



# Building Web Service Applications



John Apps and Mick  
Keyes  
Technical Architects  
Hewlett-Packard

Jonathan Halliday  
Solution Architect  
Arjuna Ltd.



# Agenda

- HP BridgeWorks
- HP BridgeWorks and BEA WebLogic Workshop
- Web Services
- WebLogic Platform and HP Integration Technologies
- Summary



# HP BridgeWorks

Generates robust EJB  
Interfaces from your TP or  
3GL applications

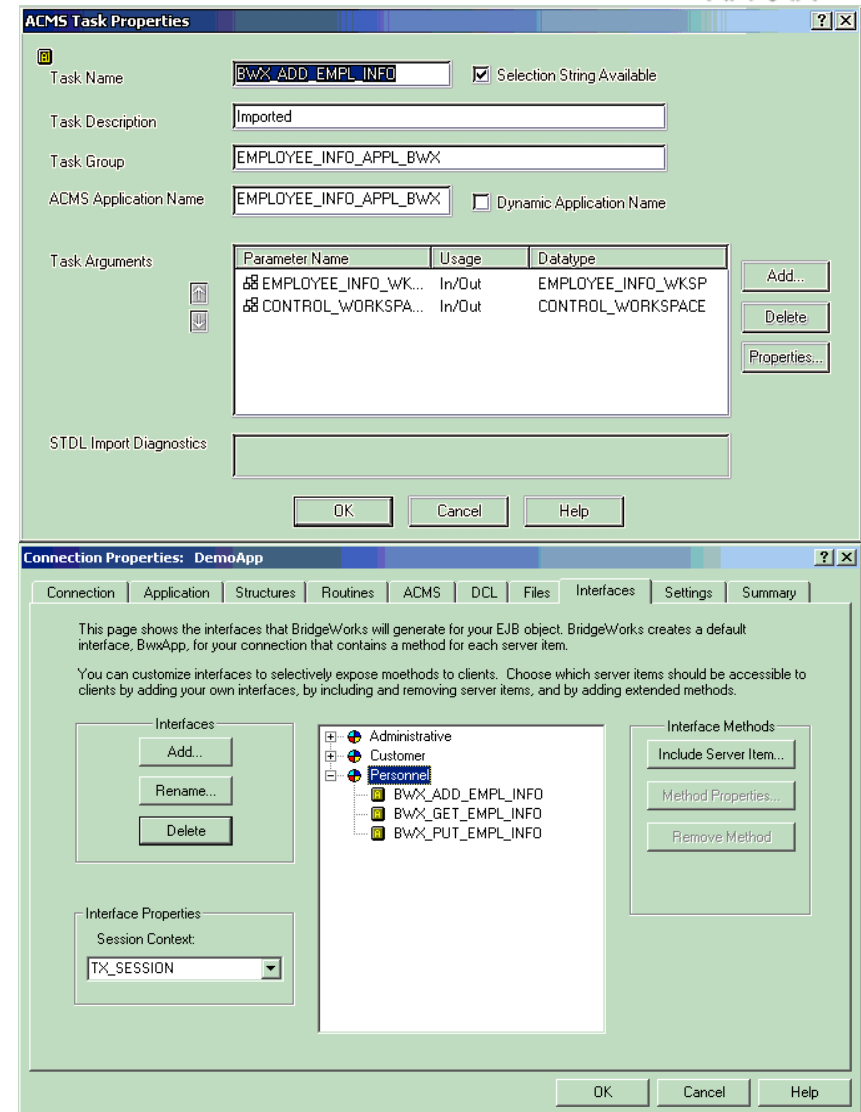
Creates J2EE Stateful Session  
Beans

Creates 1 EJB per interface

Generates JavaBean  
Interfaces from your TP or  
3GL application

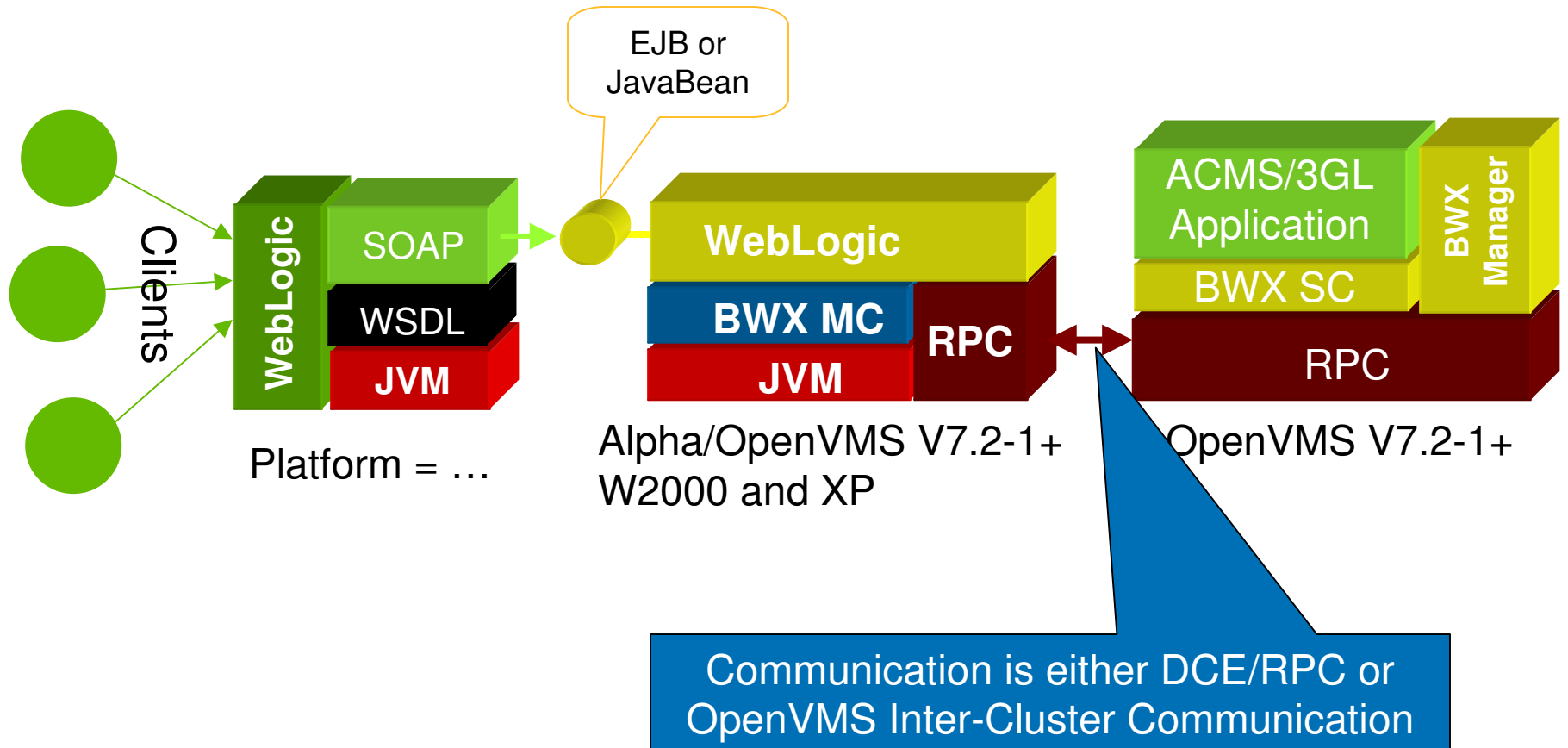
Creates a single simple class per  
interface

J2EE environment not needed to  
build or run



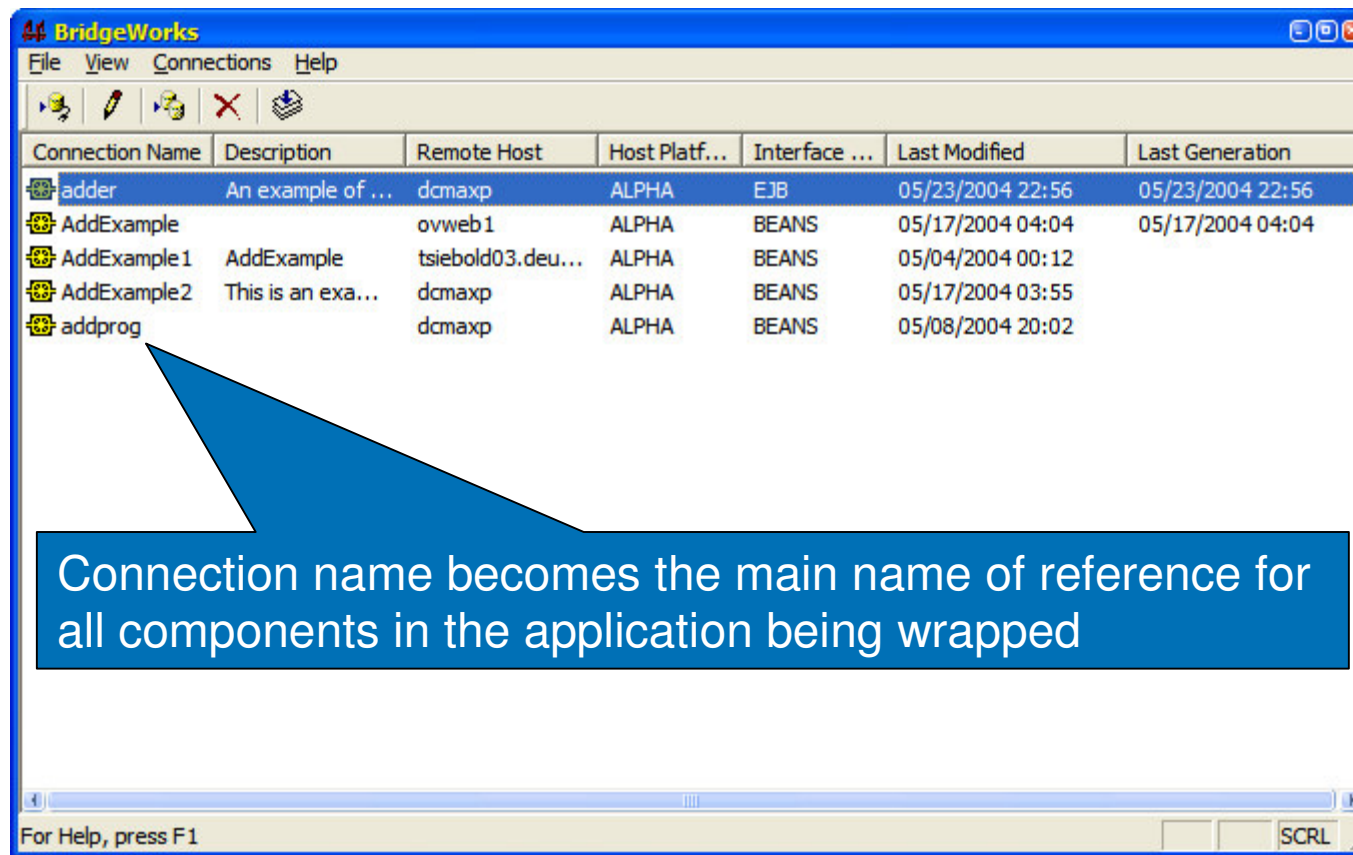
# HP BridgeWorks (BWX) Components

Components in **green** you supply, **BridgeWorks** provides/generates the rest



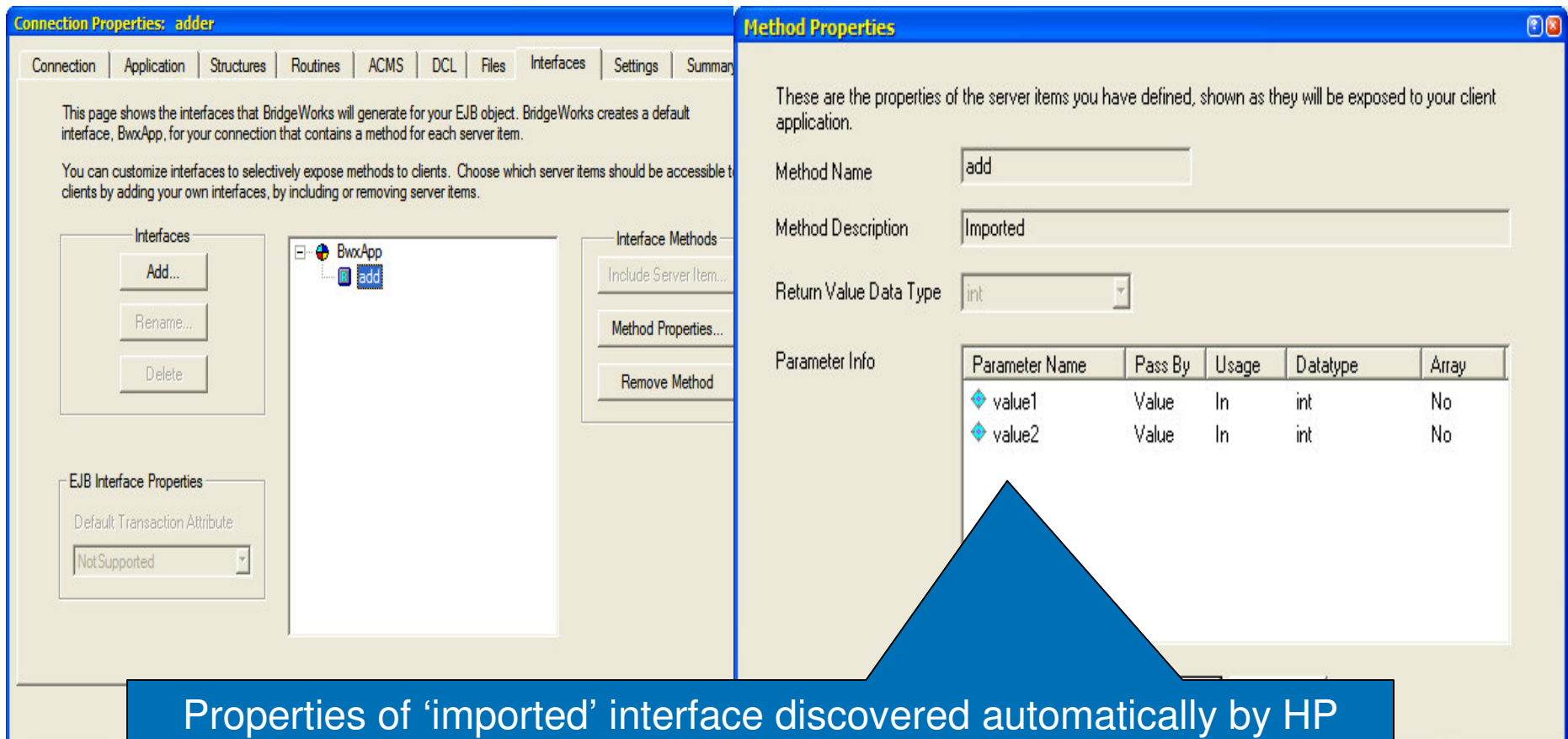
# HP BridgeWorks

- Main dialog screen showing applications which have been wrapped and which can now be modified



# Hp BridgeWorks

## Easy Point & Click building of Components/Interfaces



The screenshot displays two windows from the HP BridgeWorks application. The 'Connection Properties' window on the left shows the 'Interfaces' tab, where a default interface 'BwxApp' is listed with a method 'add'. The 'Method Properties' window on the right shows the details for the 'add' method, including its name, description, return type, and parameter information.

**Connection Properties: adder**

Connection | Application | Structures | Routines | ACMS | DCL | Files | Interfaces | Settings | Summary

This page shows the interfaces that BridgeWorks will generate for your EJB object. BridgeWorks creates a default interface, BwxApp, for your connection that contains a method for each server item.

You can customize interfaces to selectively expose methods to clients. Choose which server items should be accessible to clients by adding your own interfaces, by including or removing server items.

**Interfaces**

Add...  
Rename...  
Delete

**EJB Interface Properties**

Default Transaction Attribute  
NotSupported

**Interface Methods**

Include Server Item...  
Method Properties...  
Remove Method

**Method Properties**

These are the properties of the server items you have defined, shown as they will be exposed to your client application.

Method Name: add

Method Description: Imported

Return Value Data Type: int

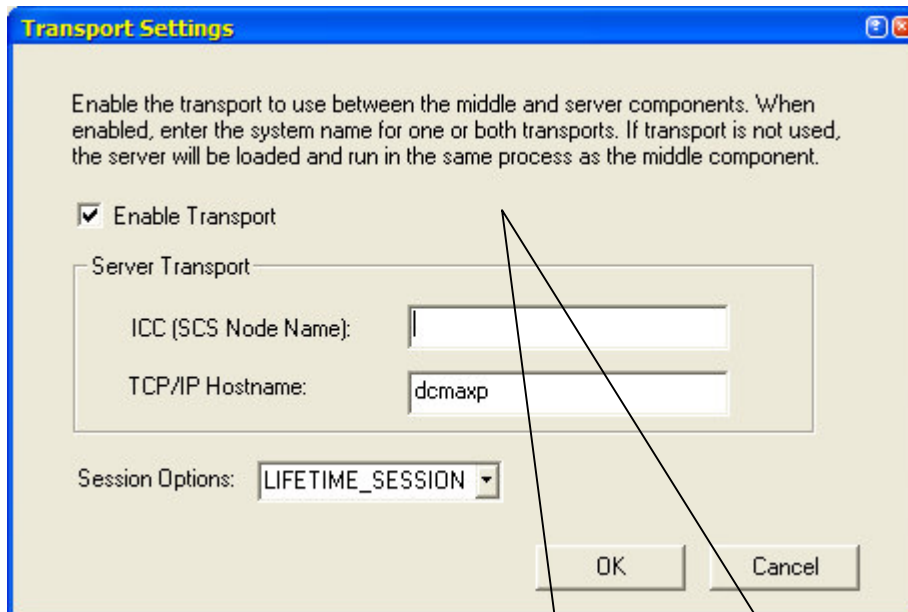
**Parameter Info**

Parameter Name	Pass By	Usage	Datatype	Array
value1	Value	In	int	No
value2	Value	In	int	No

Properties of 'imported' interface discovered automatically by HP BridgeWorks from compiler-generated file

# Hp BridgeWorks

Easy modification of previously generated 'connections'



**Transport Settings**

Enable the transport to use between the middle and server components. When enabled, enter the system name for one or both transports. If transport is not used, the server will be loaded and run in the same process as the middle component.

☒ Enable Transport

Server Transport:

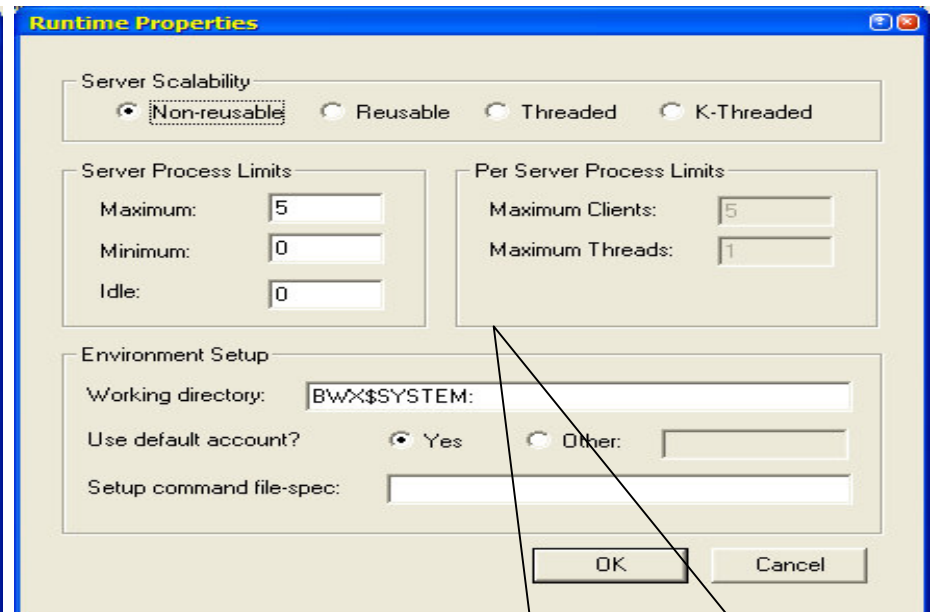
ICC (SCS Node Name):

TCP/IP Hostname:

Session Options:

OK Cancel

Transport to be used and node on which application resides



**Runtime Properties**

Server Scalability:

☒ Non-reusable ☐ Reusable ☐ Threaded ☐ K-Threaded

Server Process Limits:

Maximum:   
Minimum:   
Idle:

Per Server Process Limits:

Maximum Clients:   
Maximum Threads:

Environment Setup:

Working directory:   
Use default account? ☒ Yes ☐ Other:   
Setup command file-spec:

OK Cancel

Runtime attributes of wrapped application

# HP BridgeWorks with 3GL

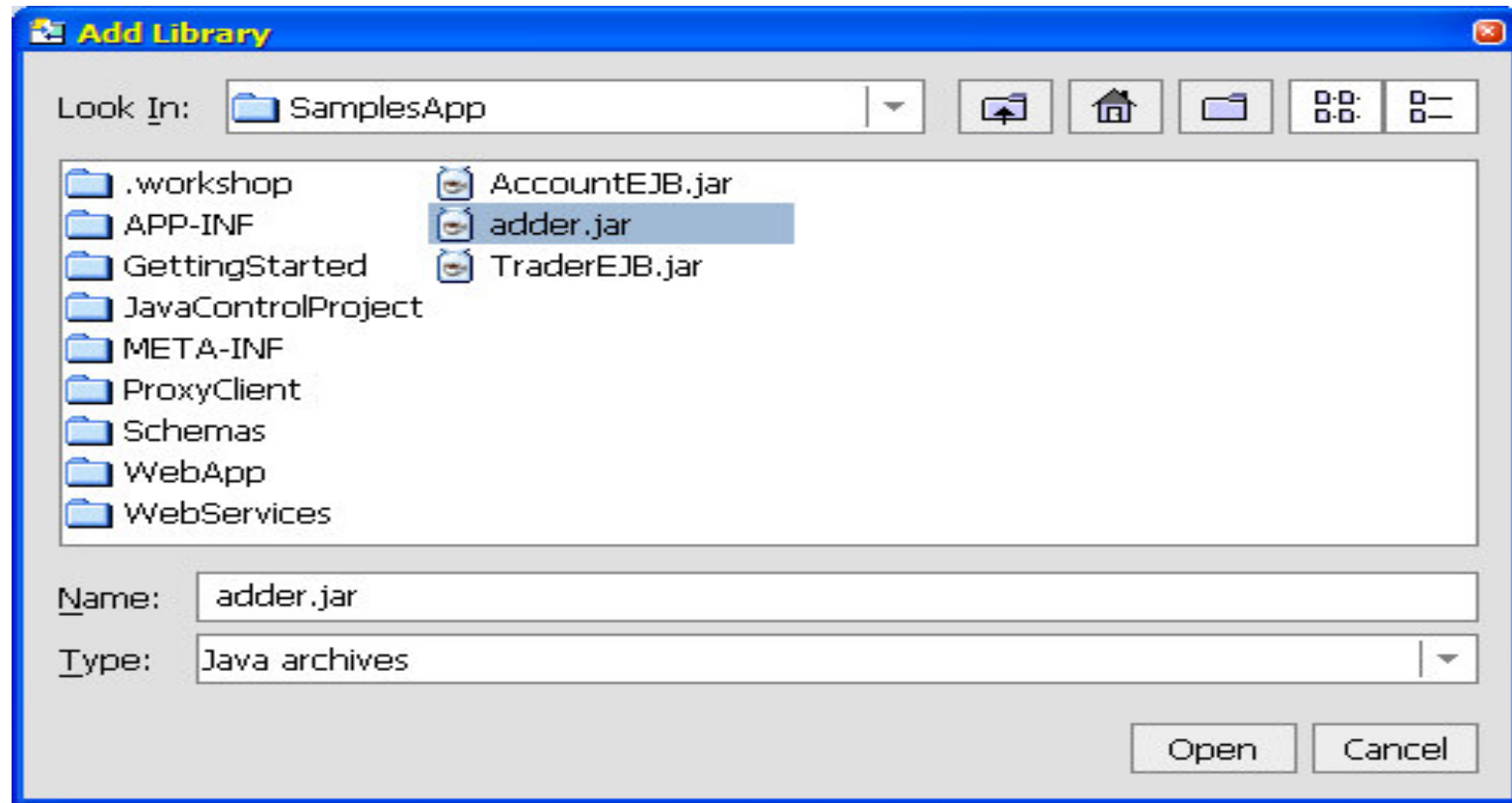
```
// add.c
// Simple program to add two
// numbers and return the sum.
// The interface is generated
// automatically by HP BridgeWorks
// and invoked as a method in an EJB
// using BEA WebLogic Server.

long add( long a, long b ){
    return( a + b );
}
```



# BridgeWorks and WebLogic Workshop

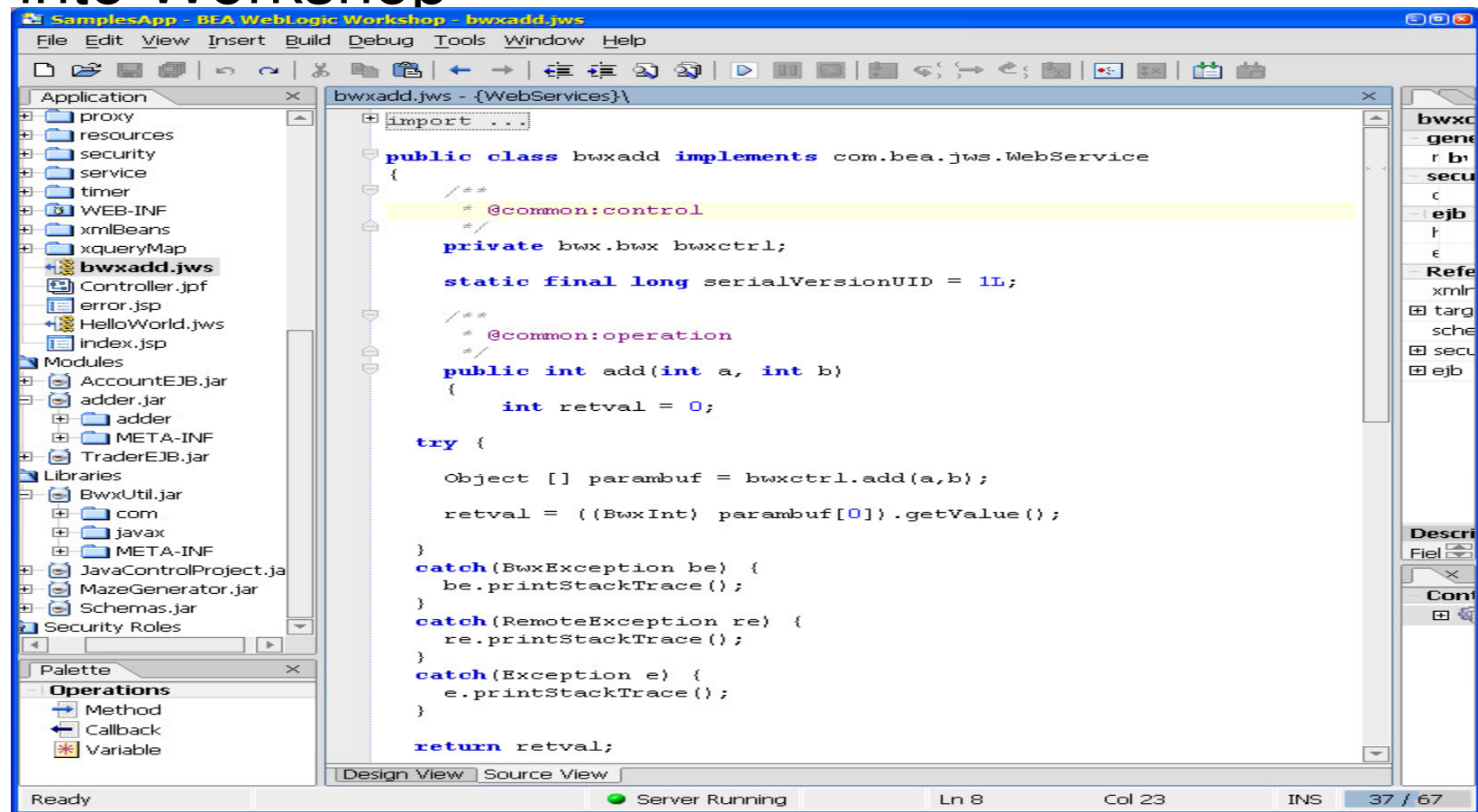
- Make the BridgeWorks JAR file with the generated EJB code known to Workshop



- 
- The screenshot displays the BEA WebLogic Workshop environment. The title bar reads "SamplesApp - BEA WebLogic Workshop - bwxadd.jws". The menu bar includes File, Edit, View, Insert, Build, Debug, Tools, Window, and Help. The toolbar contains various icons for file operations and development tools. The left-hand pane is divided into two sections: "Application" and "Palette". The "Application" section shows a project tree with folders like "service", "timer", "WEB-INF", "xmlBeans", "xqueryMap", and a file named "bwxadd.jws". The "Palette" section shows a list of "Operations". The main design view shows a "bwxadd Web Service" component. A "bwxctrl" component is connected to the "add" operation of the "bwxadd Web Service" via five arrows labeled "add", "BwxGetFile", "BwxInvoke", "BwxInvokeDcl", and "create". The status bar at the bottom indicates "Ready", "Server Running", and "INS 35 / 67".

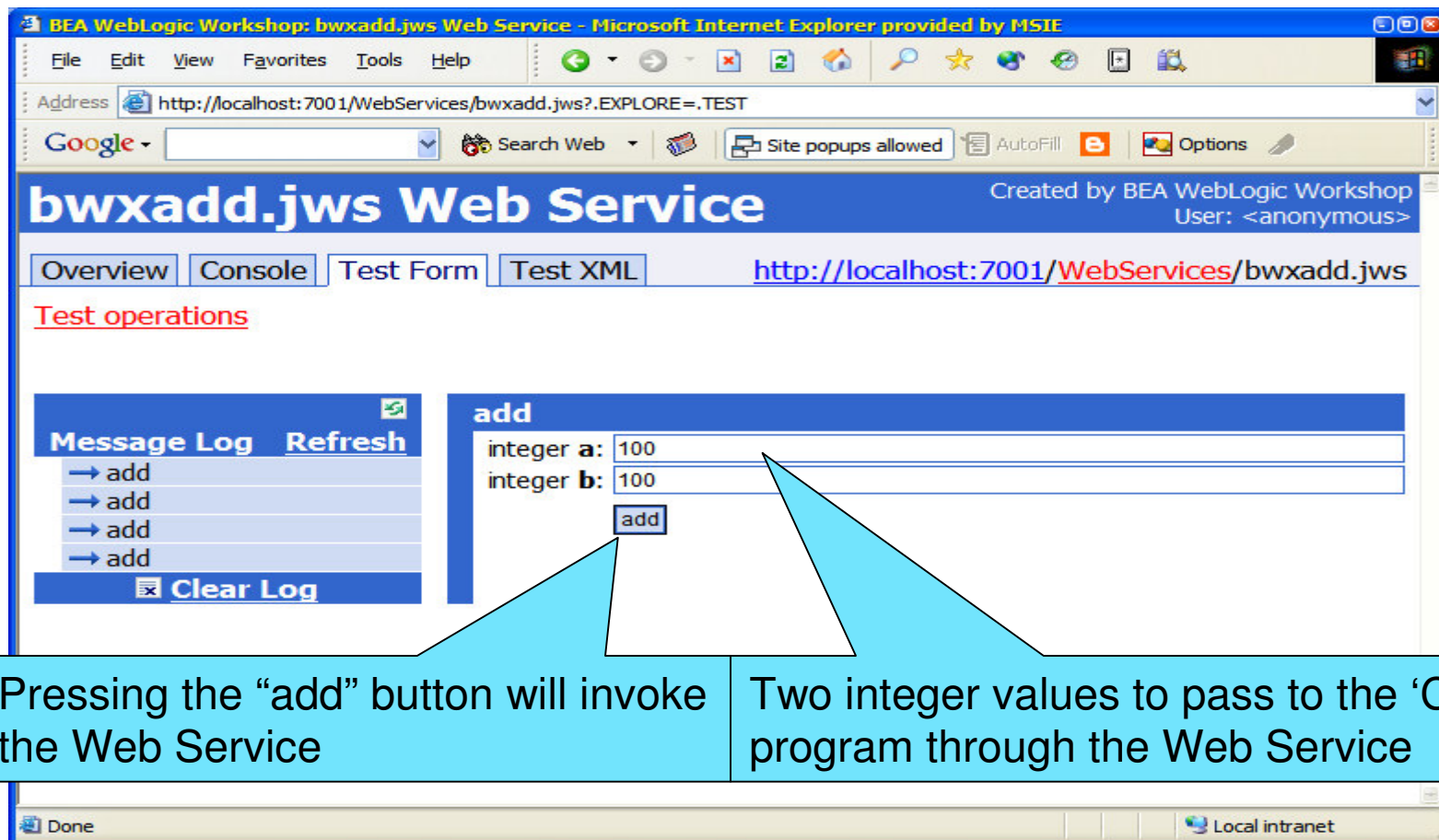
# BridgeWorks and WebLogic Workshop

- Insert the appropriate code for the Web Service into Workshop



# BridgeWorks and WebLogic Workshop

- Testing the Web Service in a browser. This will invoke the 'C' program on the remote system as if it were an Enterprise JavaBean



BEA WebLogic Workshop: bwxadd.jws Web Service - Microsoft Internet Explorer provided by MSIE

Address: <http://localhost:7001/WebServices/bwxadd.jws?EXPLORE=,TEST>

Google Search Web Site popups allowed AutoFill Options

**bwxadd.jws Web Service** Created by BEA WebLogic Workshop User: <anonymous>

Overview Console Test Form Test XML <http://localhost:7001/WebServices/bwxadd.jws>

Test operations

**Message Log** Refresh

- add
- add
- add
- add

Clear Log

**add**

integer a: 100

integer b: 100

add

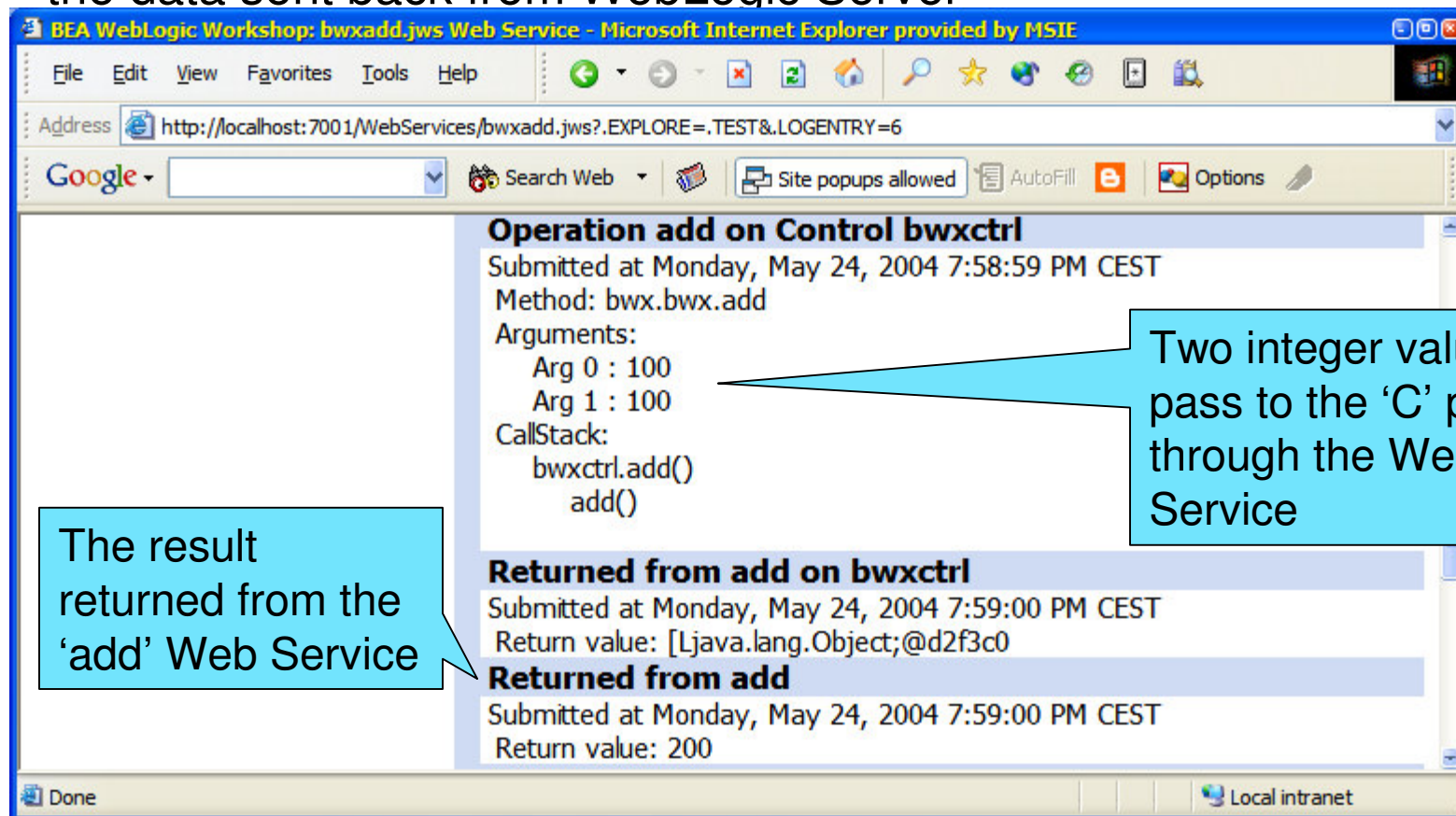
Pressing the "add" button will invoke the Web Service

Two integer values to pass to the 'C' program through the Web Service

Done Local intranet

# BridgeWorks and WebLogic Workshop

- Testing the Web Service in a browser. The results will be shown in the data sent back from WebLogic Server



**BEA WebLogic Workshop: bwxadd.jws Web Service - Microsoft Internet Explorer provided by MSIE**

Address <http://localhost:7001/WebServices/bwxadd.jws?.EXPLORE=,TEST&.LOGENTRY=6>

Google Search Web Site popups allowed AutoFill Options

**Operation add on Control bwxctrl**  
Submitted at Monday, May 24, 2004 7:58:59 PM CEST  
Method: bwx.bwx.add  
Arguments:  
Arg 0 : 100  
Arg 1 : 100  
CallStack:  
bwxctrl.add()  
add()

**Returned from add on bwxctrl**  
Submitted at Monday, May 24, 2004 7:59:00 PM CEST  
Return value: [Ljava.lang.Object;@d2f3c0

**Returned from add**  
Submitted at Monday, May 24, 2004 7:59:00 PM CEST  
Return value: 200

Done Local intranet

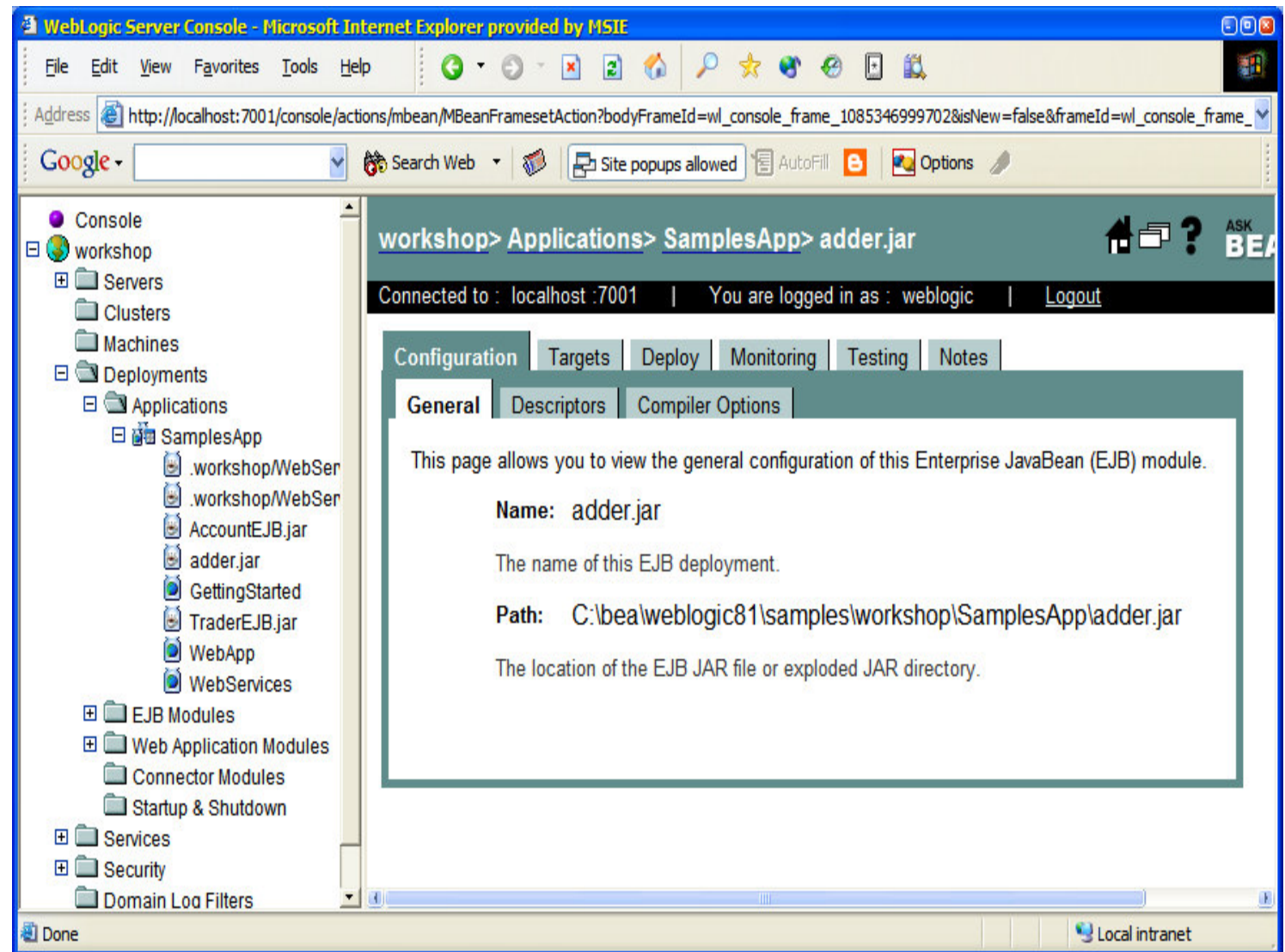
The result  
returned from the  
'add' Web Service

Two integer values to  
pass to the 'C' program  
through the Web  
Service



# BEA WebLogic Server

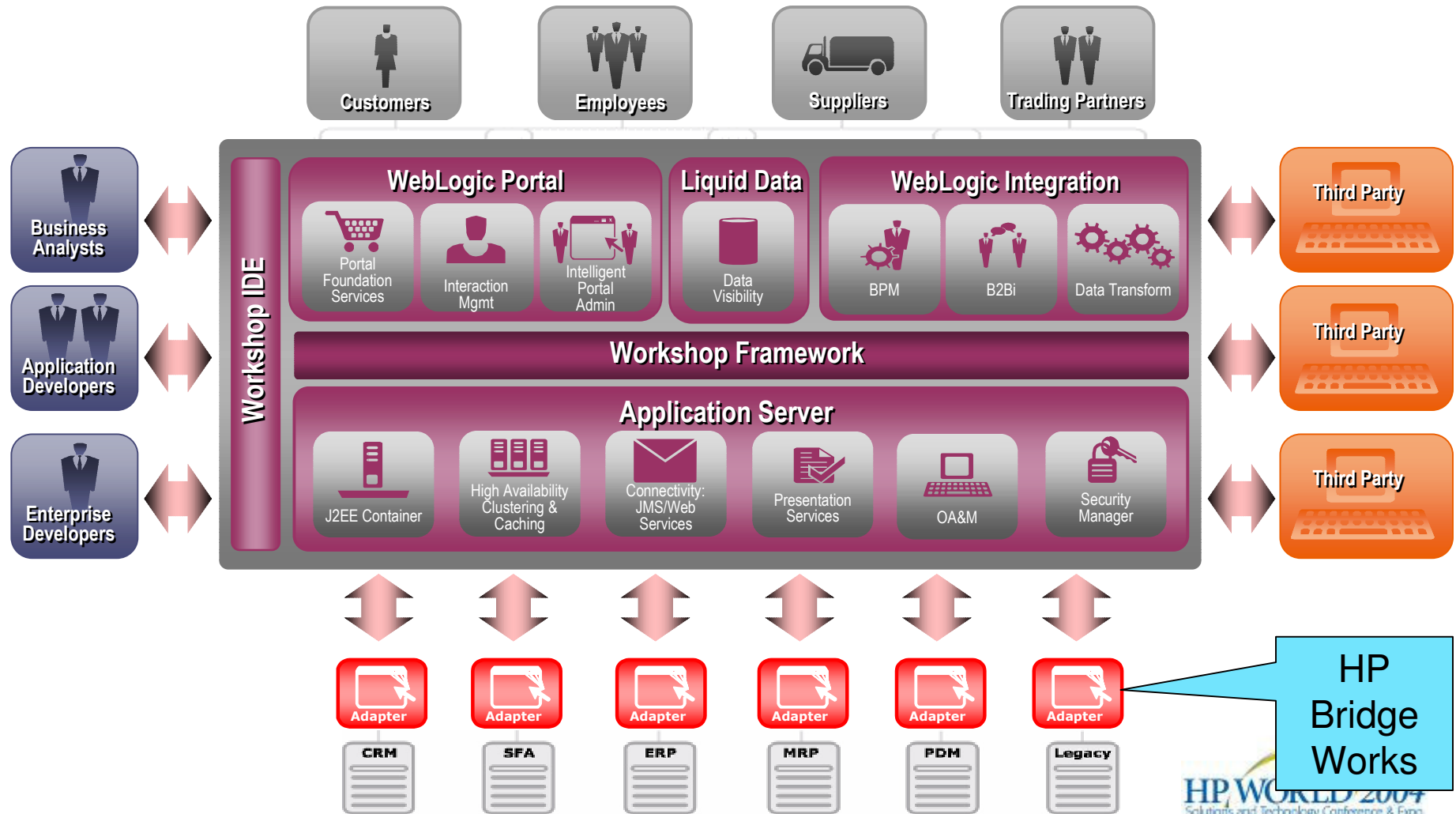
Used to  
invoke and  
control the  
EJB or  
JavaBean  
methods  
generated by  
BridgeWorks  
from ACMS  
and 3GL  
program  
meta-data



# Web Services

- Provide the 'wire protocol' to:
  - Enable the invocation of remote methods
  - Handle marshalling of arguments
  - Allow message or RPC based connections
- Enable the discovery of services through standards-based directories
- Hide the intricacies of the methods being invoked
- Allow invocation of methods regardless of language and platform
- Are prerequisite for Service-oriented Architectures

# WebLogic Platform and HP Integration Technologies





# Summary

- Quickly generate all necessary components with little to no manual intervention or changes
- Generate EJB or JavaBean components from same definitions with no changes
- Integrate bespoke transactions with new applications using Web Services quickly and easily
- Reuse well-tested, reliable and long-running transaction processing functions
- Increase return on IT investment

# HP WORLD 2004

Solutions and Technology Conference & Expo

Co-produced by:



RECOMMENDED TRAINING VENUE FOR THE  
**HP Certified Professional**

