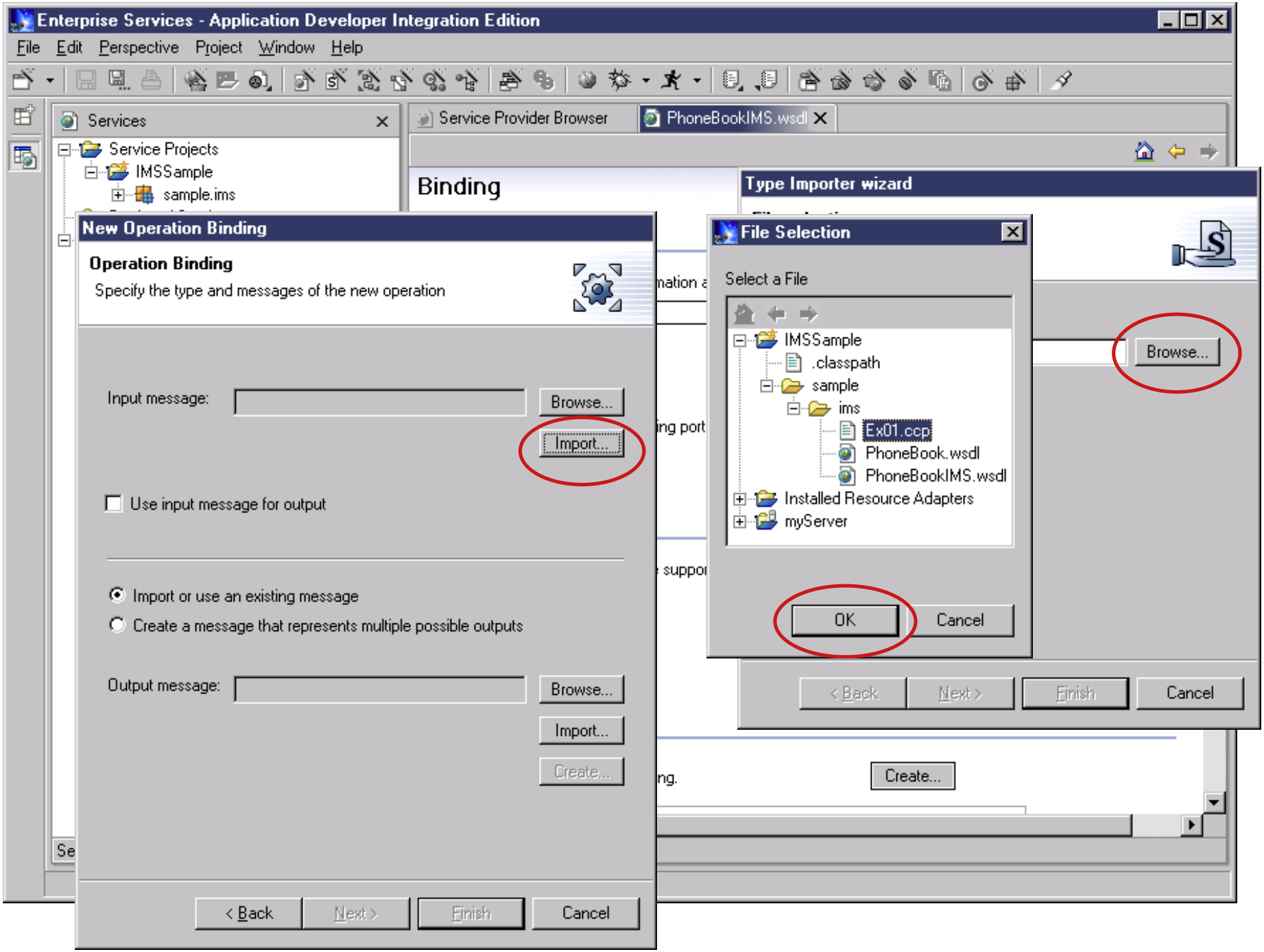

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Create a service project icon
Service Provider Browser
Service Providers
Click on links below to browse Service Providers available within your workspace, or on Enterprise Information Systems (EIS), and create WSDL Service Definitions.
Java Class Services
Click here to create WSDL service definitions from Java classes in your workspace.
EJB Session Bean Services
Click here to create WSDL service definitions from EJB session beans in your workspace.
3270 Terminal Services
Click here to create WSDL service definitions from services available on an EIS host via a 3270 Terminal resource adapter.
CICS ECI Services
Click here to create WSDL service definitions from services available on an EIS host via a CICS ECI resource adapter.
IMS Services
Click here to create WSDL service definitions from services available on an EIS host via a IMS resource adapter.
Services Packages
IMSSample/sample/ims

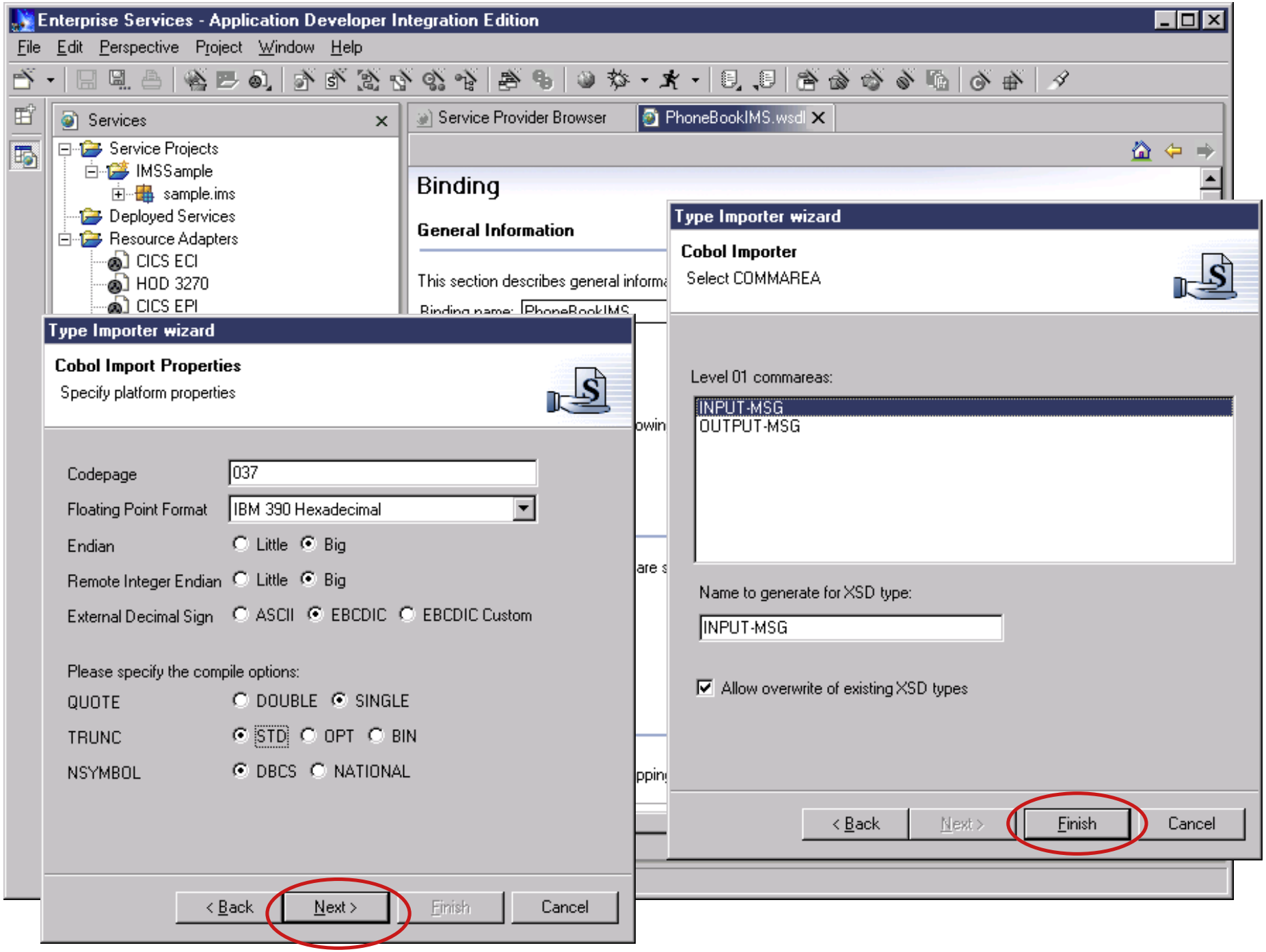

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Services Packages
Service Provider Browser
IMS Service
Connection Properties
Please specify the Port connection pro
Property
dataStoreName
groupName
hostName
overwriteLookup
password
portNumber
required
traceLevel
userName
Add Service...
Add EIS Service
Add Service
Specify a folder and the names of the new WSDL files.
Target namespace: http://ims.sample
PortType name: PhoneBook
Service name: PhoneBookIMSService
Specify the files to import into:
Destination folder: /IMSSample/sample/ims
Browse...
Interface WSDL file: PhoneBook.wsdl
Service WSDL file: PhoneBookIMS.wsdl
Generate inline XSD schemas
< Back
Next >
Finish
Cancel

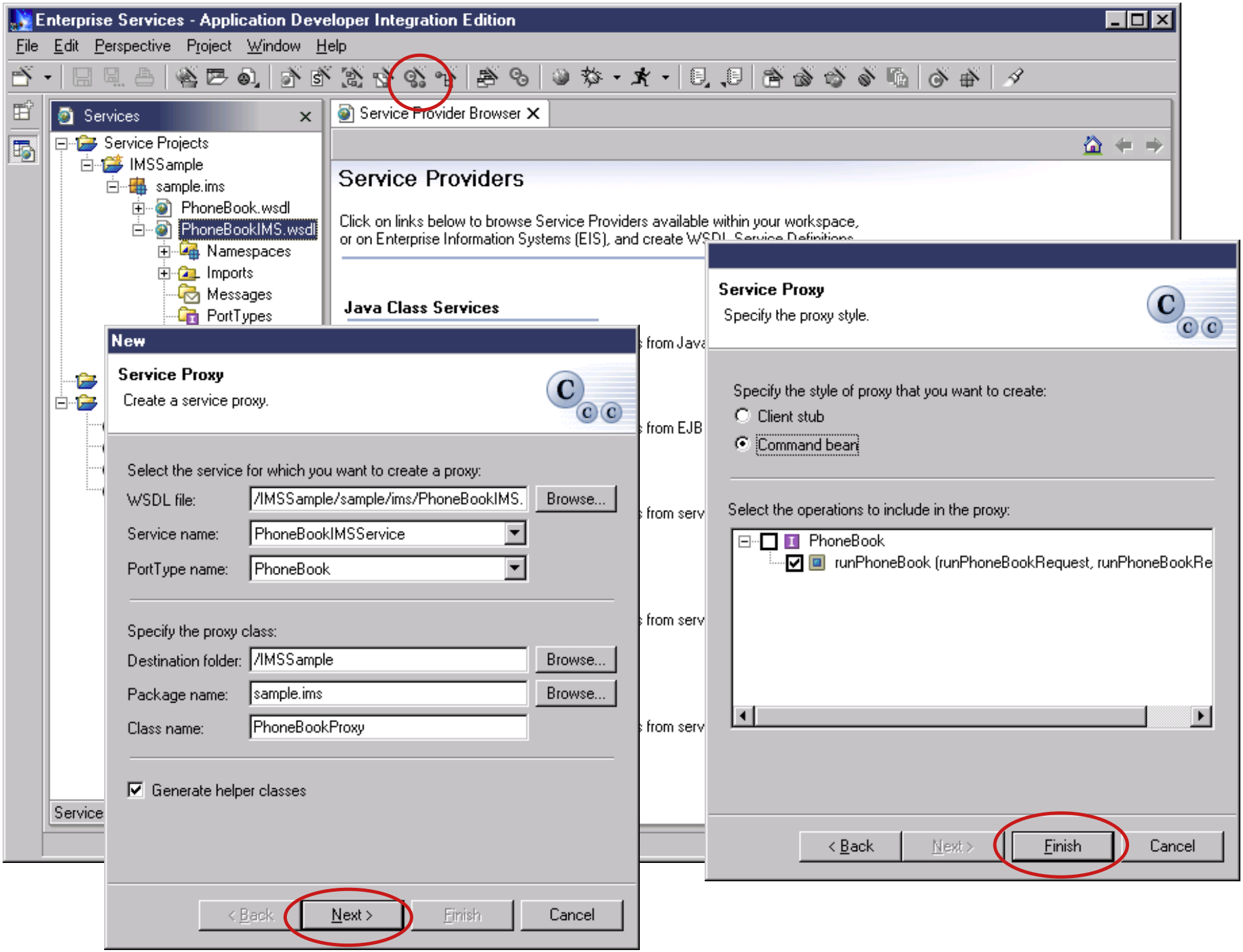

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Services Packages
Service Provider Browser
PhoneBookIMS.wsdl
Binding
General Information
This section describes general information about this operation
Binding name: PhoneBookIMS
Extensibility Elements
This binding is defined for the following port type:
PhoneBook
Browse...
Operations
The following operation bindings are supported:
New...
Type Mapping
This binding contains no type mapping.
Create...
Design Source

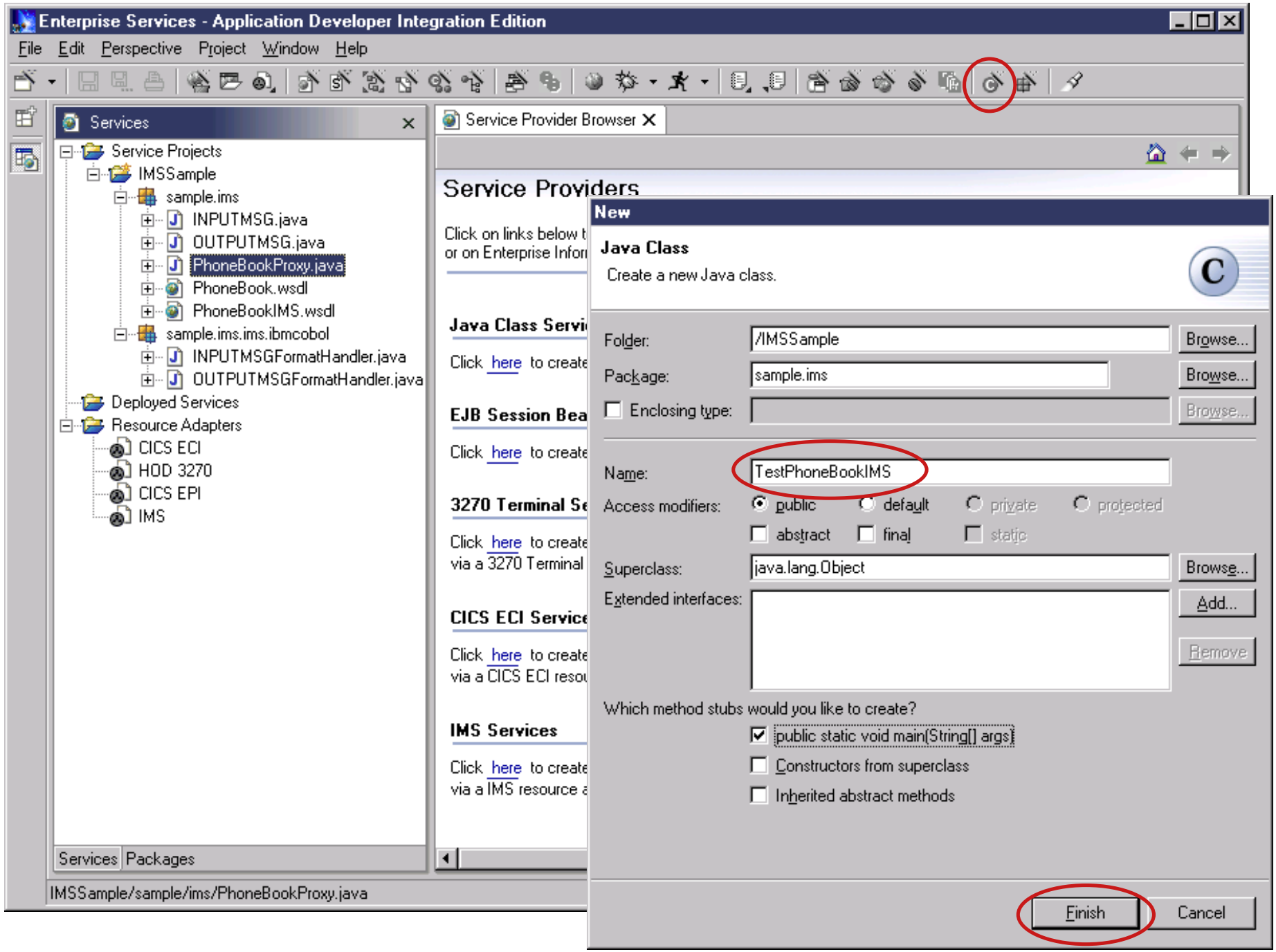

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Service Provider Browser
PhoneBookIMS.wsdl
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="PhoneBookIMSService"
targetNamespace="http://ims.sample"
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:ims="http://schemas.xmlsoap.org/wsdl/ims/" xmlns:tns="http://in
<import location="PhoneBook.wsdl" namespace="http://ims.sample"/>
<binding name="PhoneBookIMS" type="tns:PhoneBook">
<ims:binding/>
</binding>
<service name="PhoneBookIMSService">
<port binding="tns:PhoneBookIMS" name="imsPort">
<ims:address dataStoreName="SOCKEYE"
hostName="CSDMEC13.STL.IBM.COM" portNumber="9999"/>
</port>
</service>
</definitions>
Services Packages
Design Source

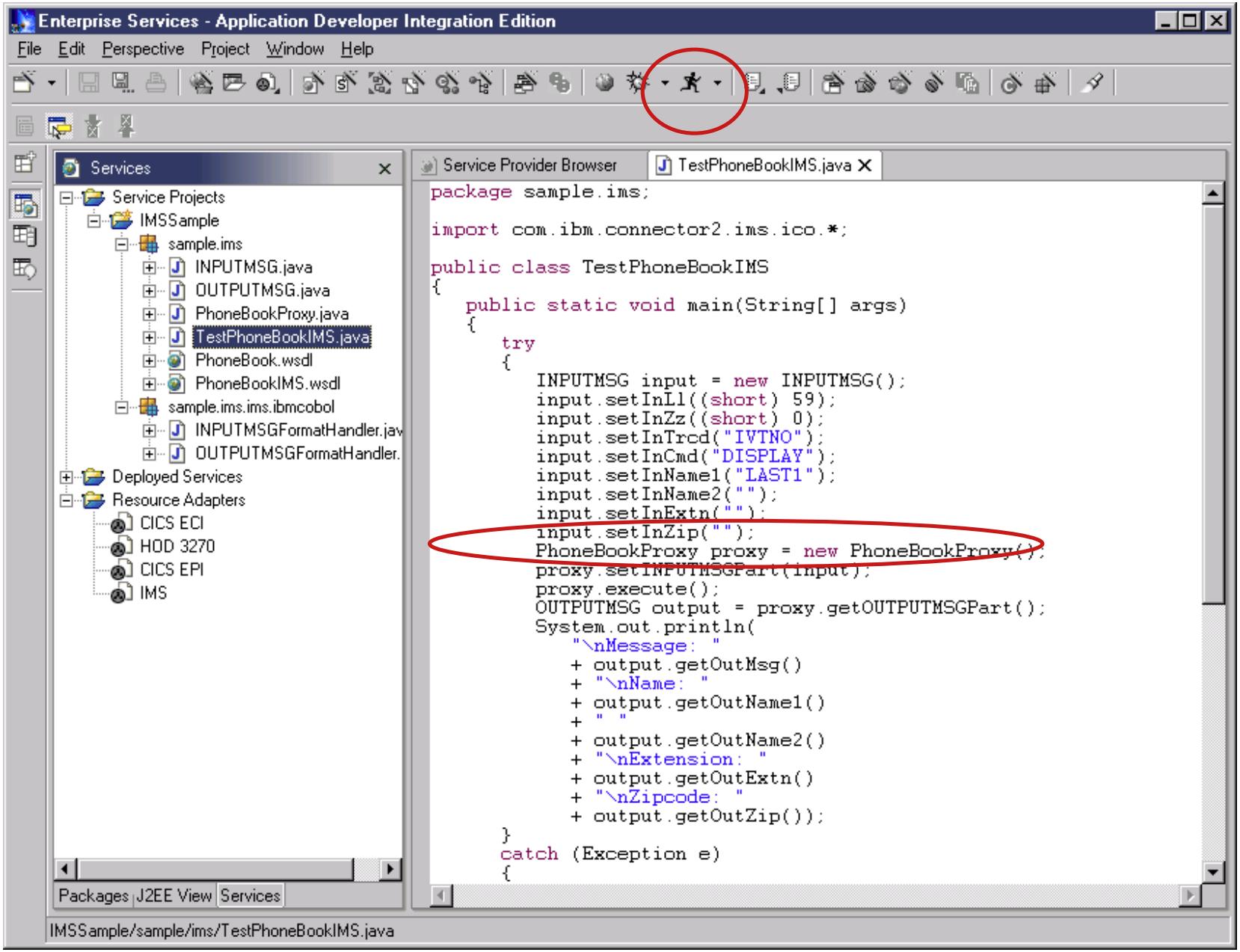

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Service Provider Browser
PhoneBookIMS.wsdl
Binding
General Information
This section describes general information about this operation
Binding name: PhoneBookIMS
Extensibility Elements
New Operation Binding
IMS Operation Binding Properties
Specify the properties you wish to use
Property
Value
convEnded
false
imsRequestType
1
interactionVerb
1
LTermName
mapName
required
< Back
Next >
Finish
Cancel
New Operation Binding
Operation Binding
Specify the name of the operation binding to create
Use an existing operation
Operation name:
Create a new operation
A new operation will be created on the referenced port type.
Operation name: runPhoneBook
Operation type: REQUEST_RESPONSE
< Back
Next >
Finish
Cancel

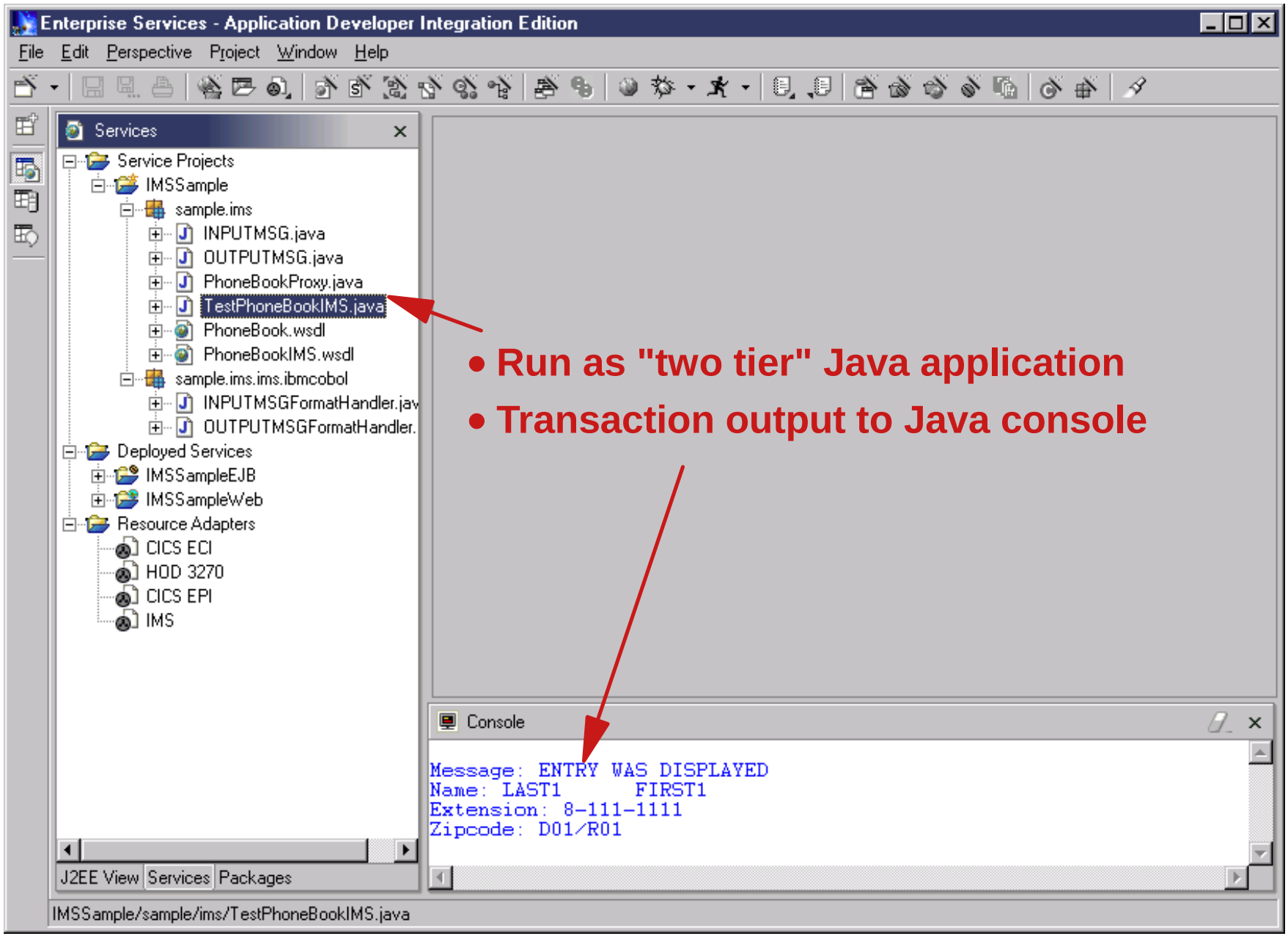

Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Service Provider Browser
PhoneBookIMS.wsdl
Binding
Type Importer wizard
Browse...
Create...
< Back
Next >
Finish
Cancel
New Operation Binding
Operation Binding
Specify the type and messages of the new operation
Input message:
Browse...
Import...
Use input message for output
Import or use an existing message
Create a message that represents multiple possible outputs
Output message:
Browse...
Import...
Create...
< Back
Next >
Finish
Cancel
File Selection
Select a File
IMSSample
.classpath
sample
ims
Ex01.ccp
PhoneBook.wsdl
PhoneBookIMS.wsdl
Installed Resource Adapters
myServer
OK
Cancel


Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
Service Provider Browser
PhoneBookIMS.wsdl
Binding
General Information
This section describes general informa
Type Importer wizard
Cobol Import Properties
Specify platform properties
Codepage
037
Floating Point Format
IBM 390 Hexadecimal
Endian
Little
Big
Remote Integer Endian
Little
Big
External Decimal Sign
ASCII
EBCDIC
EBCDIC Custom
Please specify the compile options:
QUOTE
DOUBLE
SINGLE
TRUNC
STD
OPT
BIN
NSYMBOL
DBCS
NATIONAL
< Back
Next >
Finish
Cancel
Type Importer wizard
Cobol Importer
Select COMMAREA
Level 01 commareas:
INPUT-MSG
OUTPUT-MSG
Name to generate for XSD type:
INPUT-MSG
Allow overwrite of existing XSD types
< Back
Next >
Finish
Cancel


Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
PhoneBook.wsdl
PhoneBookIMS.wsdl
Namespaces
Imports
Messages
PortTypes
Service Provider Browser
Service Providers
Click on links below to browse Service Providers available within your workspace,
Java Class Services
New
Service Proxy
Create a service proxy.
Select the service for which you want to create a proxy:
WSDL file: /IMSSample/sample/ims/PhoneBookIMS.
Browse...
Service name: PhoneBookIMSService
PortType name: PhoneBook
Specify the proxy class:
Destination folder: /IMSSample
Package name: sample.ims
Class name: PhoneBookProxy
Generate helper classes
< Back
Next >
Finish
Cancel
Service Proxy
Specify the proxy style.
Specify the style of proxy that you want to create:
Client stub
Command bean
Select the operations to include in the proxy:
PhoneBook
runPhoneBook (runPhoneBookRequest, runPhoneBookRe
< Back
Next >
Finish
Cancel


Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
INPUTMSG.java
OUTPUTMSG.java
PhoneBookProxy.java
PhoneBook.wsdl
PhoneBookIMS.wsdl
sample.ims.ims.ibmcobol
INPUTMSGFormatHandler.java
OUTPUTMSGFormatHandler.java
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Services Packages
IMSSample/sample/ims/PhoneBookProxy.java
Service Provider Browser
Service Providers
Java Class Servi
Click here to create
EJB Session Bea
Click here to create
3270 Terminal Se
Click here to create
via a 3270 Terminal
CICS ECI Service
Click here to create
via a CICS ECI resou
IMS Services
Click here to create
via a IMS resource a
New
Java Class
Create a new Java class.
Folder: /IMSSample
Browse...
Package: sample.ims
Browse...
Enclosing type:
Browse...
Name: TestPhoneBookIMS
Access modifiers: public default private protected
abstract final static
Superclass: java.lang.Object
Browse...
Extended interfaces:
Add...
Remove
Which method stubs would you like to create?
public static void main(String[] args)
Constructors from superclass
Inherited abstract methods
Finish
Cancel


Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
INPUTMSG.java
OUTPUTMSG.java
PhoneBookProxy.java
TestPhoneBookIMS.java
PhoneBook.wsdl
PhoneBookIMS.wsdl
sample.ims.ims.ibmcobol
INPUTMSGFormatHandler.jav
OUTPUTMSGFormatHandler.
Deployed Services
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
Packages J2EE View Services
Service Provider Browser
TestPhoneBookIMS.java
package sample.ims;
import com.ibm.connector2.ims.ico.*;
public class TestPhoneBookIMS
{
public static void main(String[] args)
{
try
{
INPUTMSG input = new INPUTMSG();
input.setInLl((short) 59);
input.setInZz((short) 0);
input.setInTrcd("IVTNO");
input.setInCmd("DISPLAY");
input.setInName1("LAST1");
input.setInName2("");
input.setInExtn("");
input.setInZip("");
PhoneBookProxy proxy = new PhoneBookProxy();
proxy.setINPUTMSGPart(input);
proxy.execute();
OUTPUTMSG output = proxy.getOUTPUTMSGPart();
System.out.println(
"\nMessage: "
+ output.getOutMsg()
+ "\nName: "
+ output.getOutName1()
+ " "
+ output.getOutName2()
+ "\nExtension: "
+ output.getOutExtn()
+ "\nZipcode: "
+ output.getOutZip());
}
catch (Exception e)
{
IMSSample/sample/ims/TestPhoneBookIMS.java


Enterprise Services - Application Developer Integration Edition
File Edit Perspective Project Window Help
Services
Service Projects
IMSSample
sample.ims
INPUTMSG.java
OUTPUTMSG.java
PhoneBookProxy.java
TestPhoneBookIMS.java
PhoneBook.wsdl
PhoneBookIMS.wsdl
sample.ims.ims.ibmcobol
INPUTMSGFormatHandler.jav
OUTPUTMSGFormatHandler.
Deployed Services
IMSSampleEJB
IMSSampleWeb
Resource Adapters
CICS ECI
HOD 3270
CICS EPI
IMS
J2EE View Services Packages
• Run as "two tier" Java application
• Transaction output to Java console
Console
Message: ENTRY WAS DISPLAYED
Name: LAST1 FIRST1
Extension: 8-111-1111
Zipcode: D01/R01
IMSSample/sample/ims/TestPhoneBookIMS.java

How to find our updates

- IMS Connector for Java **development**
  - Latest versions included in WSAD IE 4.1
- IMS Connector for Java **runtime** (aka IMS Resource Adapter, aka WebSphere Adapter for IMS):
  - For distributed platforms:
    - 1.2.0/1.2.1 executables on CD with IMS Connect 1.2
      - Linux for z/OS executable downloadable from IMS Web site
    - Later versions downloadable from IMS Web site
      - www.ibm.com/ims
  - For z/OS platform:
    - SMPE installable APARs

Appendix: Current IMS Connector Support

Function	z/390 client	distributed client
Send Only	No	No
Send Receive	Yes	Yes
Resume TPipe (Asynchronous output)	No	No
Commit Mode 0	No	No
Commit Mode 1	Yes	Yes
Sync Level NONE	Yes	Yes
Sync Level CONFIRM	Yes (CCF only)	Yes (CCF only)
Sync Level SYNC (2-phase commit)	Yes (JCA with Local Option only)	No
Non-conversational transactions	Yes	Yes
Conversational transactions	Yes	Yes
Access IMS Full Function DB	Yes	Yes
Access IMS Fast Path DB	Yes	Yes
MulitSegment Input and Output Messages	Yes	Yes
Local Option (Program call)	Yes	N/A
TCP/IP	Yes	Yes
JCA Transaction support	Yes (Currently support global transaction with RRS, non-XA)	No
JCA Component-Managed Security	Yes	Yes
JCA Container-Managed Security	Yes	No
JCA CCI Interface	Yes	Yes
SSL	No	No
Execution Timeout	No	No
Cascaded Transaction Support	No	No

Appendix: Future IMS Connector Support

Function	z/390 client	distributed client
Send Only	Yes	Yes
Send Receive	Yes	Yes
Resume TPipe (Asynchronous output)	Yes	Yes
Commit Mode 0	Yes	Yes
Commit Mode 1	Yes	Yes
Sync Level NONE	Yes	Yes
Sync Level CONFIRM	Yes (CCF only)	Yes (CCF only)
Sync Level SYNC (2-phase commit)	Yes (JCA with Local Option only)	Yes
Non-conversational transactions	Yes	Yes
Conversational transactions	Yes	Yes
Access IMS Full Function DB	Yes	Yes
Access IMS Fast Path DB	Yes	Yes
MulitSegment Input and Output Messages	Yes	Yes
Local Option (Program call)	Yes	N/A
TCP/IP	Yes	Yes
JCA Transaction support	Yes	Yes
JCA Component-Managed Security	Yes	Yes
JCA Container-Managed Security	Yes	Yes
JCA CCI Interface	Yes	Yes
SSL	Yes	Yes
Execution Timeout	Yes	Yes
Cascaded Transaction Support	Yes	Yes

Related Web Sites

e-business powered by IMS

- **IMS, IMS Connect, IMS Connector for Java**
  - http://www.ibm.com/ims
- **WebSphere Application Server**
  - http://www.ibm.com/software/webservers
- **WebSphere Studio Application Developer, Integration Edition**
  - http://www.ibm.com/software/ad/studioappdev/

© IBM Corporation 2002

RETURN TO INDEX