Web Tools in WebSphere Development Studio Client 5.0

George Farr, Claus Weiss,
Phil Coulthard
IBM Toronto Laboratory

ITSO iSeries Technical Forum
SP12

WDS for iSeries

New World

New Servers

New Tools

Disclaimer

Acknowledgement:

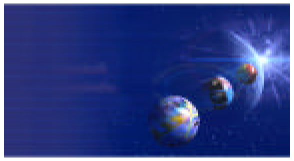
- This presentation is a collaborative effort of the IBM Toronto AS/400 Application Development presentation team, including work done by:
  - Phil Coulthard, George Farr, Claus Weiss, Don Yantzi

Disclaimer:

- The information contained in this document has not been submitted to any formal IBM test and is distributed on an as is basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customers' ability to evaluate and integrate them into the customers' operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will result elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

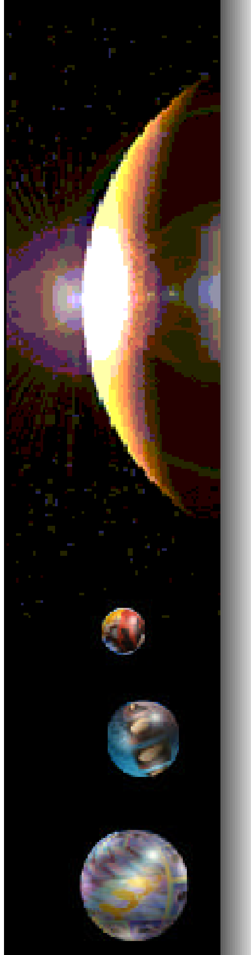
Reproduction:

- The base presentation is the property of IBM Corporation. Permission must be obtained PRIOR to making copies of this material for any reason.

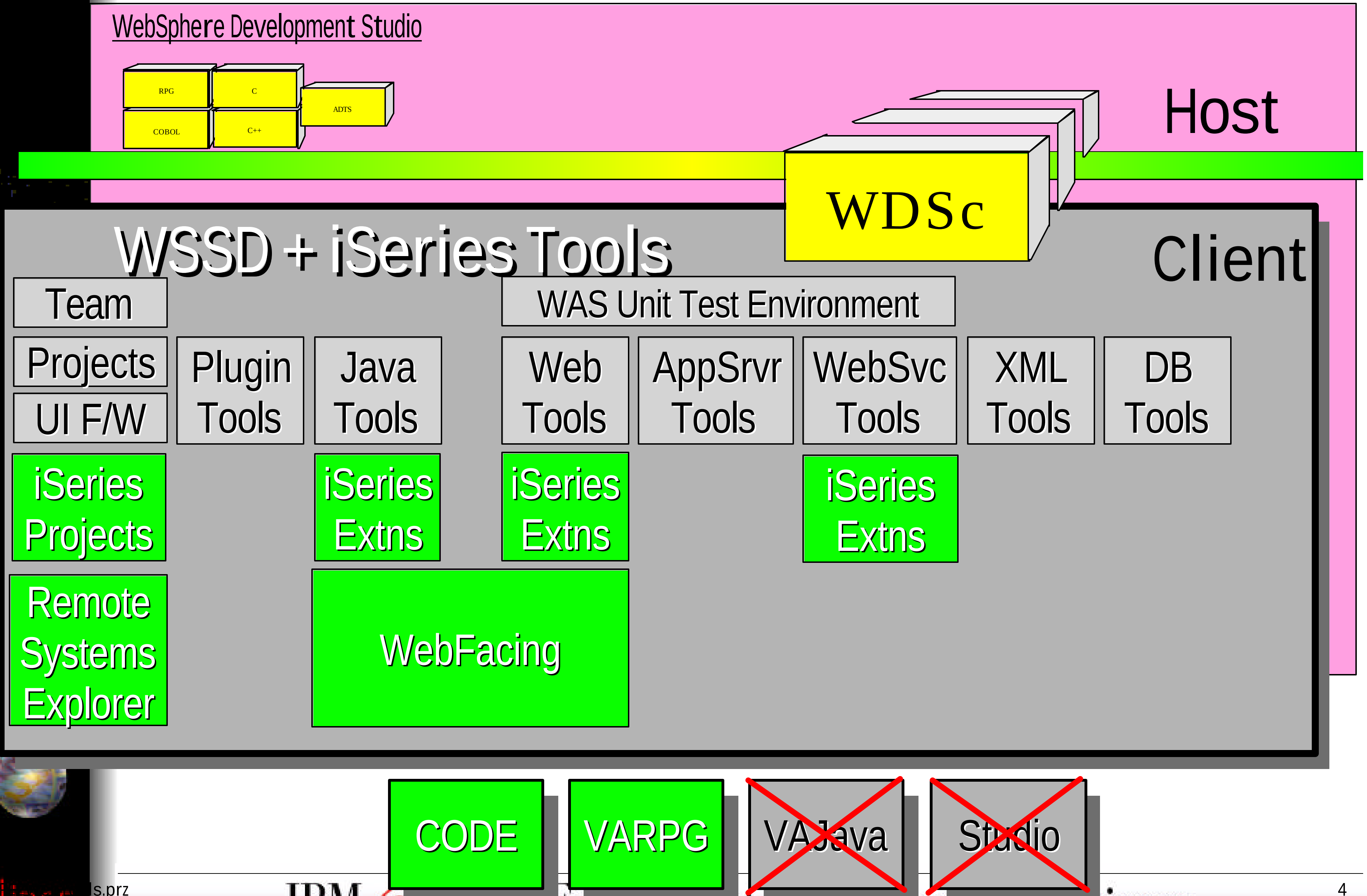


AGENDA

- What is in WDSc 5.0?
- e-business Primer
  - ► AD Model, traditional and web
- Web Application Primer
  - ► JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - ► RPG Example
  - ► iSeries Design time controls (DTCs)
  - ► Web Interaction
  - ► Publishing
- Conclusion



WDSc 5.0 Standard!



WebSphere Development Studio

RPG

C

COBOL

C++

ADTS

Host

WDSc

Client

WSSD + iSeries Tools

Team

Projects

UI F/W

Plugin Tools

Java Tools

WAS Unit Test Environment

Web Tools

AppSrvr Tools

WebSvc Tools

XML Tools

DB Tools

iSeries Projects

iSeries Extns

iSeries Extns

iSeries Extns

Remote Systems Explorer

WebFacing

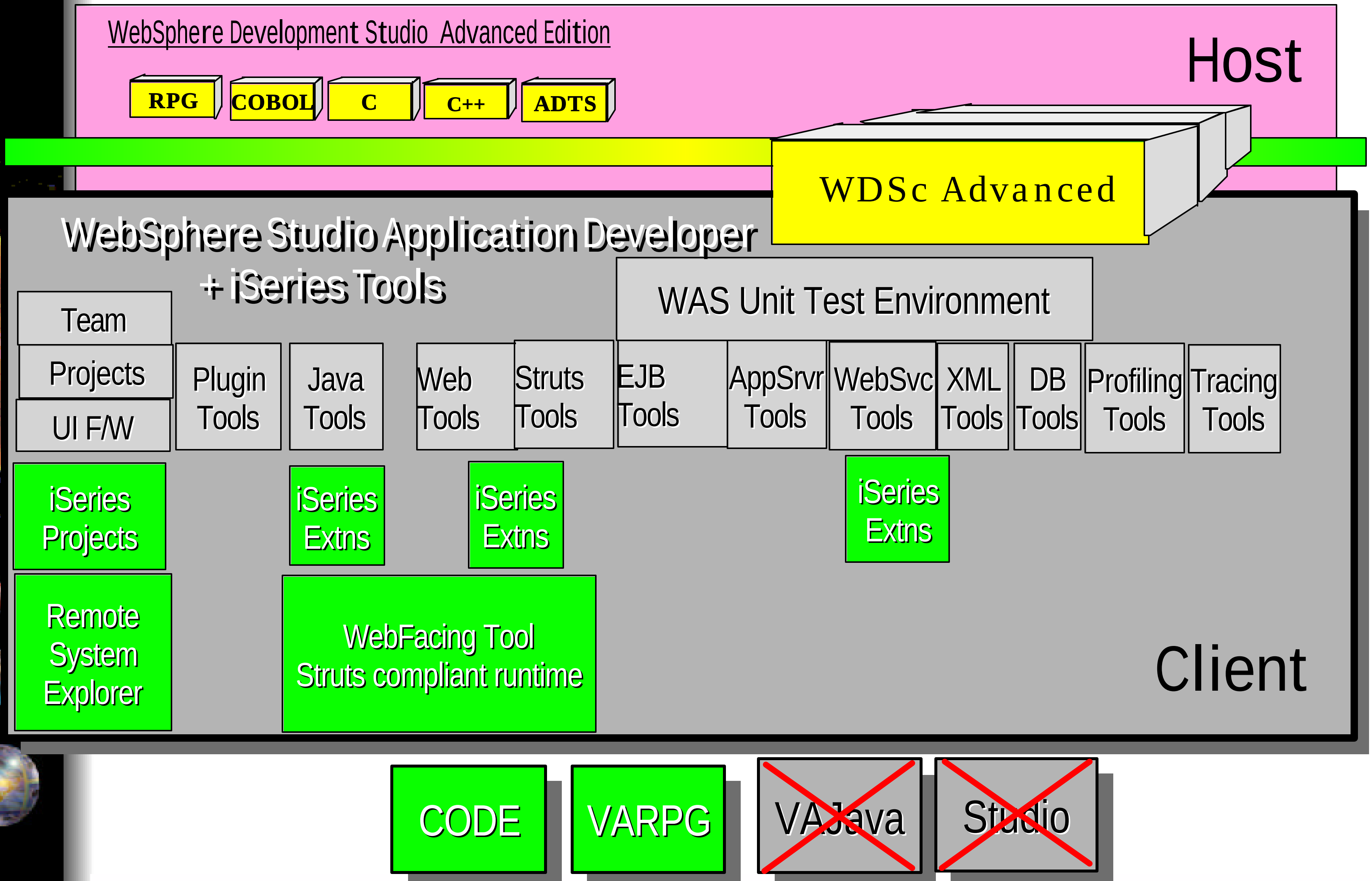
CODE

VARPG

~~VAJava~~

~~Studio~~

WDSc 5.0 Advanced!



WebSphere Development Studio Advanced Edition

Host

RPG COBOL C C++ ADTS

WDSc Advanced

WebSphere Studio Application Developer + iSeries Tools

WAS Unit Test Environment

Team

Projects

UI F/W

Plugin Tools

Java Tools

Web Tools

Struts Tools

EJB Tools

AppSrvr Tools

WebSvc Tools

XML Tools

DB Tools

Profiling Tools

Tracing Tools

iSeries Projects

iSeries Extns

iSeries Extns

iSeries Extns

Remote System Explorer

WebFacing Tool
Struts compliant runtime

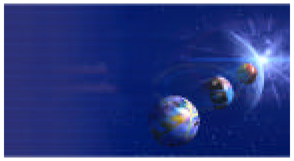
Client

CODE

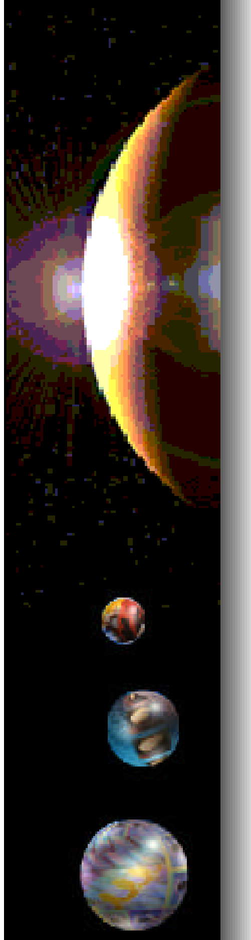
VARPG

~~VAJava~~

~~Studio~~

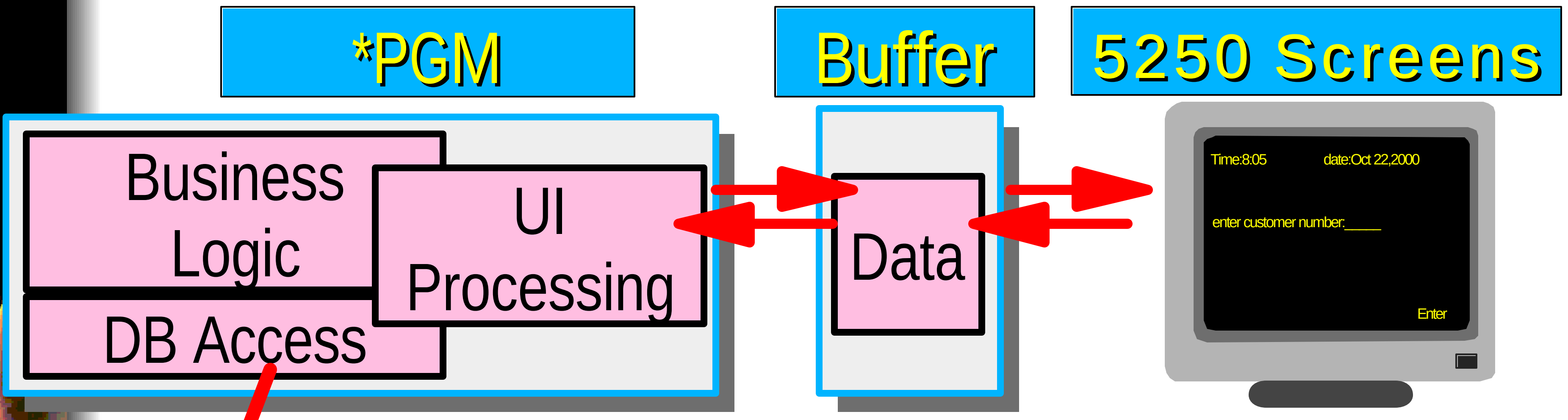


AGENDA



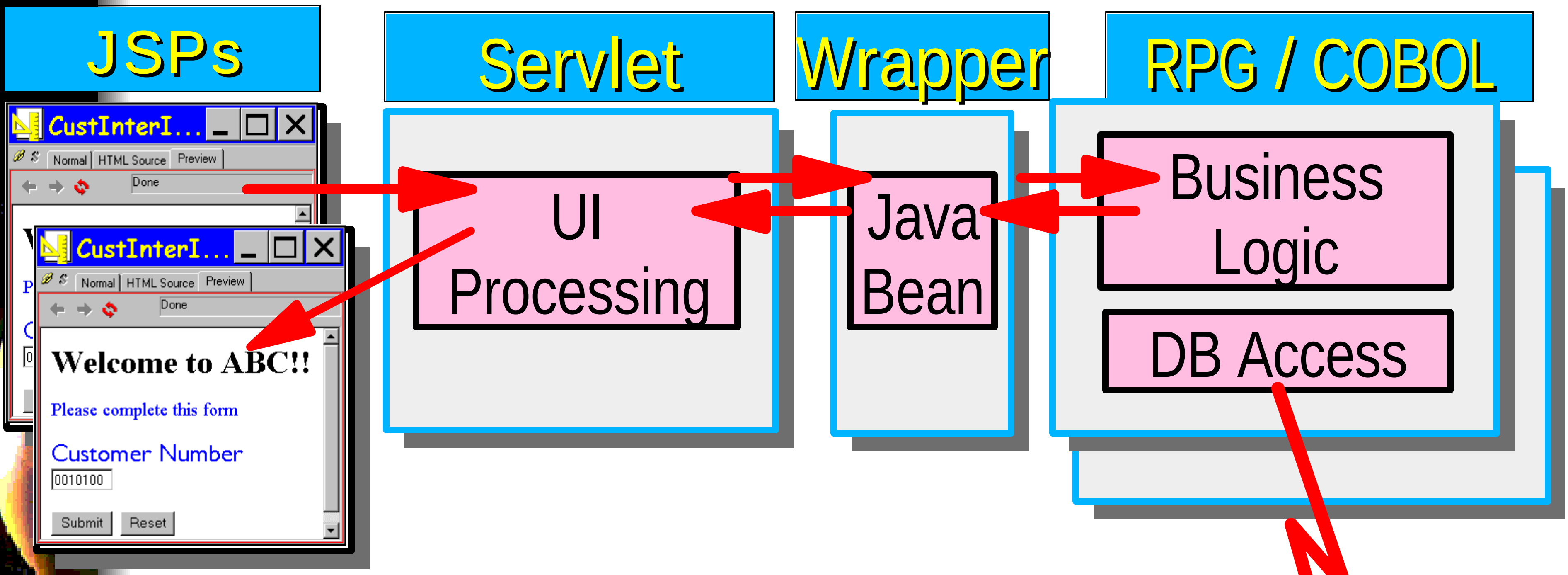
- What is in WDSc 5.0?
- e-business Primer
  - ► AD Model, traditional and web
- Web Application Primer
  - ► JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - ► RPG Example
  - ► iSeries Design time controls (DTCs)
  - ► Web Interaction
  - ► Publishing
- Conclusion

Today's Model



1. **Program puts up screen, waits for input**
2. **Program processes input, does business logic**

eBusiness Application



Welcome to ABC!!

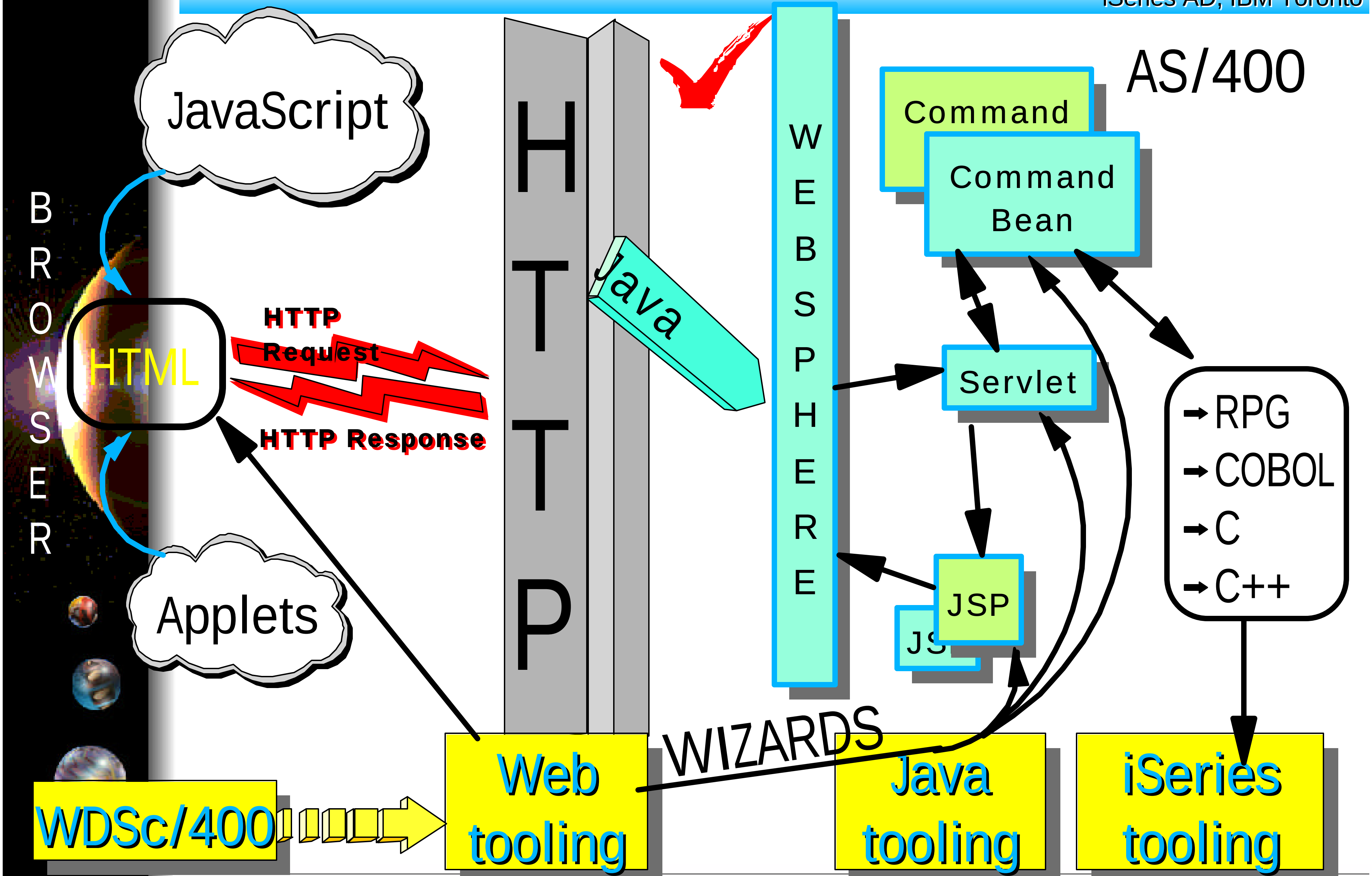
Please complete this form

Customer Number

0010100

1. **JavaServer Pages (JSPs) for UI**
   - ►HTML tags for constant part
   - ►JSP tags for dynamically substituted data
   - ►HTML FORMs for user input fields
2. **Servlets for UI processing**
3. **Java Bean encapsulation of business logic**
4. **Business Logic: *PGM or ILE procedures**

Big Picture



BROWSER

JavaScript

HTML

Applets

HTTP Request

HTTP Response

HTTP

Java

WEBSPHERE

AS/400

Command

Command Bean

Servlet

JSP

JS

- RPG
- COBOL
- C
- C++

WIZARDS

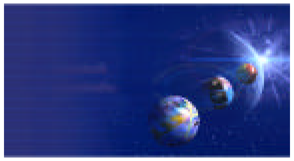
WDSc/400

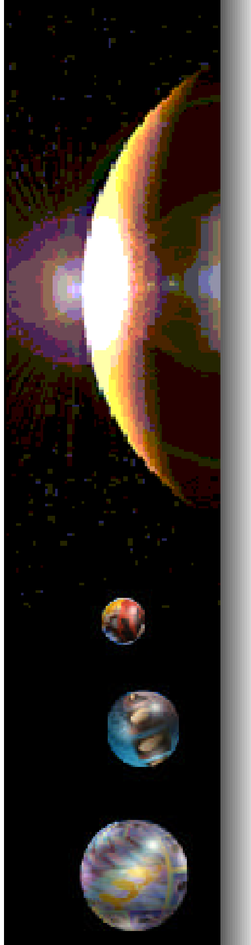
Web tooling

Java tooling

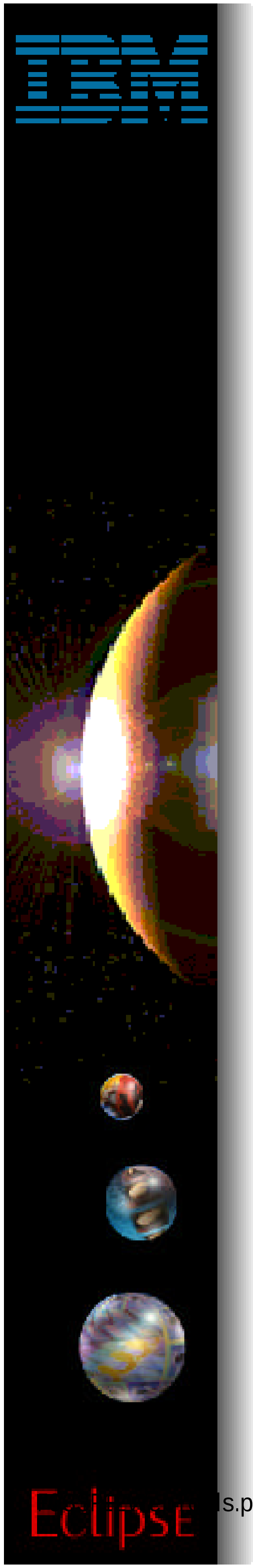
iSeries tooling

AGENDA

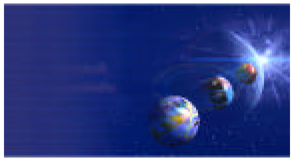




- What is in WDSc 5.0?
- e-business Primer
  - AD Model, traditional and web
- Web Application Primer
  - JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - RPG Example
  - iSeries Design time controls (DTCs)
  - Web Interaction
  - Publishing
- Conclusion



What Are JSPs?



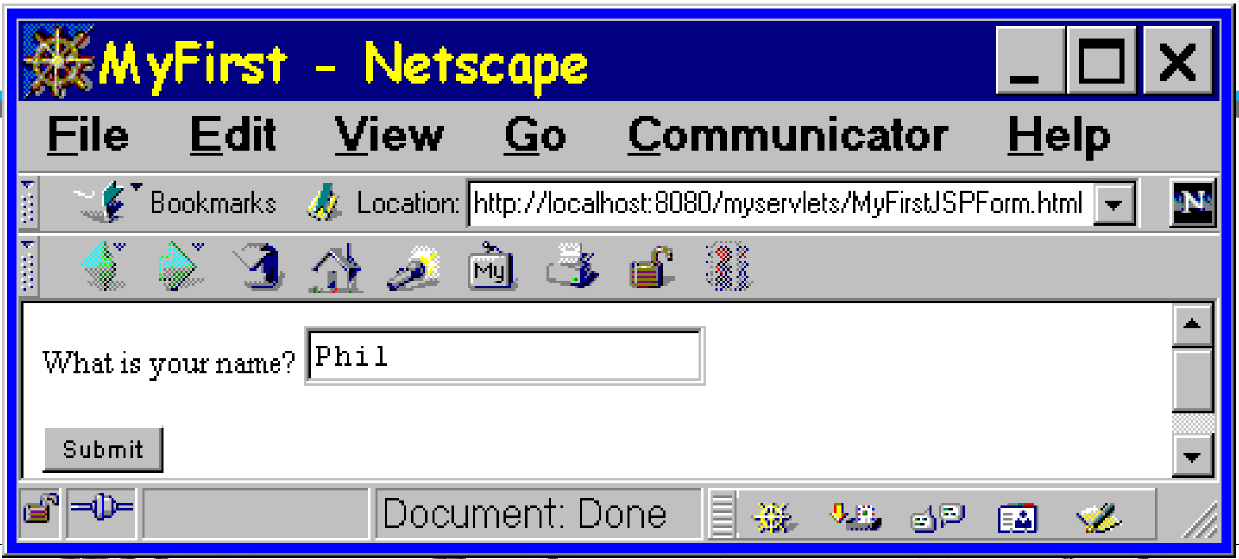
- JavaServer Pages (JSPs) are . . .
  - **.jsp files**
    - containing html tags plus JSP tags
- JSP tags . . .
  - **Allow dynamic data to be inserted into the static HTML**
    - Data is extracted from Java Beans passed to the JSP
- JSPs are called . . .
  - **By your servlet**
  - **The input to JSPs are . . .**
    - Java Beans passed from your Servlet
  - **The output of a JSP is . . .**
    - A full Web page, displayed to user

JSP example . . .

\myservlets\MyFirstJSPForm.html

```
<HTML>
<HEAD><TITLE> MyFirst </TITLE></HEAD>
<BODY>
<FORM METHOD=POST
 ACTION="http://localhost/myservlets/MyFirstjsp.jsp">
What is your name?
<INPUT TYPE="text" NAME="name"><P>
<INPUT TYPE="submit" VALUE="Submit">
</FORM>
</BODY>
</HTML>
```

Call the JSP when SUBMIT pressed



http://localhost:8080/myservlets/MyFirstJSPForm.html

What is your name? Phil

Submit

Document: Done

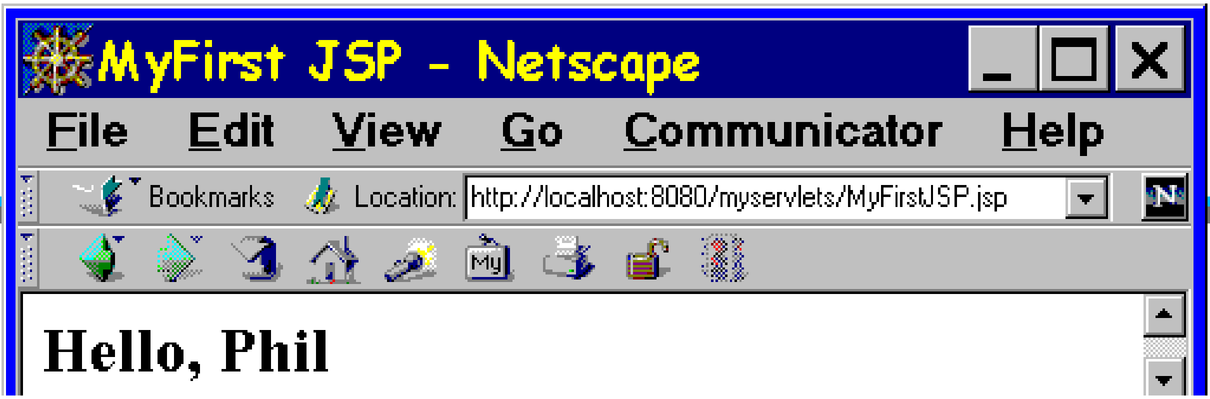
JSP Example . . .

\myservlets\WEB-INF\jsps\MyFirstJSP.jsp

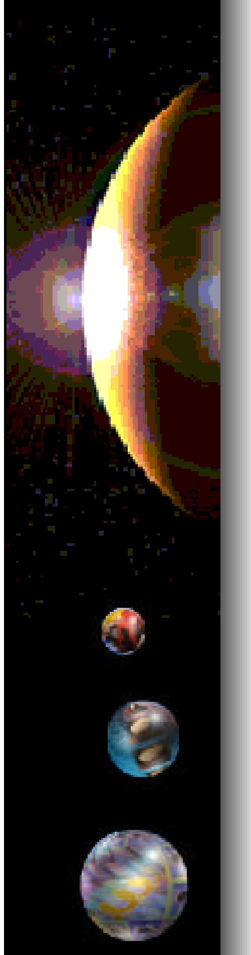
```
<HTML>
<HEAD><TITLE> MyFirst JSP </TITLE></HEAD>
<BODY>
<h1>
<% if
 (request.getParameter("name")
  == null) {%>
Hello World
<%} else {%>
Hello,
<%= request.getParameter("name")%>
<%}%>
</H1>
</BODY>
</HTML>
```

Embedded Java code! Java code is run on the server, not the client. The resulting all-html file is sent to the client

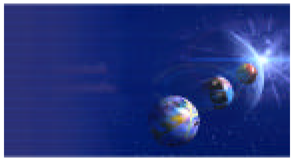
You don't have to compile!



Hello, Phil

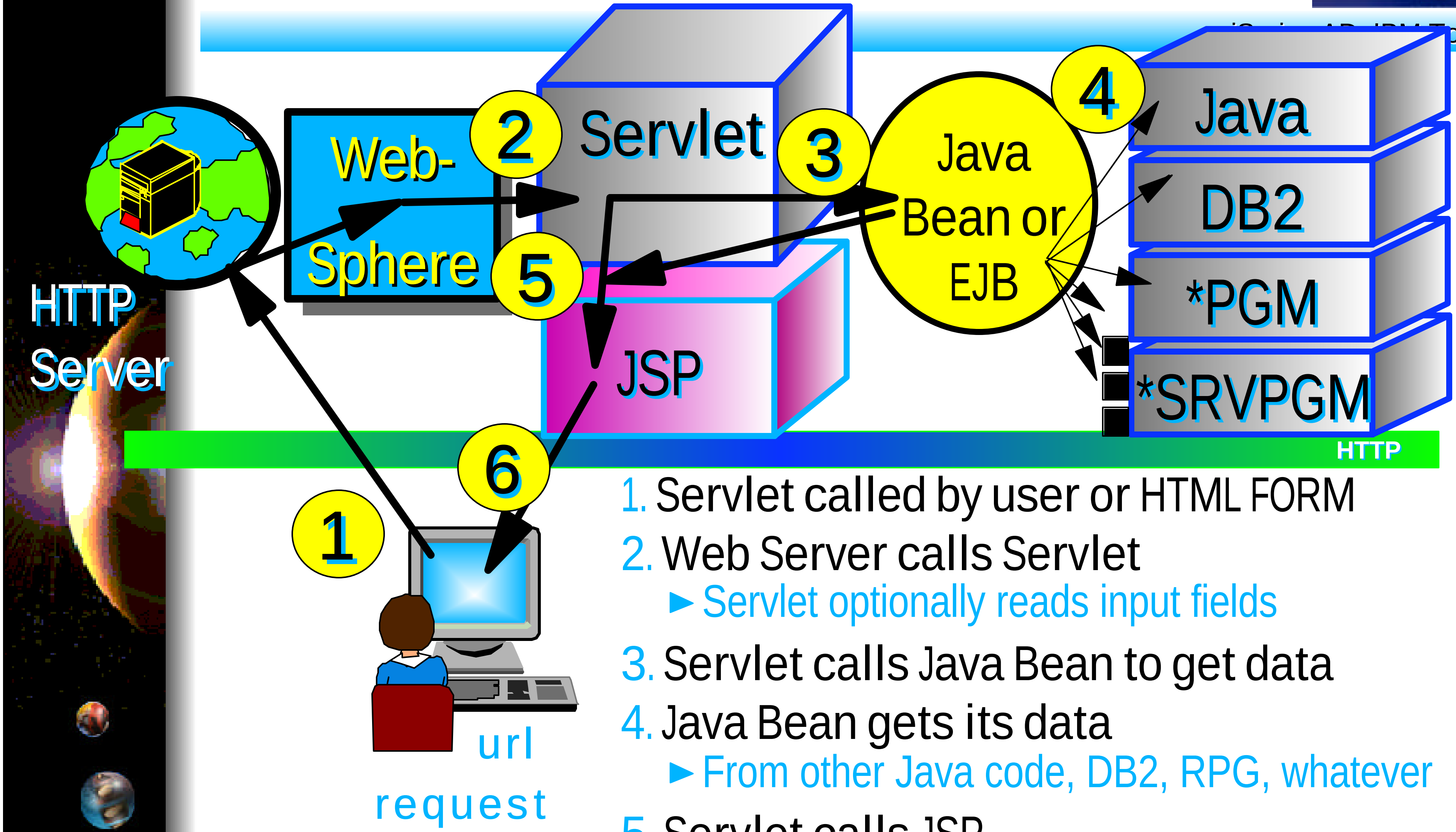


What Are Servlets?

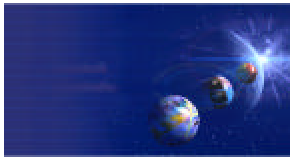


- ► Servlets are . . .
  - ● **Java classes (programs written in Java)**
- ► Servlets run . . .
  - ● **On the server (eg, AS/400)**
- ► Servlets are called . . .
  - ● **By your HTTP Server software**
  - ● **When a user goes to your Web page**
- ► The input to Servlets are . . .
  - ● **User-entered data from a Web page**
- ► The output of a Servlet is . . .
  - ● **Java Bean, passed to a JavaServer Page**

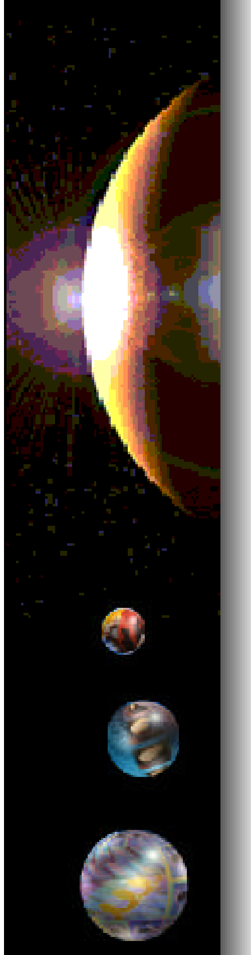
Web Model



1. Servlet called by user or HTML FORM
2. Web Server calls Servlet
   - ► Servlet optionally reads input fields
3. Servlet calls Java Bean to get data
4. Java Bean gets its data
   - ► From other Java code, DB2, RPG, whatever
5. Servlet calls JSP
   - ► Passes Java Bean containing data
   - ► JSP merges data with HTML => new page
6. HTTP Server sends new page to user

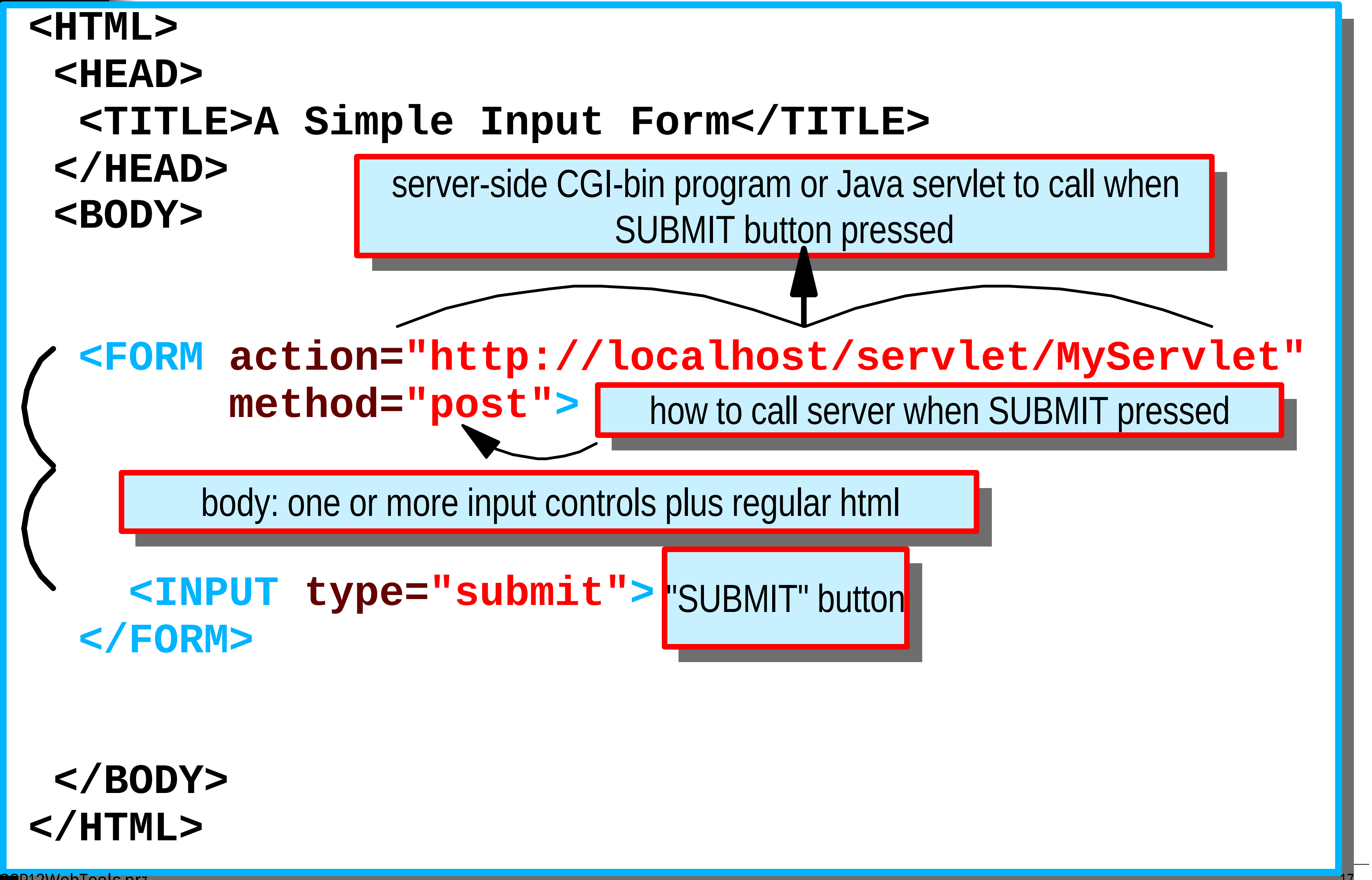


HTML Forms



- Allows you to solicit user input
- Can build HTML document that contains:
  - checkboxes
  - radio buttons
  - push buttons
  - entry fields
  - selectable lists
  - ... others
- Collect data from user
  - **send to server program**
    - The FORM tag specifies the name of the program
      - ✓ Historically a CGI-bin program
      - ✓ New option is a Java Servlet

Anatomy of a FORM



```
<HTML>
 <HEAD>
  <TITLE>A Simple Input Form</TITLE>
 </HEAD>
 <BODY>


  <FORM action="http://localhost/servlet/MyServlet"
        method="post">


    <INPUT type="submit">
  </FORM>



 </BODY>
</HTML>
```

server-side CGI-bin program or Java servlet to call when SUBMIT button pressed

how to call server when SUBMIT pressed

body: one or more input controls plus regular html

"SUBMIT" button

Example of a FORM

```
<FORM action="http://localhost/servlet/MyServlet"
      method="post">
   Name <INPUT type="text" name="name"><BR>
   Age <INPUT type="text" name="age" size="3" maxlength="3"><BR>
   Country <SELECT name="country">
              <OPTION selected>Canada</OPTION>
              <OPTION>Mexico</OPTION>
              <OPTION>United States</OPTION>
           </SELECT> <BR>
   e-mail <INPUT size="30" type="text" name="email"><BR>
   <BR>
   <INPUT type="radio" name="sex" value="M" checked>Male
   <INPUT type="radio" name="sex" value="F">Female<BR><BR>
   <INPUT type="checkbox" name="mail" checked>e-mail me<BR><BR>

   <INPUT type="submit" value="Register">
   <INPUT type="reset"  value="Reset">
</FORM>
```

entry field

selectable list

entry field

radio buttons

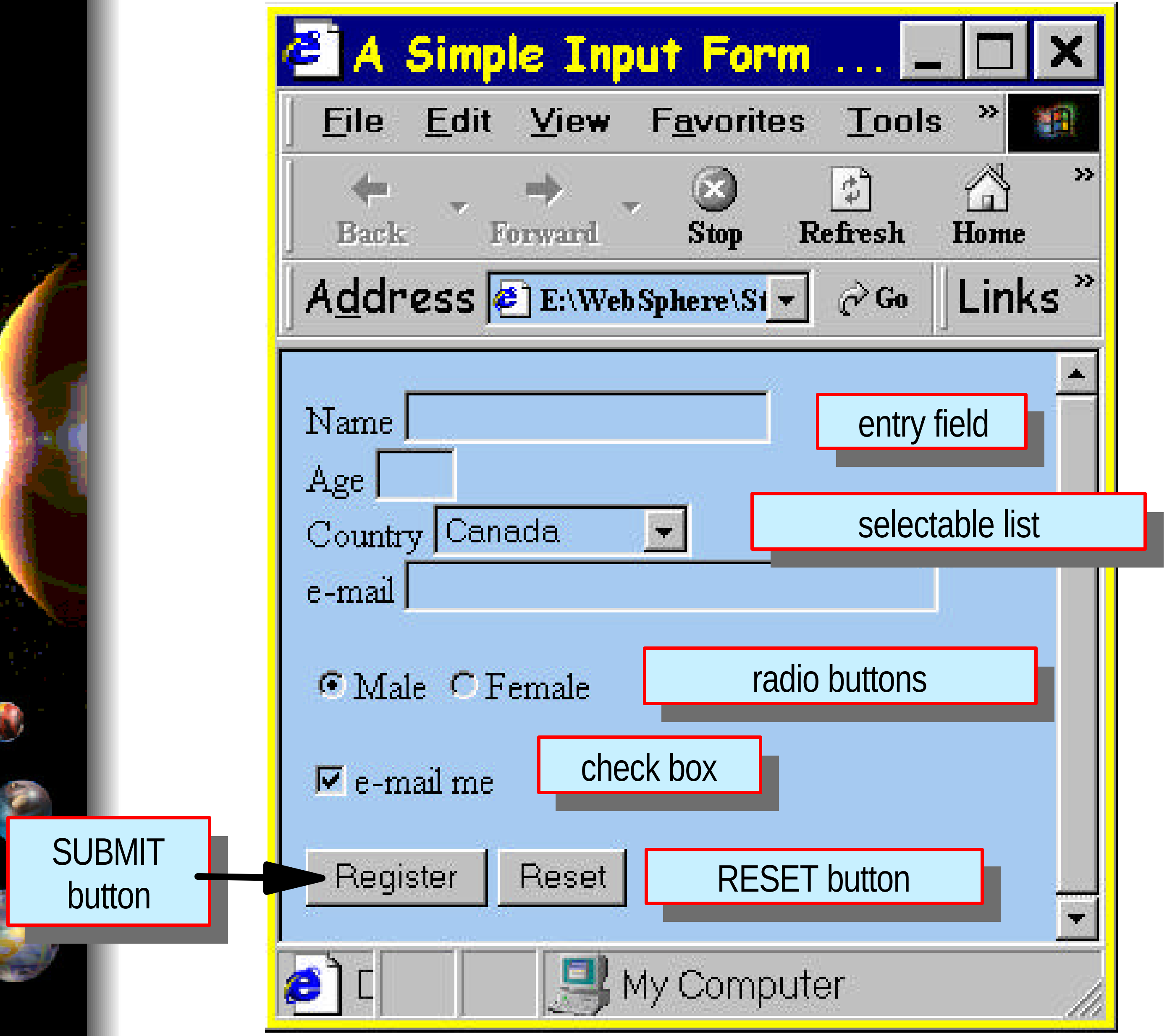
check box

SUBMIT button

RESET button

http://www.w3.org/TR/REC-html40/

Example of a FORM



A Simple Input Form ...

Address E:\WebSphere\St

Name

Age

Country Canada

e-mail

● Male ○ Female

☑ e-mail me

Register Reset

entry field

selectable list

radio buttons

check box

SUBMIT button

RESET button

AGENDA

- What is in WDSc 5.0?
- e-business Primer
  - AD Model, traditional and web
- Web Application Primer
  - JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - RPG Example
  - iSeries Design time controls (DTCs)
  - Web Interaction
  - Publishing
- Conclusion

Java Calling RPG

Say we have the following RPG code . . .

```
FCUSTOML3  IF   E           K DISK
DCUSTINFO         DS
D Number                  1      7A
D Name                    8     47A
C      *ENTRY     PLIST
C                 PARM                 CUSTINFO
C      Number     SETLL     CUSTOM01
C      Number     READE     CUSTOM01            9091
C                 EVAL      Name = CUSTNA
C                 MOVE      *ON             *INLR
**************** End of data ************************
```

Pass in Customer ID and receive back customer name.

Create XML required tags

```
<pcml version="1.0">

 <!-- Create a Data Structure -->
 <struct name="custinfo">
   <data name="Number" type="char" length="7"
         usage="inputoutput" init="0014400"> </data>
   <data name="Name"   type="char" length="40"
         usage="inputoutput" init=" "> </data>
 </struct>

 <!-- Program getcust -->
 <program name="getcust"
          path="/QSYS.lib/FARR.lib/GETCUST.pgm">
   <data name="gotback" type="struct"
         usage="inputoutput" struct="custinfo"> </data>
 </program>

</pcml>
```

Call RPG from Java Servlet

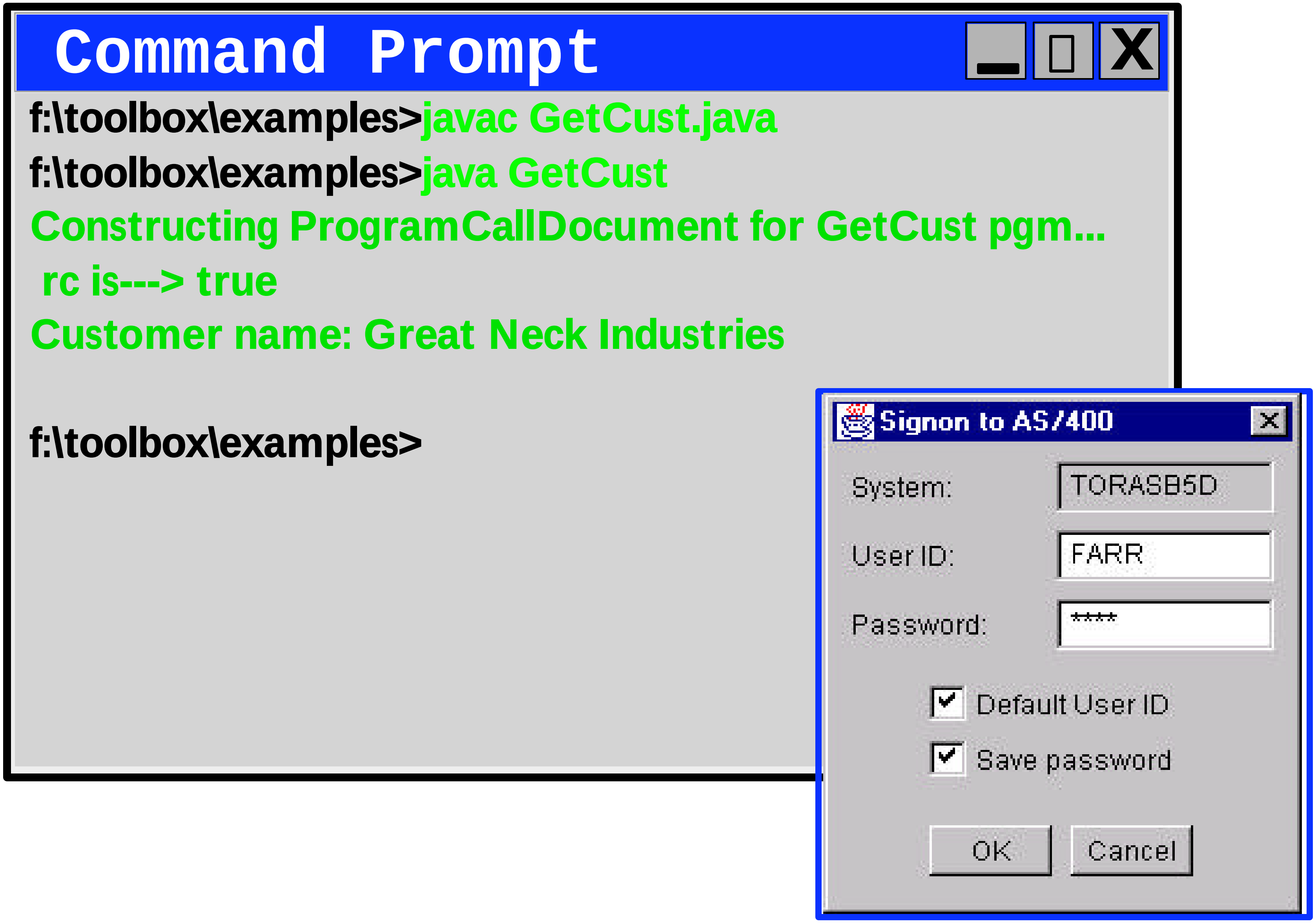
File: GetCust.java

Class: GetCust

```
public static void main(String[] argv)
{
    AS400 as400System = new AS400();
    ProgramCallDocument pcml = null;
    String msgId, msgText;
    Object value = null;

    try {
      System.out.println(
         "Creating ProgramCallDocument for GetCust pgm.");
      pcml = new ProgramCallDocument(as400System, "GETCUST");
      boolean ok = pcml.callProgram("getcust");
      System.out.println(" rc is---> " + rc);
      if (!ok)
        { /* Retrieve list of AS/400 messages & display them */ }
      else
        {
          value = pcml.getValue("getcust.gotback.Name");
          System.out.println("Customer name: " + value);
        }
    } catch (PcmlException exc) {
      System.out.println("*** Call to getcust failed. ***");
      System.exit(0);
    }
    System.exit(0);
} // end main method
```

Mixing Java and RPG

Results . . .



```
f:\toolbox\examples>javac GetCust.java
f:\toolbox\examples>java GetCust
Constructing ProgramCallDocument for GetCust pgm...
 rc is---> true
Customer name: Great Neck Industries

f:\toolbox\examples>
```

System: TORASB5D

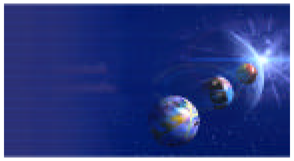
User ID: FARR

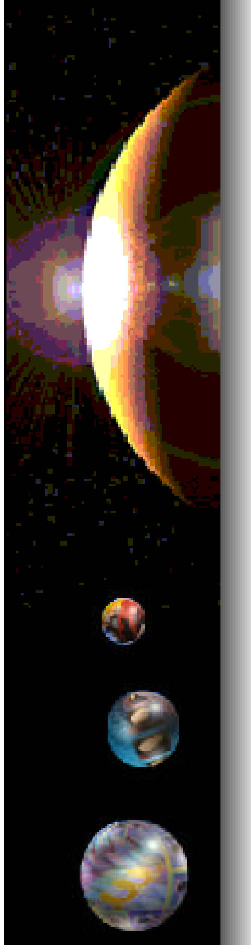
Password: ****

☑ Default User ID

☑ Save password

AGENDA





- What is in WDSc 5.0?
- e-business Primer
  - AD Model, traditional and web
- Web Application Primer
  - JSPs, Servlets, forms, etc
- Mixing Java and RPG
- WebTooling primer
- What is Web Tools for iSeries
- Introducing WDSc for iSeries
  - RPG Example
  - iSeries Design time controls (DTCs)
  - Web Interaction
  - Publishing
- Conclusion

Terms: Web Application

J2EE Servlet Spec 2.2

Web App folder structure:

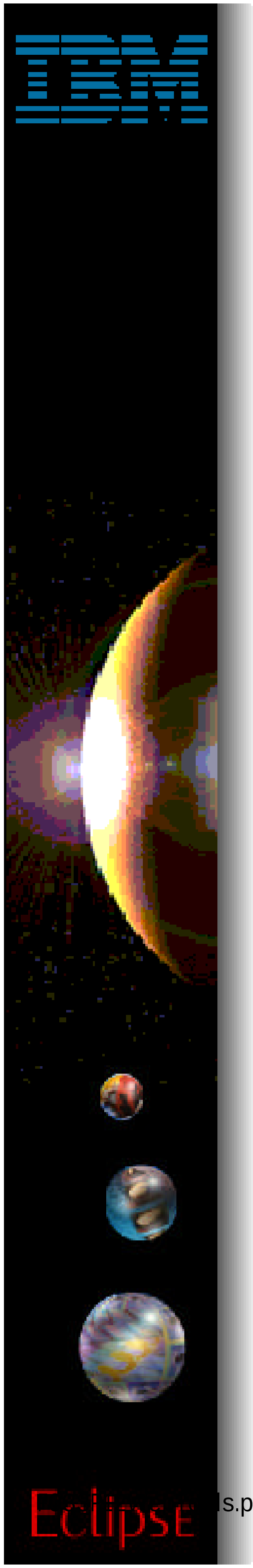
- Web application folder (root folder)
  - **source**
    - *all non-deployed files (java)*
  - **webApplication**
    - *all Web files (html, jsp, gif, ...)*
    - **META-INF**
      - MANIFEST.MF
    - **theme**
      - *.css style sheets*
    - **WEB-INF**
      - web.xml
      - **classes**
        - *Java classes of this app (usually generated)*
      - **lib**
        - *Supporting classes and jar files*

collectively known as "Web Resources"

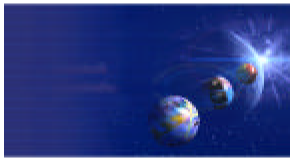
maps dependent jar files in other Web apps

Web application deployment descriptor:

- identifies servlets, security, env vars, mime types, key pages, external references and session configuration info



Terms: Web Application



► Example Web Application

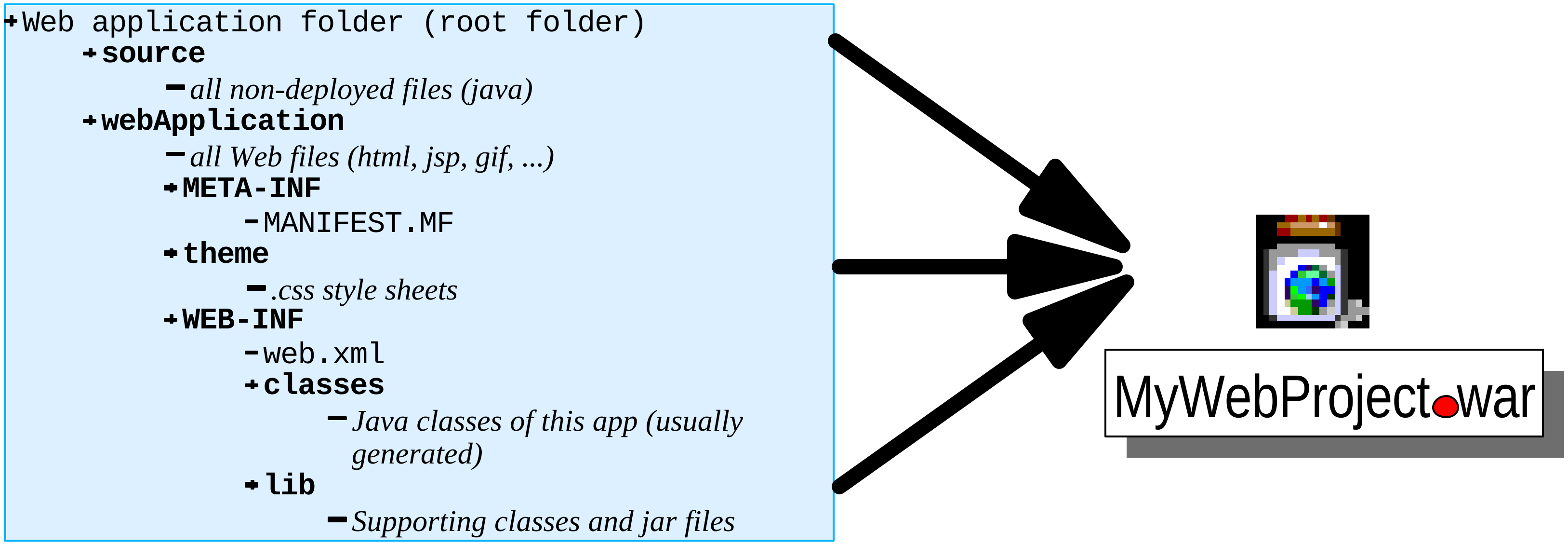
```
+accounts
  +source
  +webApplication
    -index.html          www.mydomain.com/accounts
    +receivable
      -page1.html        www.mydomain.com/accounts/receivable/page1.html
    +payable
      -page1.html        www.mydomain.com/accounts/payable/page1.html
    +META-INF
      -MANIFEST.MF
    +theme
      -corporate.css
    +WEB-INF
      -web.xml
      +classes
      +lib
```

Terms: WAR Files

J2EE Servlet Spec 2.2

Web Archive Files (WAR)

- **One file containing**
  - Whole folder structure of Web application
  - Including web.xml file
  - Optionally including source
- **Used to**
  - Install and configure Web application in an application server



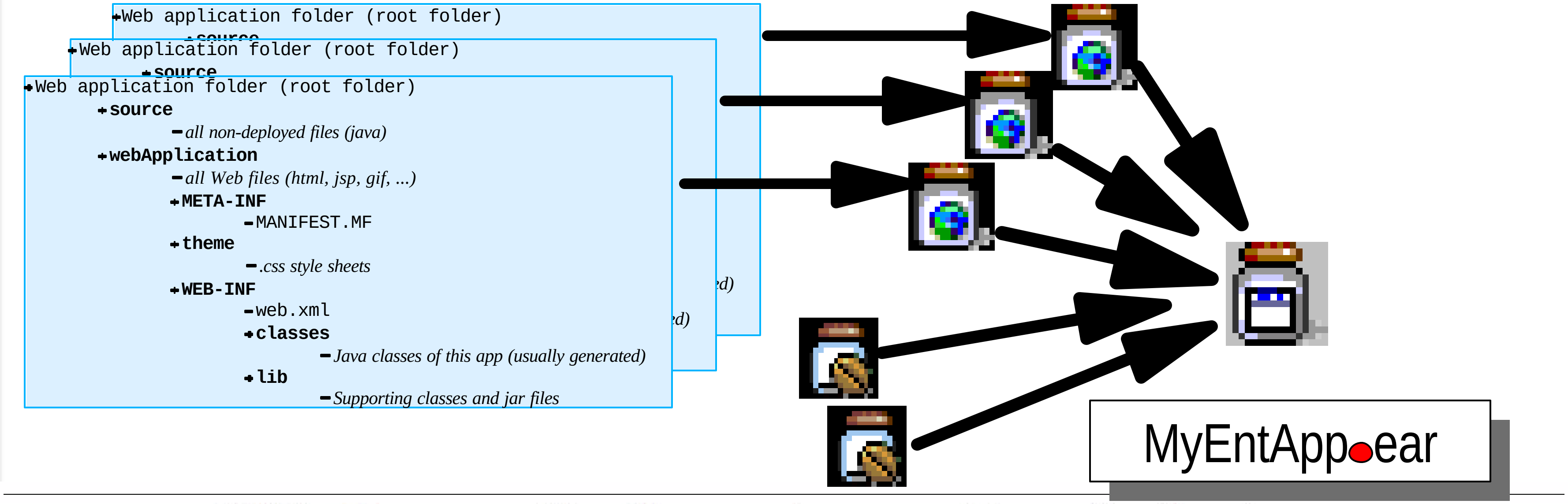
```
+Web application folder (root folder)
    +source
        -all non-deployed files (java)
    +webApplication
        -all Web files (html, jsp, gif, ...)
        +META-INF
            -MANIFEST.MF
        +theme
            -.css style sheets
        +WEB-INF
            -web.xml
            +classes
                -Java classes of this app (usually
                 generated)
            +lib
                -Supporting classes and jar files
```

Terms: EAR Files

J2EE
EJB
Spec 1.1

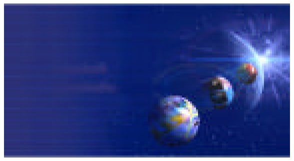
▶ Enterprise Archive Files (EAR)

- **One file containing:**
  - ▶ Zero or more Web Archive (war) files
  - ▶ Zero or more EJB jar files
  - ▶ A J2EE deployment descriptor
- **Used to install and configure:**
  - ▶ All pieces of a J2EE Enterprise Application
    - ✓ Web application plus EBJs plus EJB clients
  - ▶ All Web applications for a Web site (say)

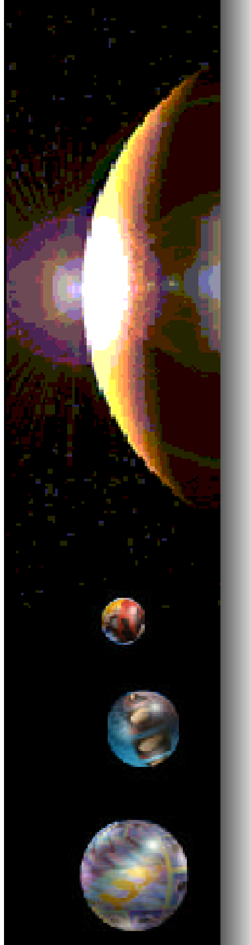


```
+Web application folder (root folder)
+Web application folder (root folder)
     +source
+Web application folder (root folder)
        +source
               -all non-deployed files (java)
        +webApplication
               -all Web files (html, jsp, gif, ...)
               +META-INF
                      -MANIFEST.MF
               +theme
                      -.css style sheets
               +WEB-INF
                      -web.xml
                      +classes
                             -Java classes of this app (usually generated)
                      +lib
                             -Supporting classes and jar files
```

MyEntApp●ear



Web Tools



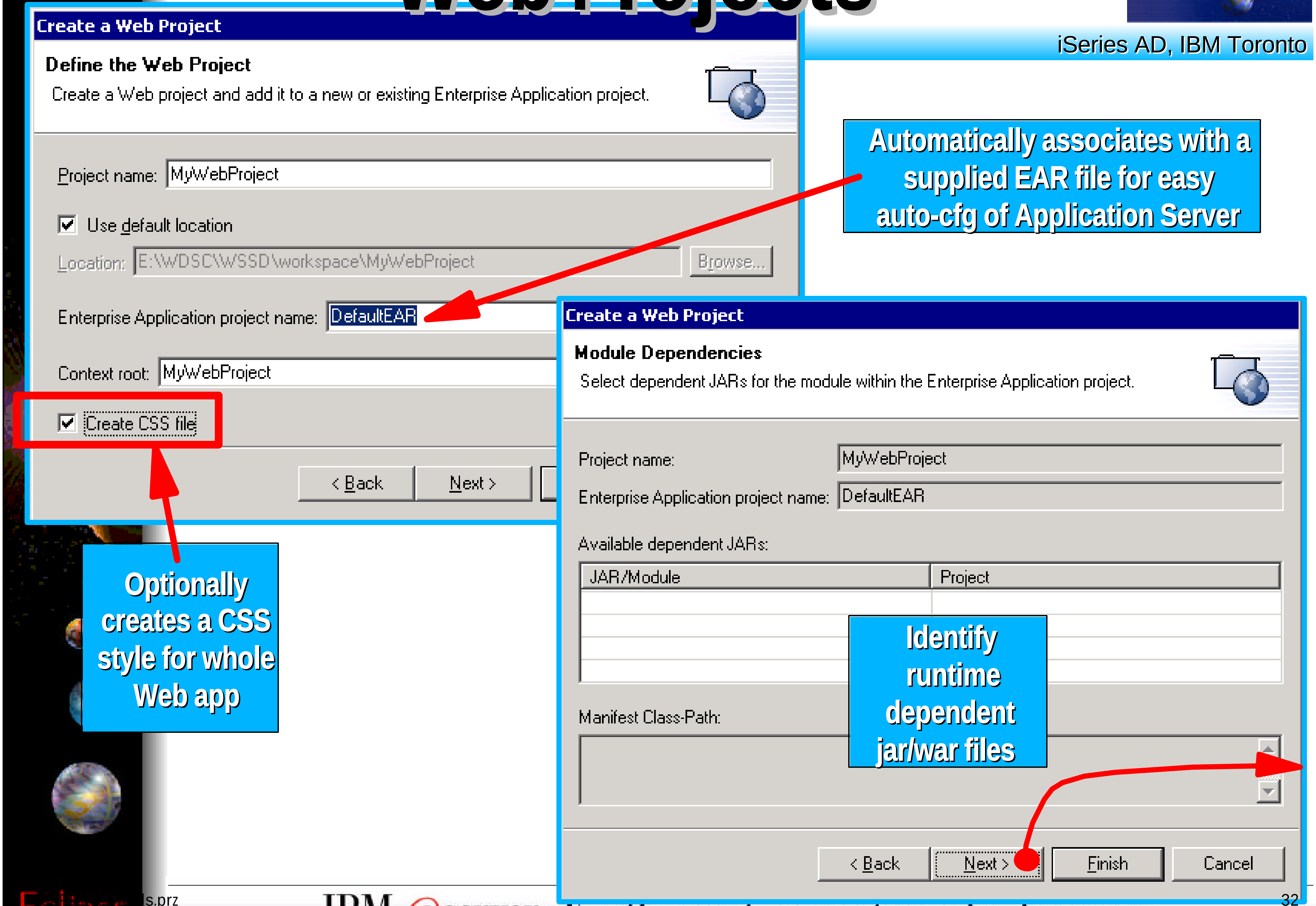
► WDSc Web Tools At A Glance:

- **Web projects**
- **Web Editors**
  - ► For JSP and HTML files
  - ► For Cascading Style Sheets
  - ► For logos, images, and animation
- **Link viewing and management**
- **Import/Export**
  - ► Numerous formats
- **Wizards**
  - ► for servlets, Web pages from DB or JavaBean
- **WebSphere 4.0 Built-in Test Environment**
- **WebSphere Local/Remote Server support**
- **iSeries special support**

Web Projects

- Web Projects
  - Special type of project
    - With its own "new" wizard
    - With its own perspective
    - With its own tools
  - Created with J2EE folder layout
  - Created with simple web.xml file
    - Automatically updated as resources are created
    - Has specialized web.xml editor
- File->New->Other...
  - ->Web->Web Project

Web Projects



Create a Web Project

Define the Web Project

Create a Web project and add it to a new or existing Enterprise Application project.

Project name: MyWebProject

☑ Use default location

Location: E:\WDSC\WSSD\workspace\MyWebProject

Enterprise Application project name: DefaultEAR

Context root: MyWebProject

☑ Create CSS file

Create a Web Project

Module Dependencies

Select dependent JARs for the module within the Enterprise Application project.

Project name: MyWebProject

Enterprise Application project name: DefaultEAR

Available dependent JARs:

JAR/Module	Project

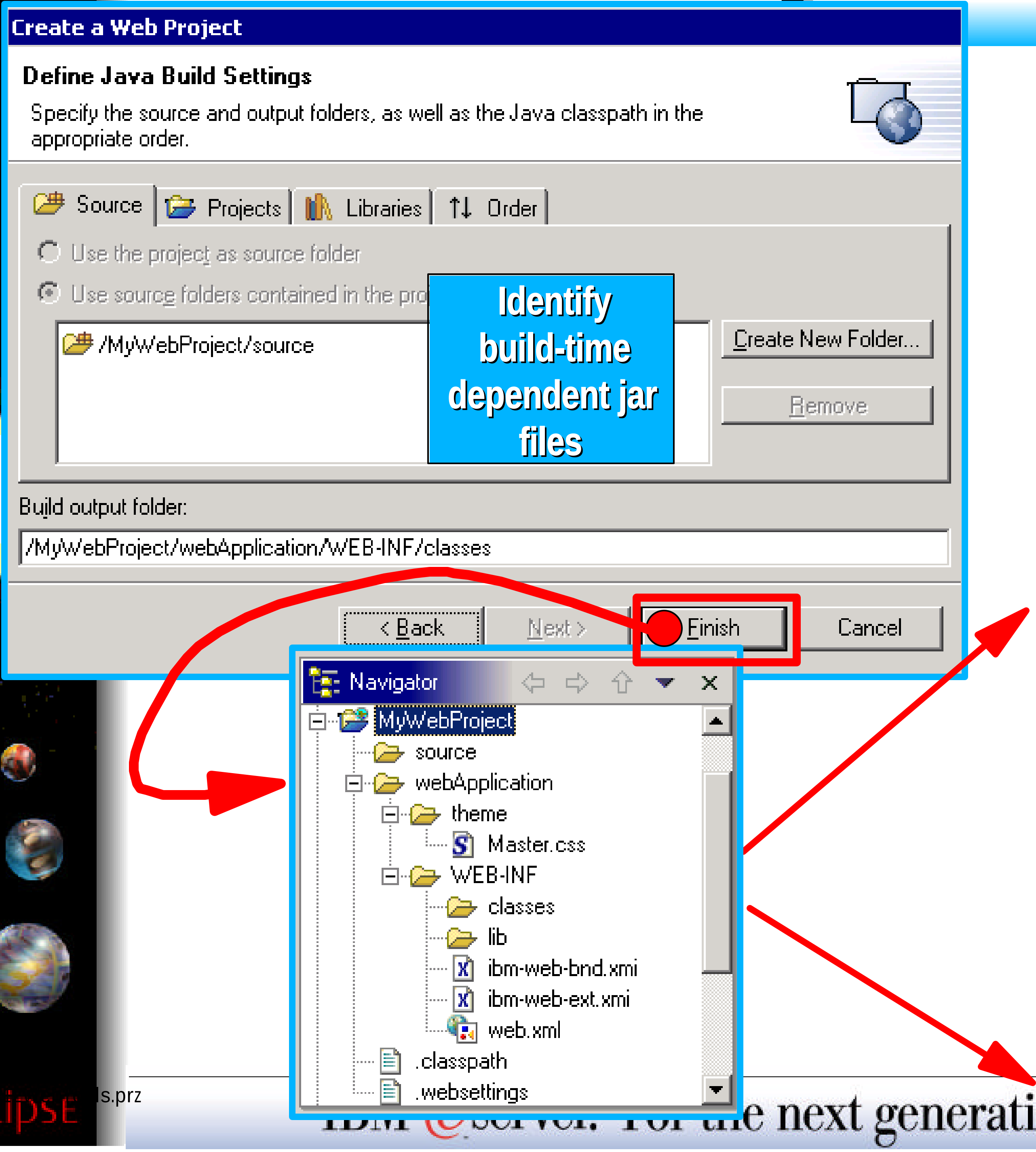
Manifest Class-Path:

Automatically associates with a supplied EAR file for easy auto-cfg of Application Server

Optionally creates a CSS style for whole Web app

Identify runtime dependent jar/war files

Web Projects



Create a Web Project

Define Java Build Settings

Specify the source and output folders, as well as the Java classpath in the appropriate order.

○ Use the project as source folder

● Use source folders contained in the pro

/MyWebProject/source

Build output folder: /MyWebProject/webApplication/WEB-INF/classes

Identify build-time dependent jar files

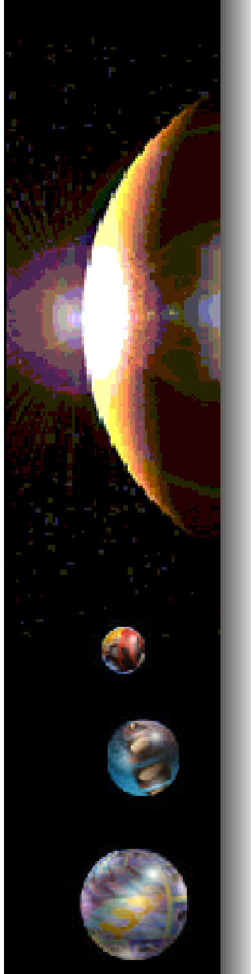
```
MyWebProject
    source
    webApplication
        theme
            Master.css
        WEB-INF
            classes
            lib
            ibm-web-bnd.xmi
            ibm-web-ext.xmi
            web.xml
    .classpath
    .websettings
```

source
- ► for non-deployed files

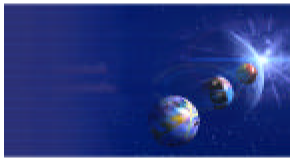
webApplication
- ► for deployed files
- ► .jsps and .html files go here

.../theme
- ► for style sheets

../WEB-INF
- ► run-time dependencies
- ► jar files go here



WDSc: Web Tools



► WDSc Web Tools At A Glance:

- ► Web projects
  - ✓ Created with J2EE-defined folder structure for Web Applications
  - ✓ Superset of Java projects (so contain all Java Tool support too)
- ► Automatic creation/maintainence of web.xml file
- ► Editor support
  - ✓ JSP and HTML files
    - ✓ Support for creating, validating, editing and debugging
    - ✓ Including WYSIWYG PageDesigner
    - ✓ Custom JSP tags (taglib) support
    - ✓ based on the Sun Microsystems JSP 1.1 Specification
  - ✓ Images and animation
  - ✓ Cascading Syle Sheets (CSS)
- ► Import/Export from/to a variety of sources
  - ✓ HTTP/FTP/WAR
- ► Link viewing and management
  - ✓ Converting links, flagging broken links, and fixing up links as linked resources are moved or renamed
- ► Wizard for servlets, Web pages from DB or JavaBean
- ► Integration with WebSphere Unit Test Environment

Web tooling Components

Page Designer For JSPs & HTML

WorkBench

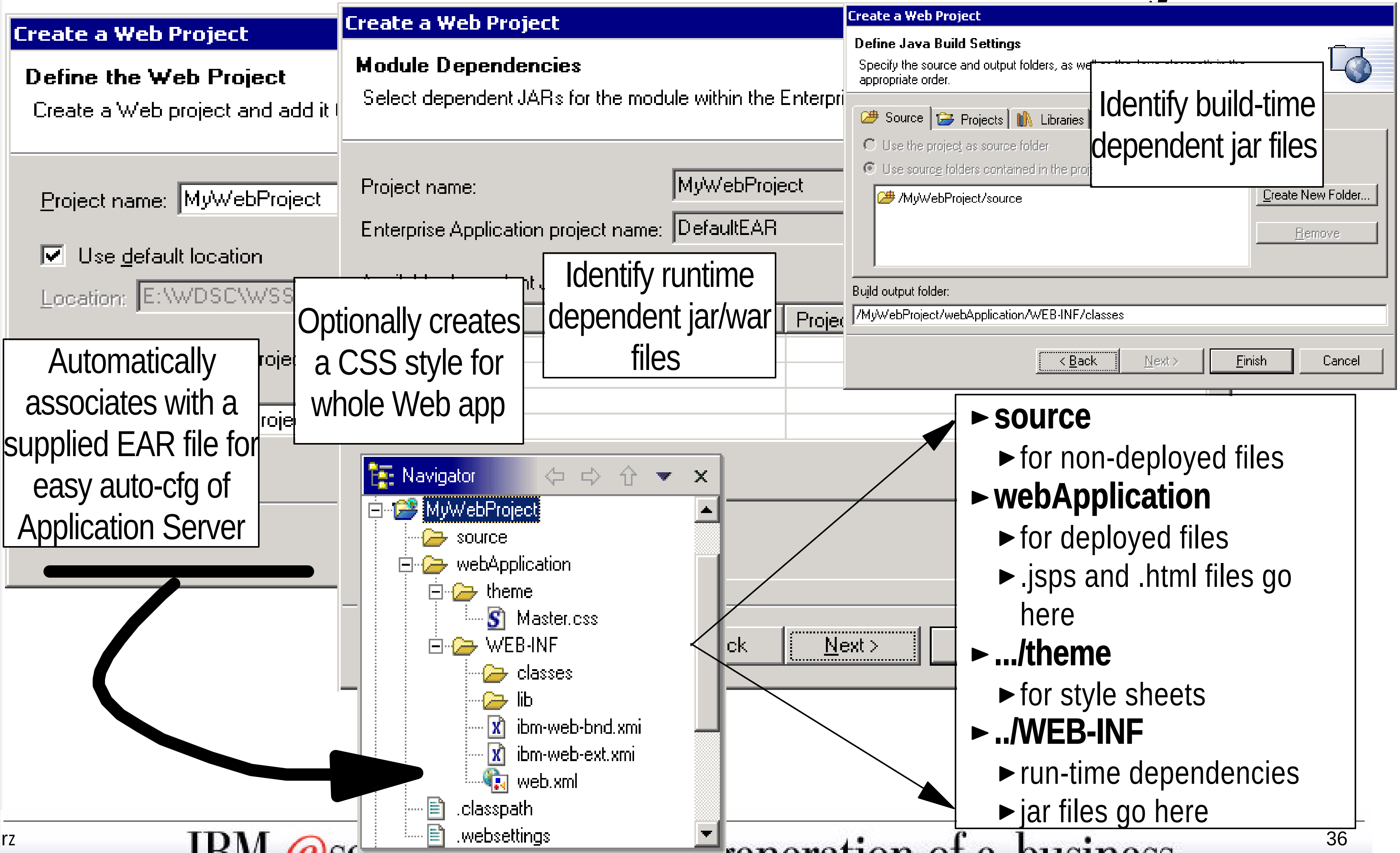
iSeries Tight Integration

Many Wizards

WDSc: Web Tools

▶ Web projects

- **File->New->Other->Web->Web Project**



Create a Web Project

Define the Web Project

Create a Web project and add it t

Project name: MyWebProject

☑ Use default location

Location: E:\WDSC\WSS

Create a Web Project

Module Dependencies

Select dependent JARs for the module within the Enterpri

Project name: MyWebProject

Enterprise Application project name: DefaultEAR

Available dependent J

Create a Web Project

Define Java Build Settings

Specify the source and output folders, as we... appropriate order.

○ Use the project as source folder

● Use source folders contained in the proj

/MyWebProject/source

Build output folder: /MyWebProject/webApplication/WEB-INF/classes

```
MyWebProject
  source
  webApplication
    theme
      Master.css
    WEB-INF
      classes
      lib
      ibm-web-bnd.xmi
      ibm-web-ext.xmi
      web.xml
  .classpath
  .websettings
```

Automatically associates with a supplied EAR file for easy auto-cfg of Application Server

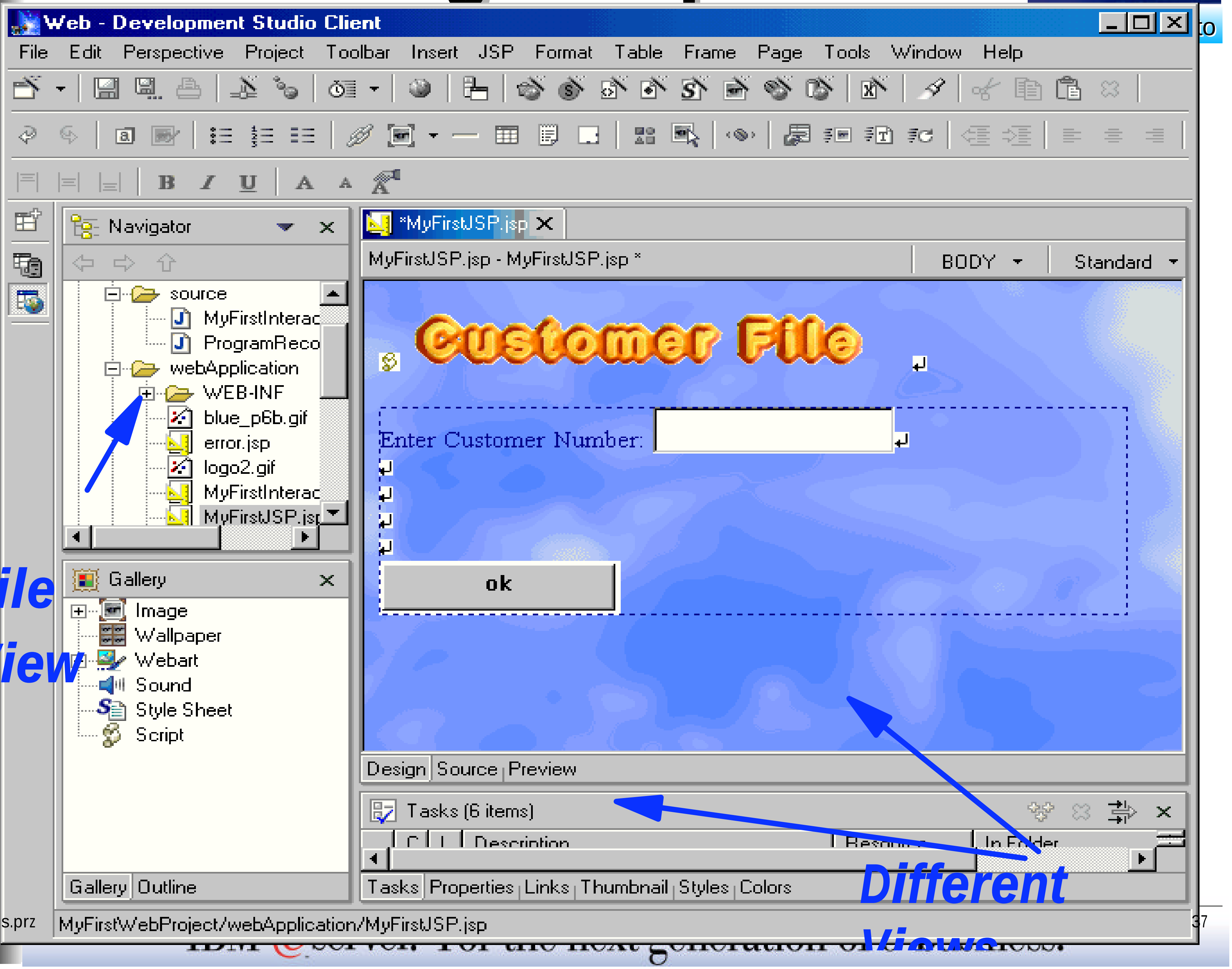
Optionally creates a CSS style for whole Web app

Identify runtime dependent jar/war files

Identify build-time dependent jar files

- **source**
  - for non-deployed files
- **webApplication**
  - for deployed files
  - .jsps and .html files go here
- **.../theme**
  - for style sheets
- **../WEB-INF**
  - run-time dependencies
  - jar files go here

WebTooling Perspective!

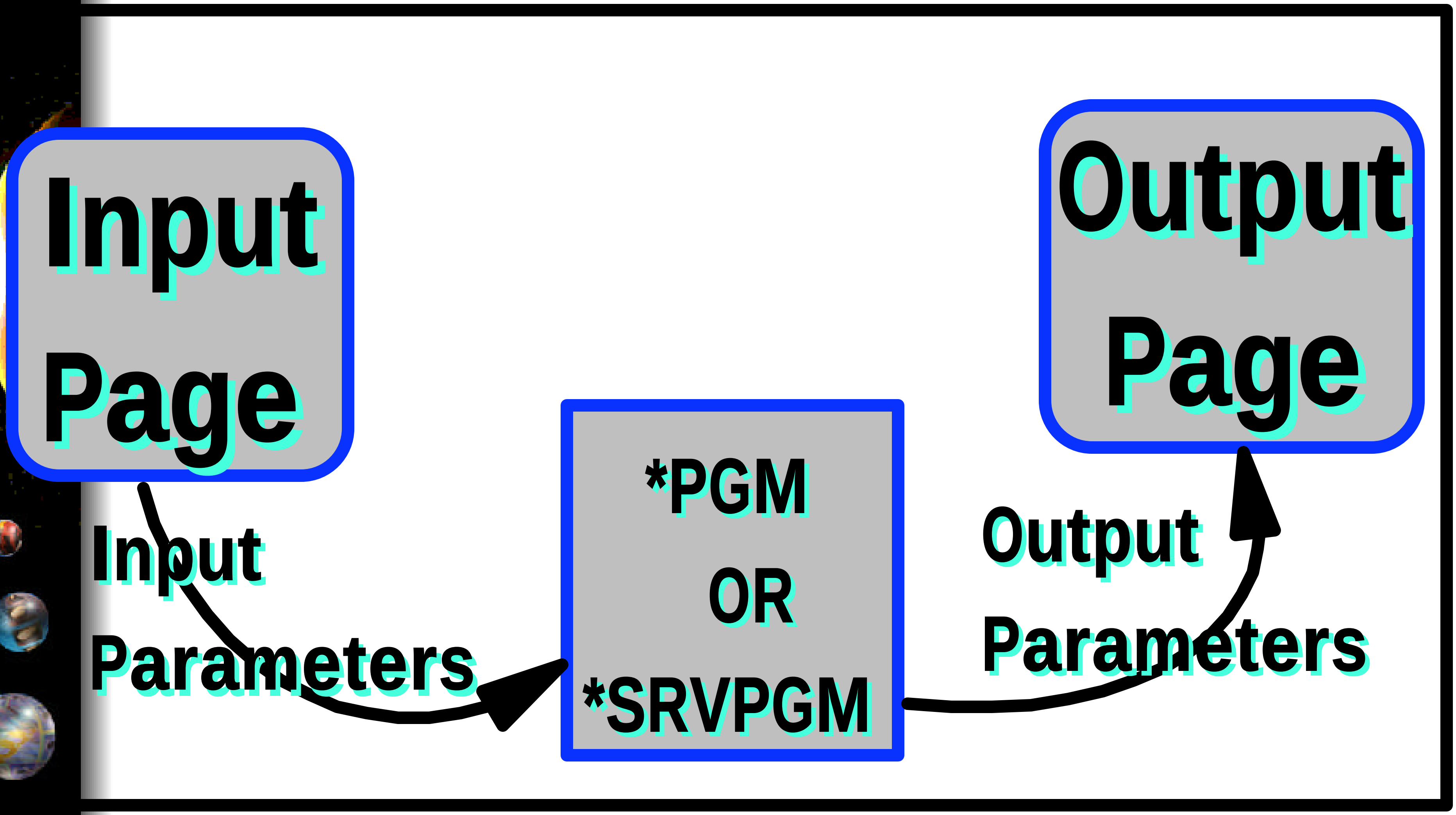


AGENDA

- What is in WDSc 5.0?
- e-business Primer
  - AD Model, traditional and web
- Web Application Primer
  - JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - RPG Example
  - iSeries Design time controls (DTCs)
  - Web Interaction
  - Publishing
- Conclusion

An Interaction . . .

An interaction



Input Page

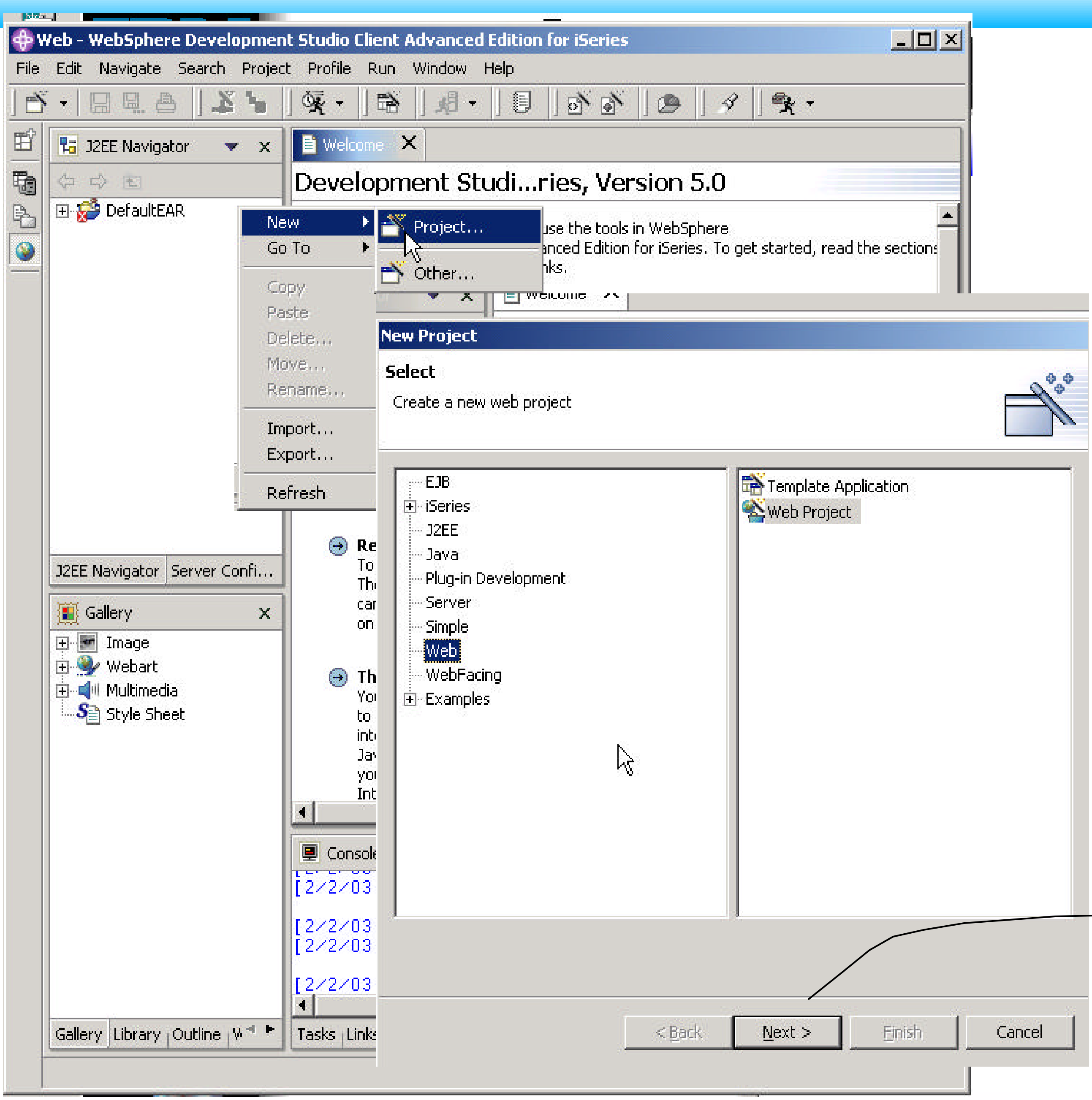
Input Parameters

*PGM OR *SRVPGM

Output Parameters

Output Page

Create a Web project



...use the tools in WebSphere ...anced Edition for iSeries. To get started, read the sections ...nks.

New Project

Select

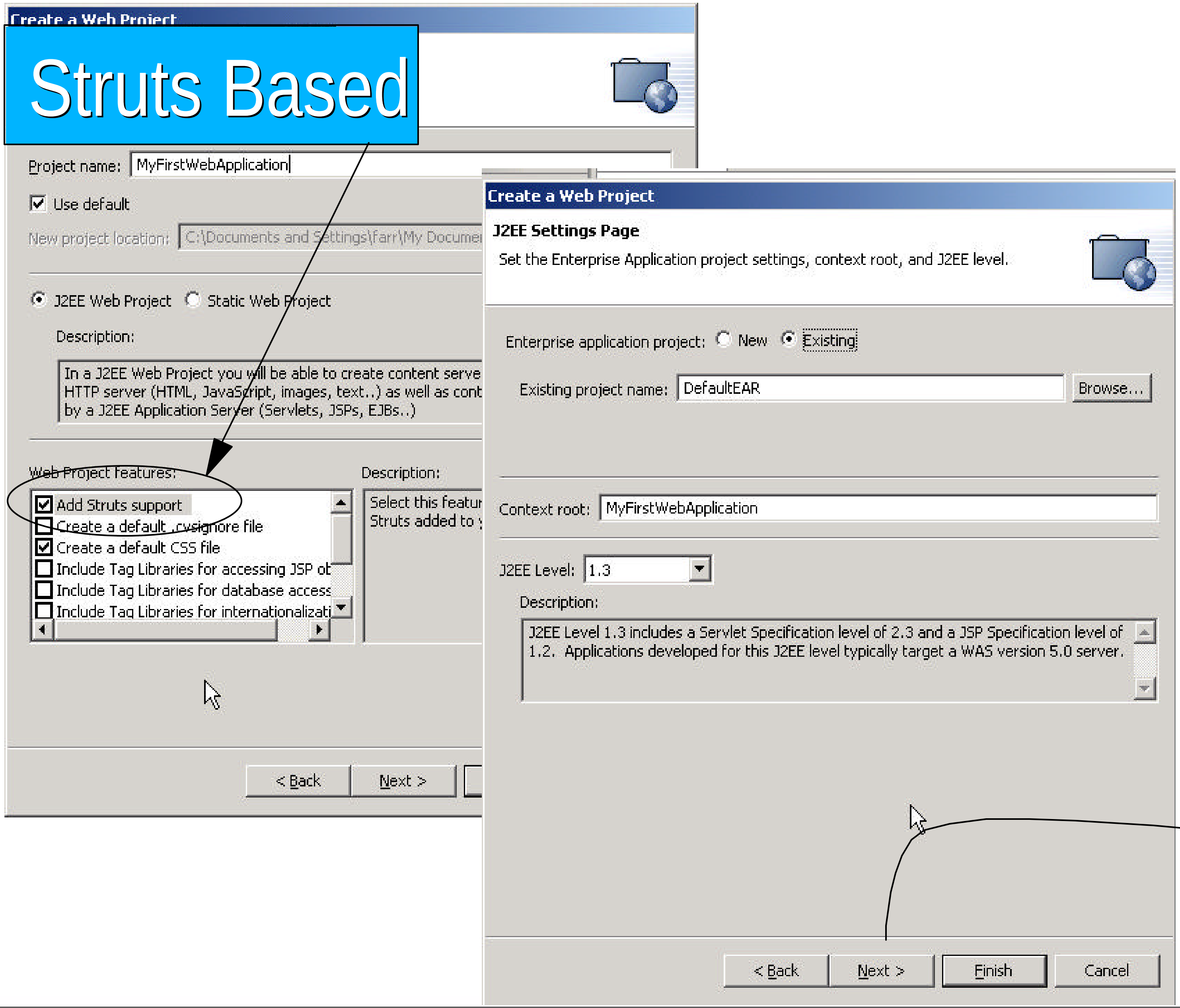
Create a new web project

- EJB
- iSeries
- J2EE
- Java
- Plug-in Development
- Server
- Simple
- Web
- WebFacing
- Examples

- Template Application
- Web Project

Next page

Create a Web Project



Struts Based

Create a Web Project

Project name: MyFirstWebApplication

☑ Use default

New project location: C:\Documents and Settings\farr\My Documen

● J2EE Web Project ○ Static Web Project

Description:

In a J2EE Web Project you will be able to create content serve
HTTP server (HTML, JavaScript, images, text..) as well as cont
by a J2EE Application Server (Servlets, JSPs, EJBs..)

Web Project features:

- ☑ Add Struts support
- ☐ Create a default .cvsignore file
- ☑ Create a default CSS file
- ☐ Include Tag Libraries for accessing JSP ob
- ☐ Include Tag Libraries for database access
- ☐ Include Tag Libraries for internationalizati

Description:

Select this featur
Struts added to y

< Back Next >

Create a Web Project

J2EE Settings Page

Set the Enterprise Application project settings, context root, and J2EE level.

Enterprise application project: ○ New ● Existing

Existing project name: DefaultEAR Browse...

Context root: MyFirstWebApplication

J2EE Level: 1.3

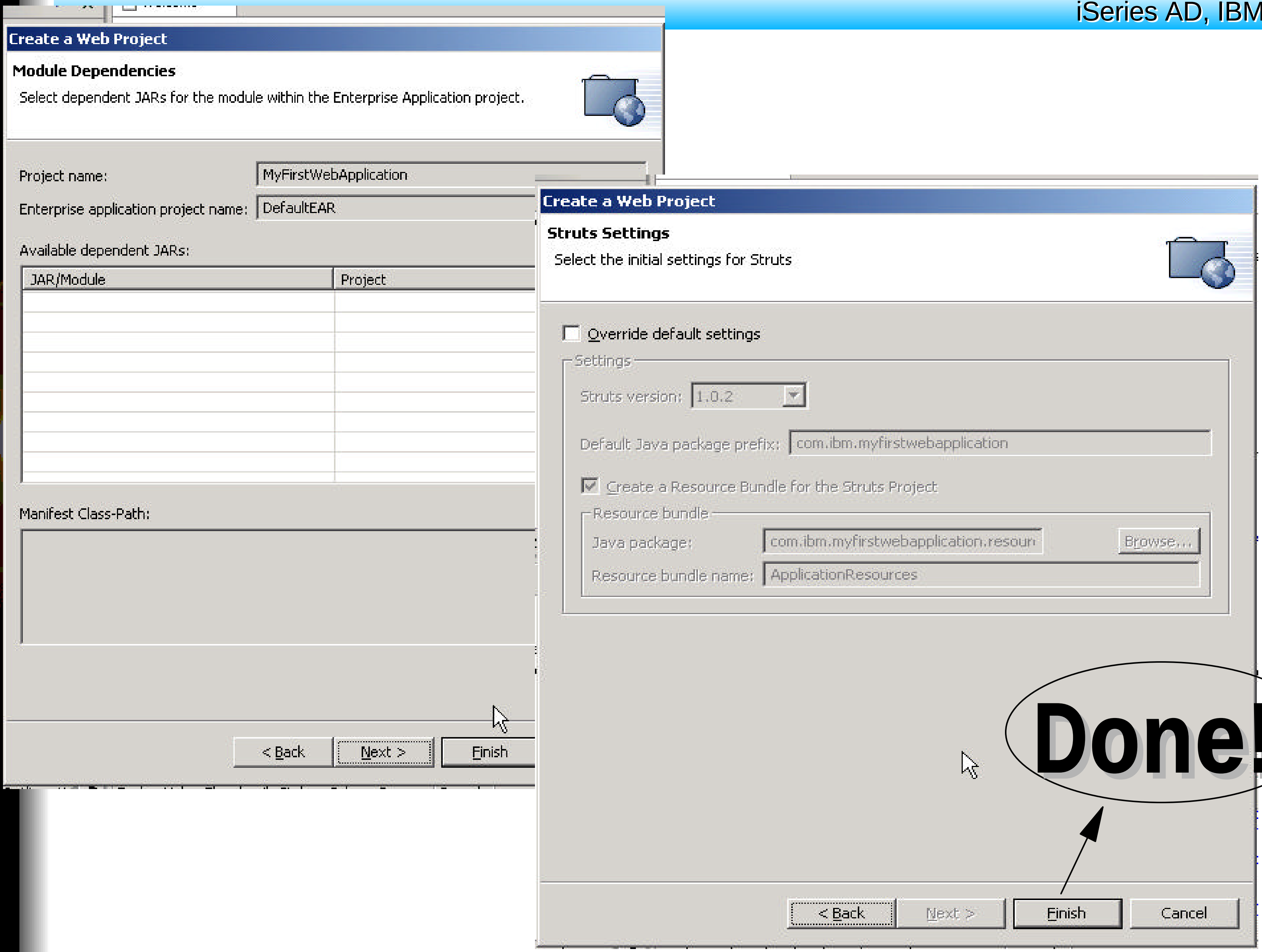
Description:

J2EE Level 1.3 includes a Servlet Specification level of 2.3 and a JSP Specification level of 1.2. Applications developed for this J2EE level typically target a WAS version 5.0 server.

< Back Next > Finish Cancel

Next page

Create a Web Project



Create a Web Project

Module Dependencies

Select dependent JARs for the module within the Enterprise Application project.

Project name: MyFirstWebApplication

Enterprise application project name: DefaultEAR

Available dependent JARs:

JAR/Module	Project

Manifest Class-Path:

Create a Web Project

Struts Settings

Select the initial settings for Struts

□ Override default settings

Settings

Struts version: 1.0.2

Default Java package prefix: com.ibm.myfirstwebapplication

☑ Create a Resource Bundle for the Struts Project

Resource bundle

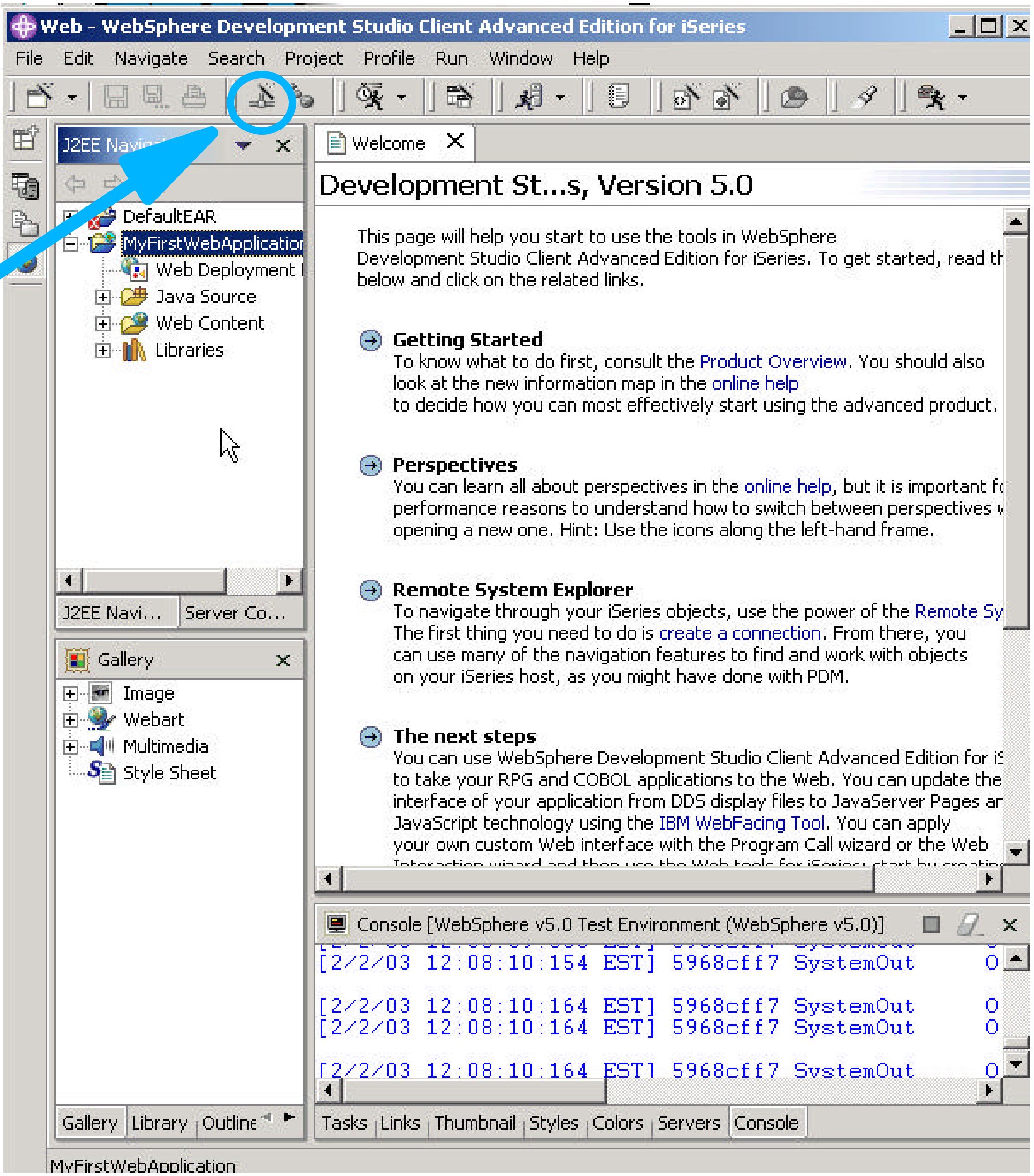
Java package: com.ibm.myfirstwebapplication.resour

Resource bundle name: ApplicationResources

Done!

Server Information

iSeries RunTime configuration!



Development St...s, Version 5.0

This page will help you start to use the tools in WebSphere Development Studio Client Advanced Edition for iSeries. To get started, read th below and click on the related links.

Getting Started
To know what to do first, consult the Product Overview. You should also look at the new information map in the online help to decide how you can most effectively start using the advanced product.

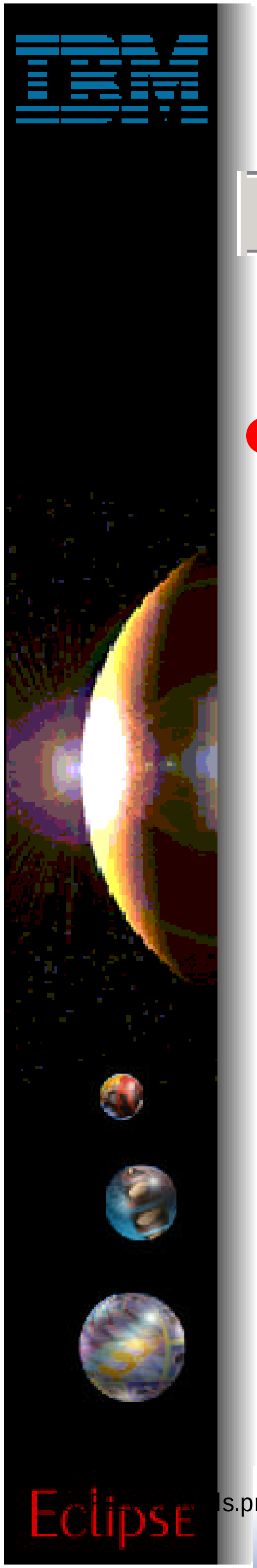
Perspectives
You can learn all about perspectives in the online help, but it is important fo performance reasons to understand how to switch between perspectives v opening a new one. Hint: Use the icons along the left-hand frame.

Remote System Explorer
To navigate through your iSeries objects, use the power of the Remote Sy The first thing you need to do is create a connection. From there, you can use many of the navigation features to find and work with objects on your iSeries host, as you might have done with PDM.

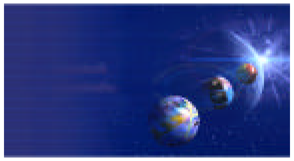
The next steps
You can use WebSphere Development Studio Client Advanced Edition for iS to take your RPG and COBOL applications to the Web. You can update the interface of your application from DDS display files to JavaServer Pages ar JavaScript technology using the IBM WebFacing Tool. You can apply your own custom Web interface with the Program Call wizard or the Web

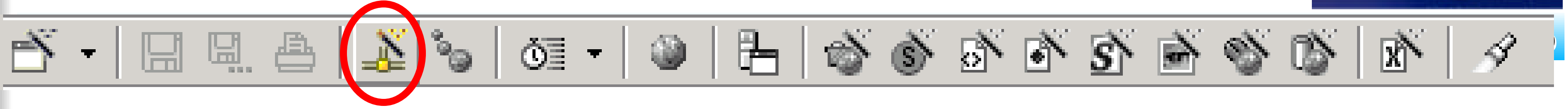
Console [WebSphere v5.0 Test Environment (WebSphere v5.0)]

```
[2/2/03 12:08:10:154 EST] 5968cff7 SystemOut     O

[2/2/03 12:08:10:164 EST] 5968cff7 SystemOut     O
[2/2/03 12:08:10:164 EST] 5968cff7 SystemOut     O

[2/2/03 12:08:10:164 EST] 5968cff7 SystemOut     O
```



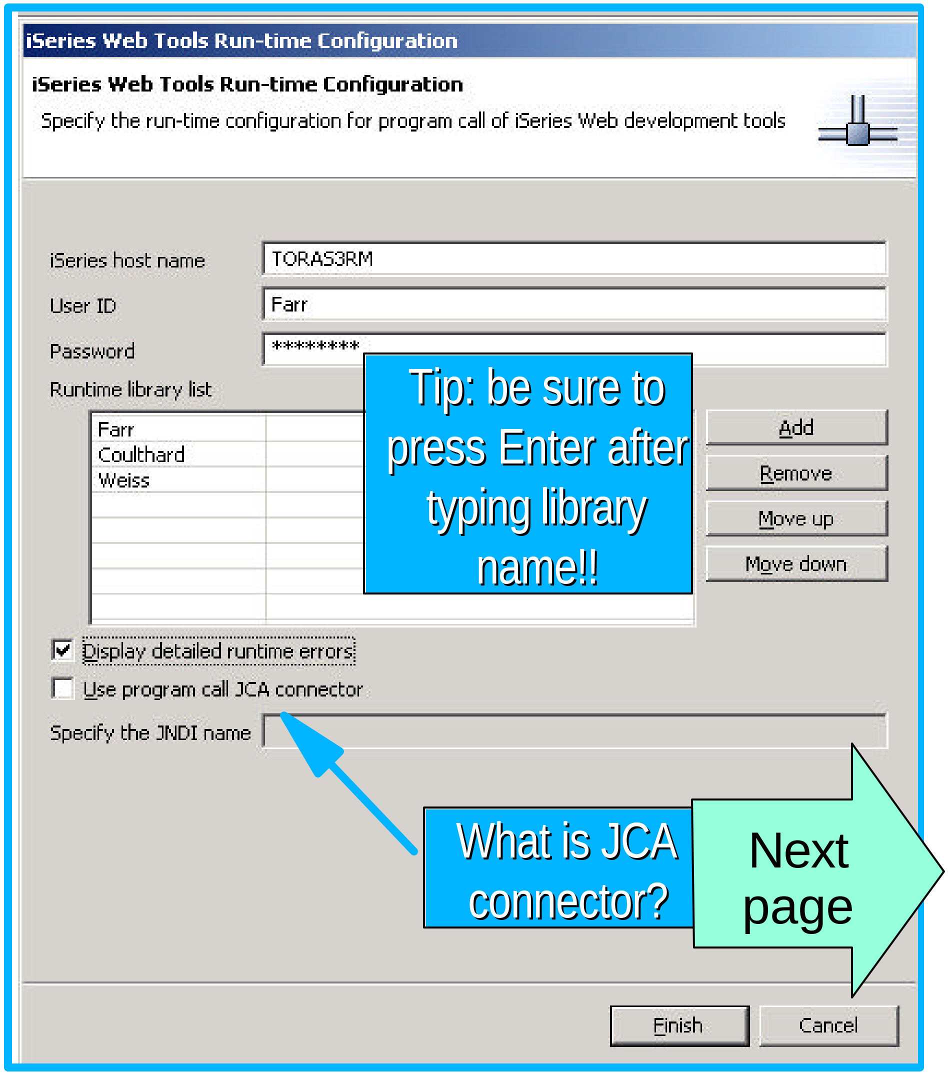
iSeries Web Tools





▶ iSeries Web Host Information Wiz

- **Information used at runtime by all generated Web servlets in this Web project**
  - ▶ **Name of host containing *PGM or ILE Procedure**
  - ▶ **User ID and password for signing on**
  - ▶ **Library list to set for *PGM/Proc job**



iSeries Web Tools Run-time Configuration

iSeries Web Tools Run-time Configuration

Specify the run-time configuration for program call of iSeries Web development tools

iSeries host name	TORAS3RM
User ID	Farr
Password	*********

Runtime library list

Farr
Coulthard
Weiss

☑ Display detailed runtime errors

☐ Use program call JCA connector

Specify the JNDI name

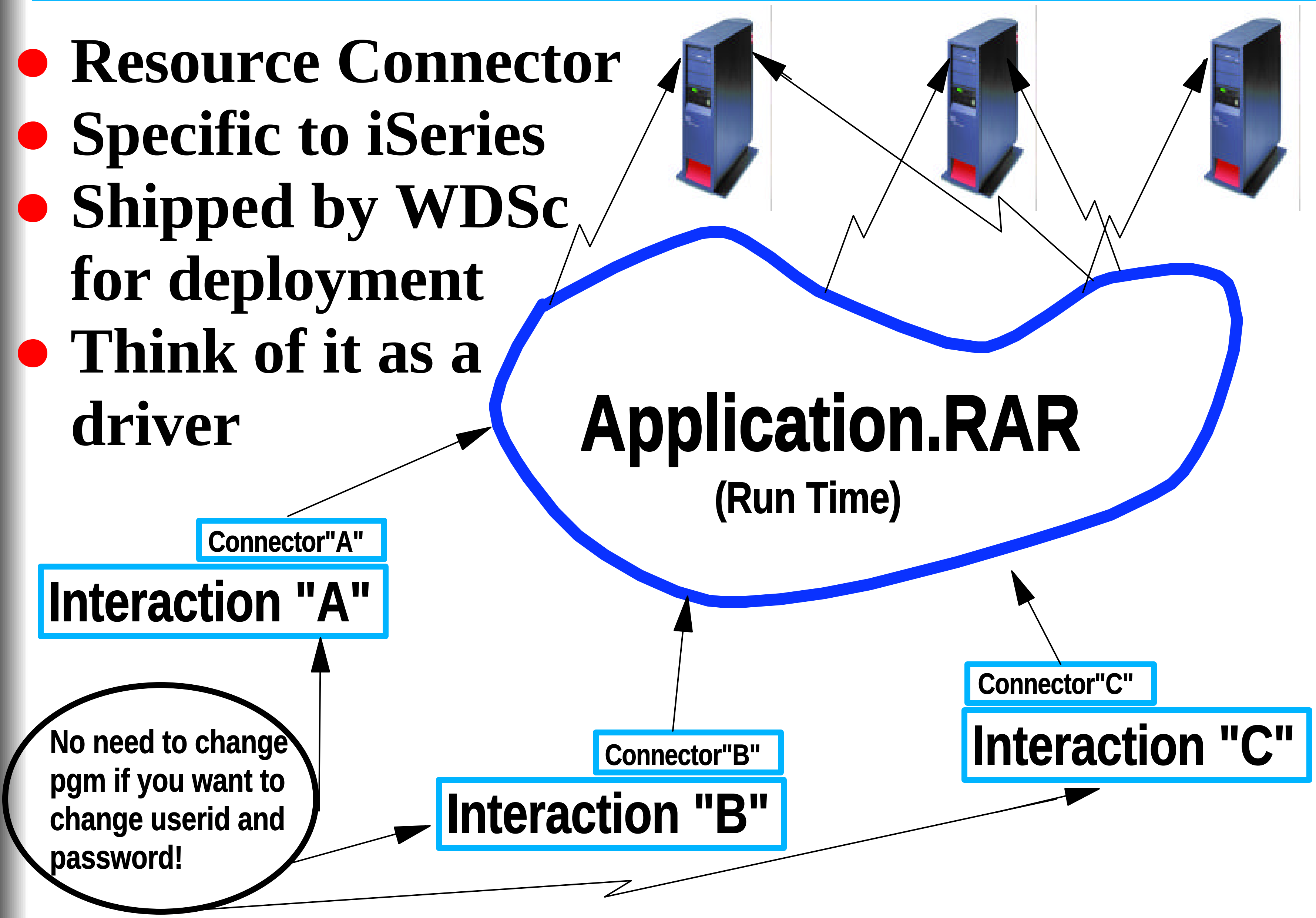
Tip: be sure to press Enter after typing library name!!

What is JCA connector?

Next page

What is JCA Connectors?

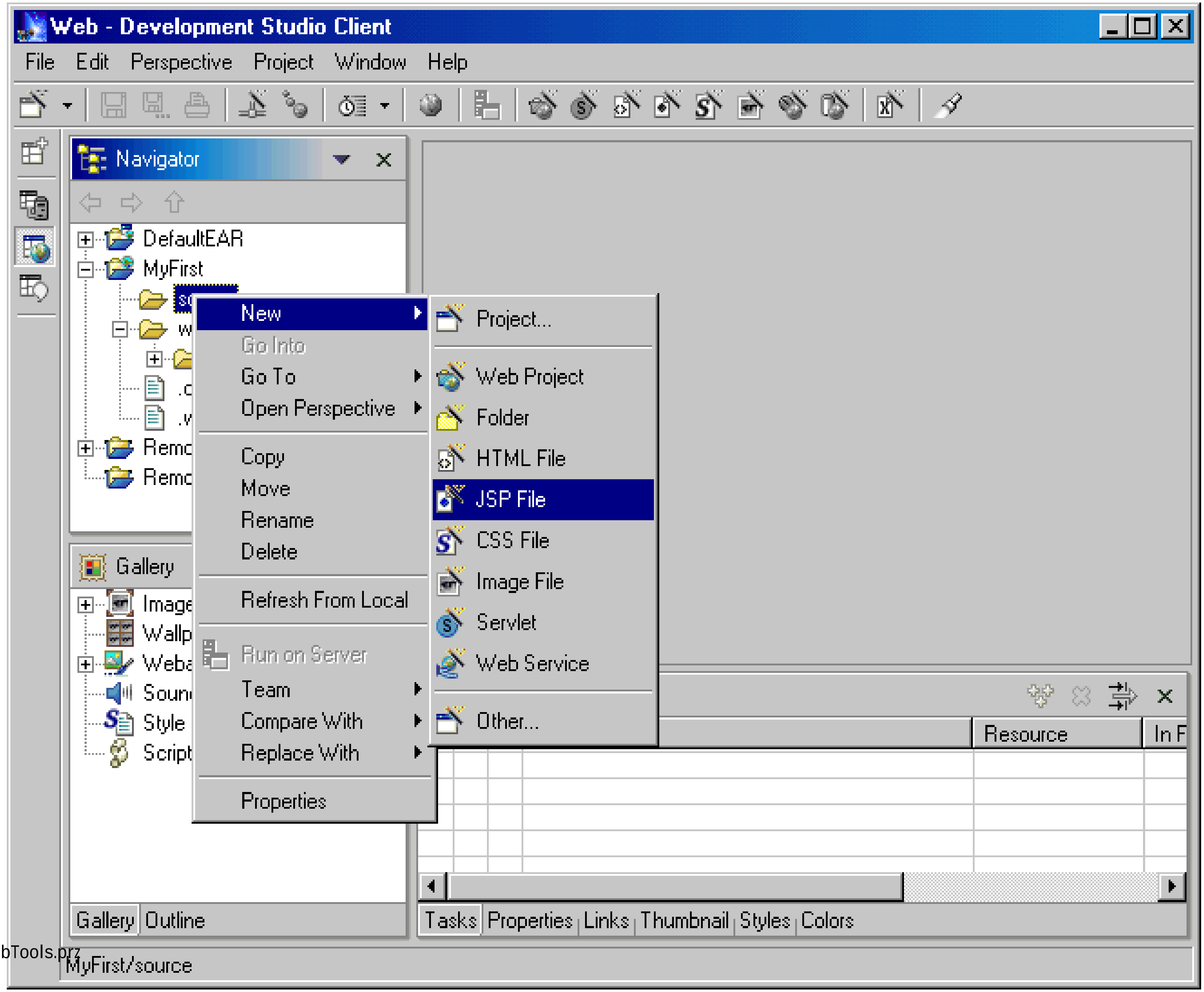
- Resource Connector
- Specific to iSeries
- Shipped by WDSc for deployment
- Think of it as a driver



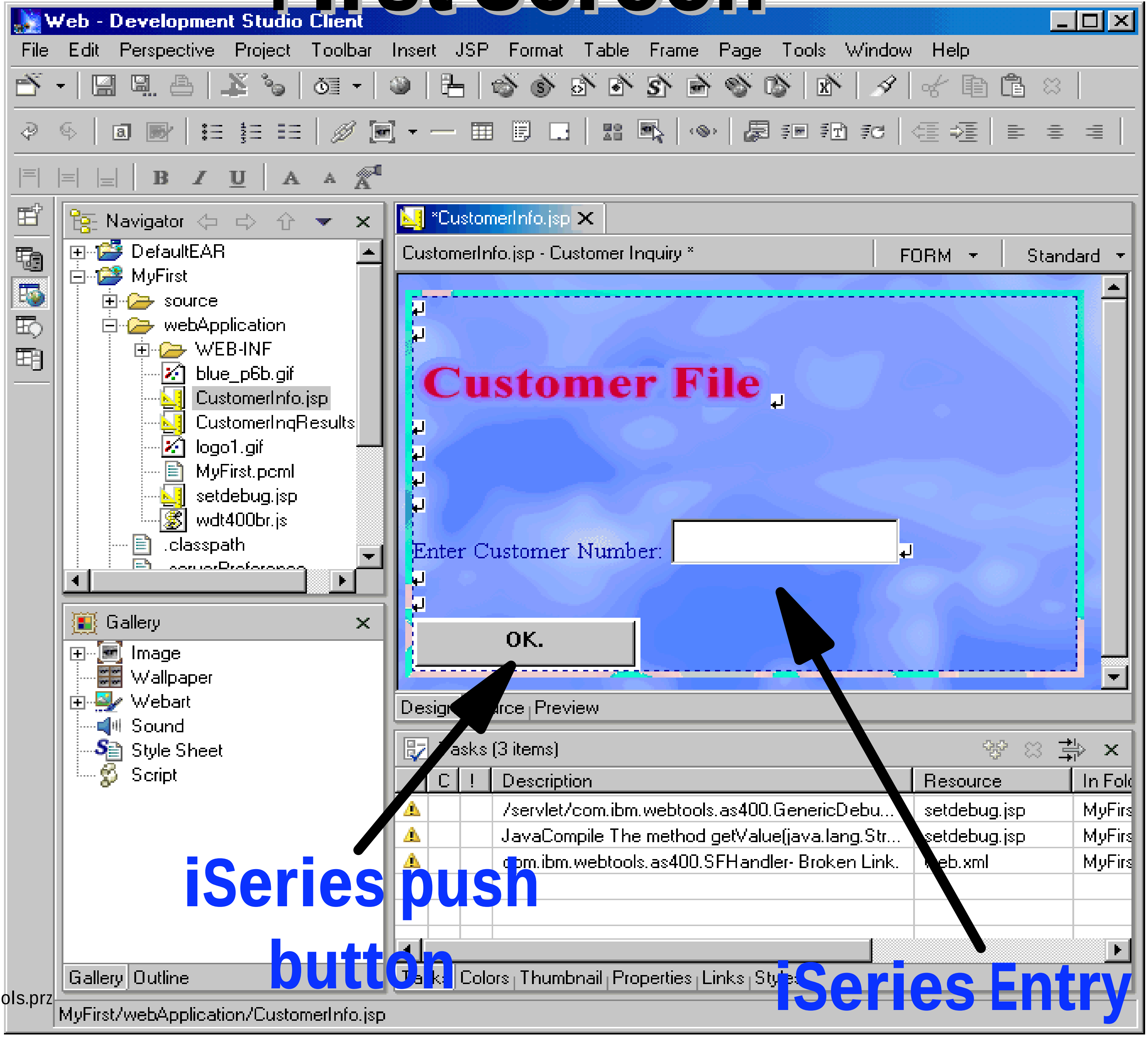
AGENDA

- What is in WDSc 5.0?
- e-business Primer
  - AD Model, traditional and web
- Web Application Primer
  - JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - RPG Example
  - iSeries Visual Co[illegible]l Tags (VTCs)
  - Web Interaction
  - Publishing
- Conclusion

Your first JSP . . .



First Screen

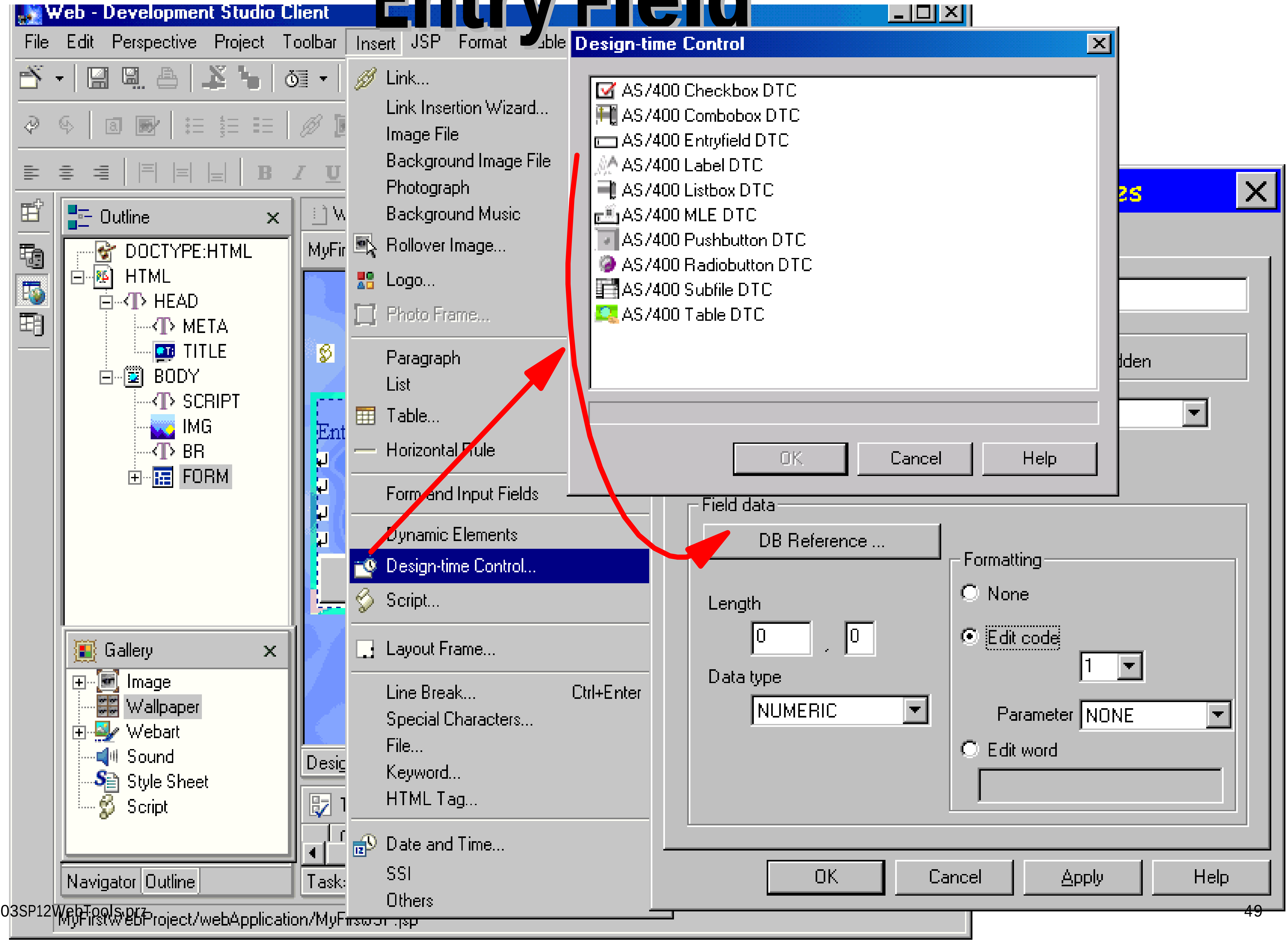


	C	!	Description	Resource	In Fol
⚠			/servlet/com.ibm.webtools.as400.GenericDebu...	setdebug.jsp	MyFirs
⚠			JavaCompile The method getValue(java.lang.Str...	setdebug.jsp	MyFirs
⚠			com.ibm.webtools.as400.SFHandler- Broken Link.	web.xml	MyFirs

iSeries push button

iSeries Entry

Entry Field

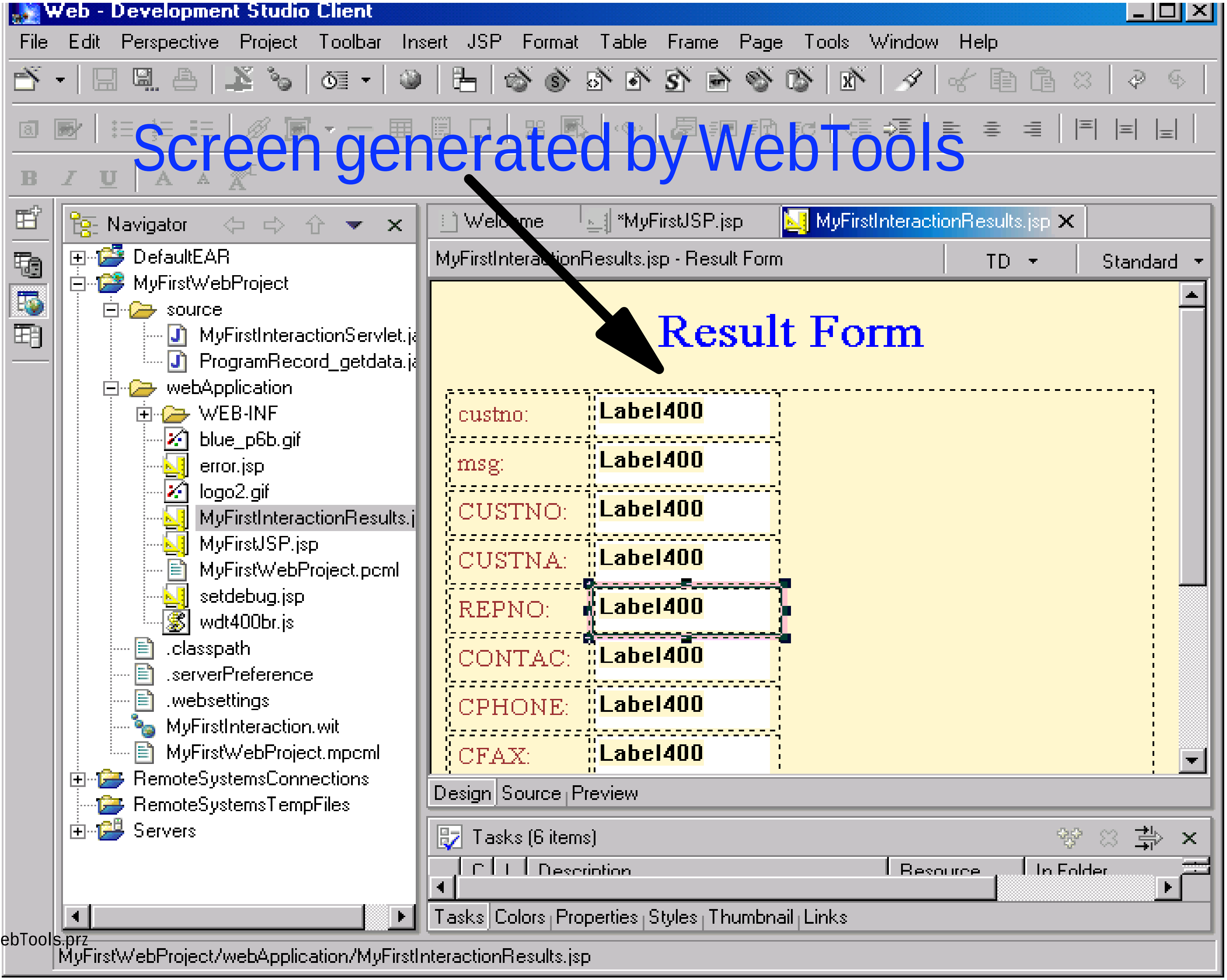


Length: 0 , 0

Data type: NUMERIC

Formatting: ○ None ● Edit code 1 Parameter NONE ○ Edit word

Customer Detail Screen



Screen generated by WebTools

Web - Development Studio Client

MyFirstInteractionResults.jsp - Result Form

Result Form

custno: Label400

msg: Label400

CUSTNO: Label400

CUSTNA: Label400

REPNO: Label400

CONTAC: Label400

CPHONE: Label400

CFAX: Label400

MyFirstWebProject/webApplication/MyFirstInteractionResults.jsp

RPG IV Program

NOTE Parameters to be passed

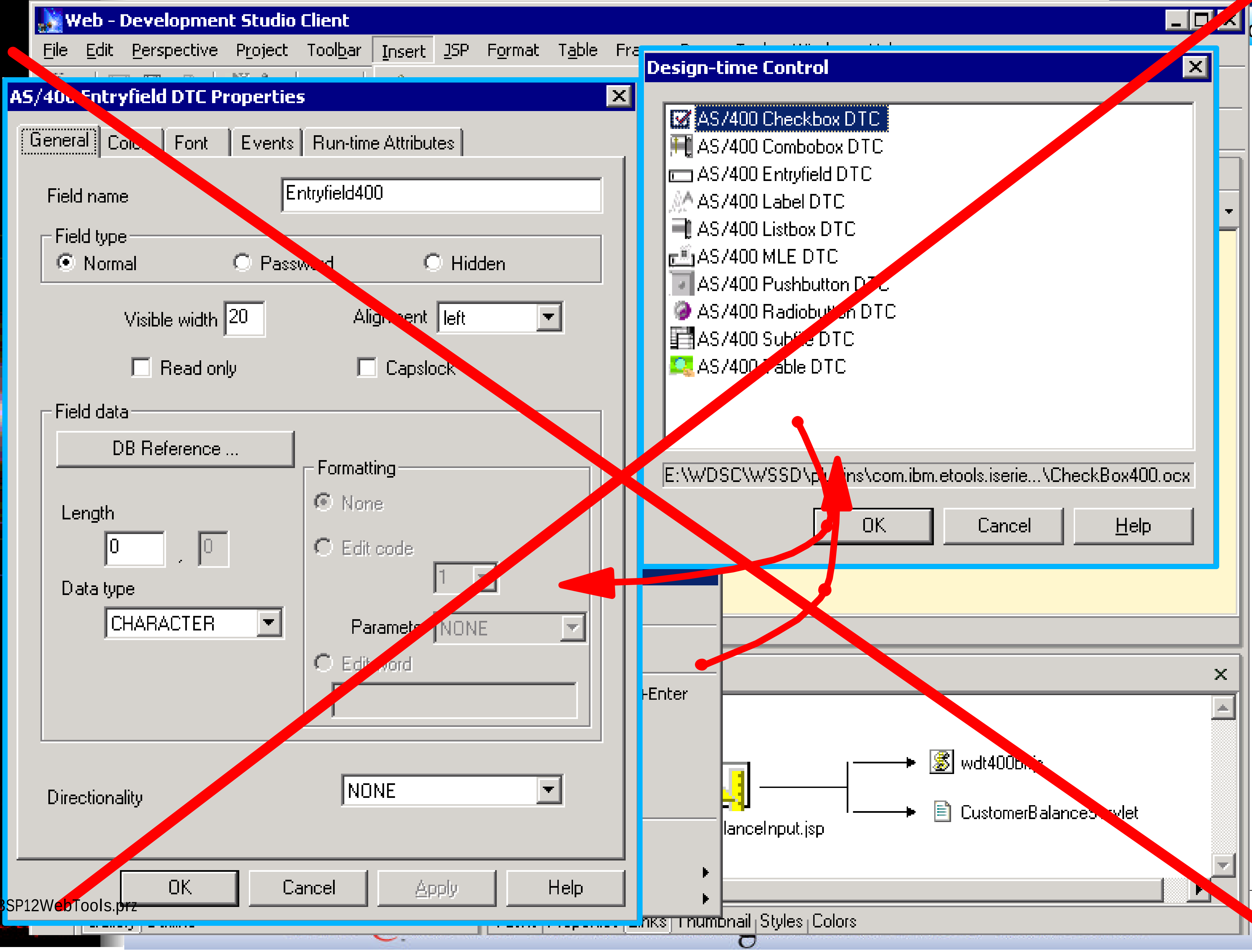
```
FCUSTOML3  IF   E           K Disk
DCustnoi          s                   like(CUSTNO)
D*
D CSTRUC        E DS                  extname(customl3:custom01)
d field2                        10
D return          s             20
D*------------------------------------------------------------------
c     *entry        plist
c                   parm                    custnoi
c                   parm                    cstruc
c                   parm                    return
c                   eval      return=*blank
c     custnoi       chain     customl3                          5050
c                   if        *in50
c                   eval      return='CUS0001 ' + CUSTNOI
c                   else
c                   eval      return='0'
c                   endif
c                   return
```

Files Used

```
A* Logical file description
A          R CUSTOM01                  PFILE(CUSTOMER)
A          K CUSTNO
```

```
A* Physical file description - CUSTOMER FILE
A      R CUSTOM01
A        CUSTNO         7            COLHDG('Customer number')
A        CUSTNA        40            COLHDG('Company name')
A        REPNO          5            COLHDG('Rep identifier')
A        CONTAC        30            COLHDG('Name')
A        CPHONE        17            COLHDG('Telephone')
A        CFAX          17            COLHDG('Fax')
A        CADDR         40            COLHDG('Address')
A        CCITY         30            COLHDG('City')
A        CCOUNT        20            COLHDG('Country')
A        CZIP          10            COLHDG('Postal Code')
A        CZIPLO         1            COLHDG('PC location')
A                                    VALUES('1' '2' '3')
A      K CUSTNO
```

No More DTCs ! ! !



Web - Development Studio Client

AS/400 Entryfield DTC Properties

Field name: Entryfield400

Field type: ● Normal ○ Password ○ Hidden

Visible width: 20 Alignment: left

□ Read only □ Capslock

Field data

DB Reference ...

Length: 0 , 0

Data type: CHARACTER

Formatting: ● None ○ Edit code 1

Parameter: NONE

○ Edit word

Directionality: NONE

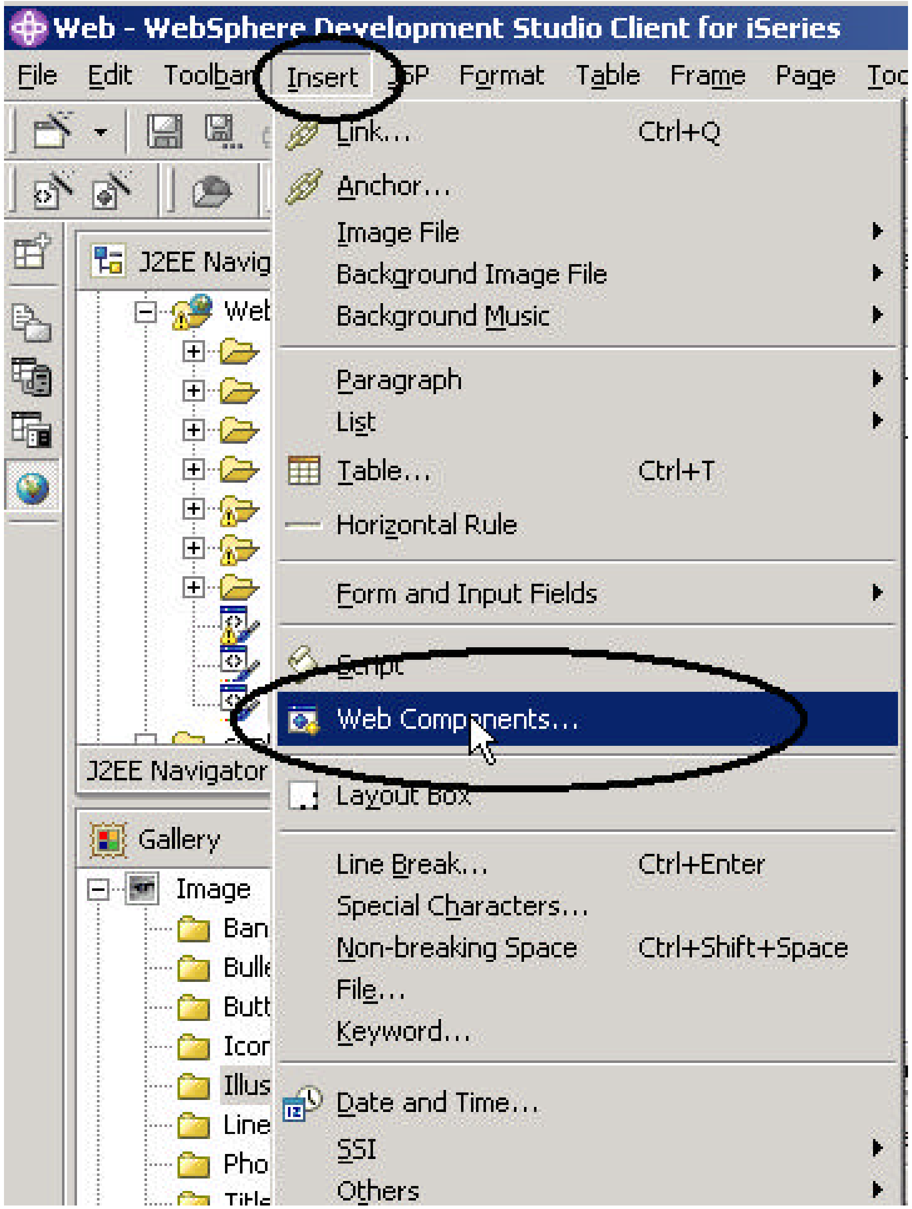
Design-time Control

- AS/400 Checkbox DTC
- AS/400 Combobox DTC
- AS/400 Entryfield DTC
- AS/400 Label DTC
- AS/400 Listbox DTC
- AS/400 MLE DTC
- AS/400 Pushbutton DTC
- AS/400 Radiobutton DTC
- AS/400 Subfile DTC
- AS/400 Table DTC

E:\WDSC\WSSD\plugins\com.ibm.etools.iserie...\CheckBox400.ocx

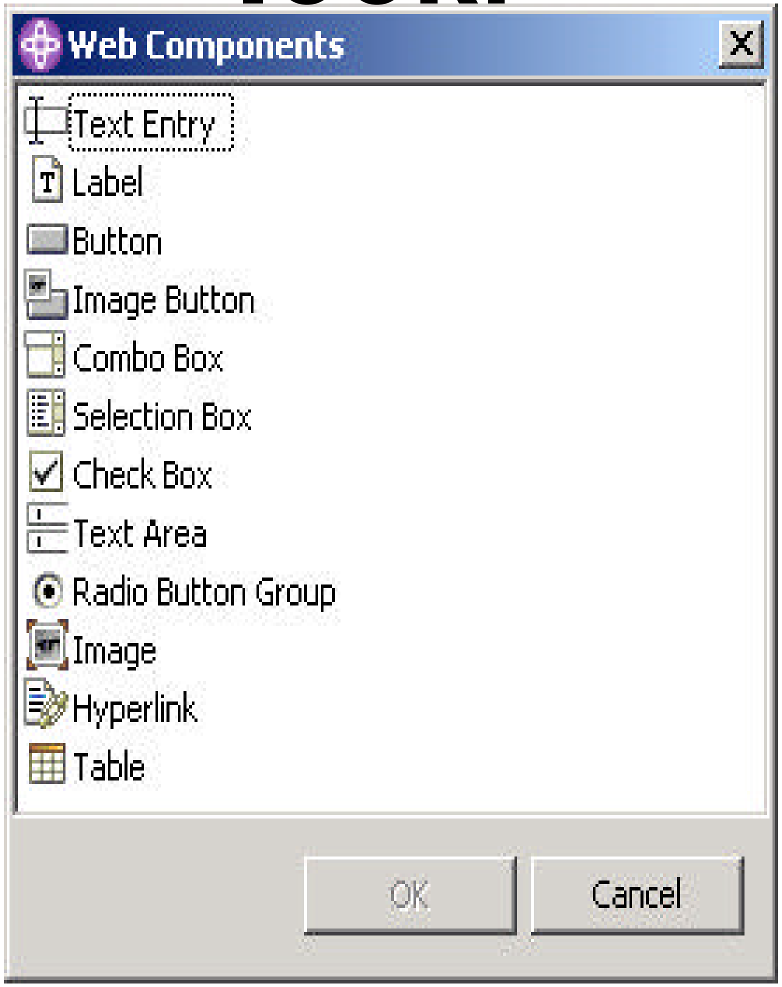
VTCs ... No more DTCs

- VTC: Visual Control Tags
  - Emerging technology
  - Blessed by IBM and SUN
- Tag library defined by:
  - www.sun.com
- Extendable
- DTCs: Old fashion
  - ActiveX technology



VTCs ... No more DTCs

Same selection . . . Better look!



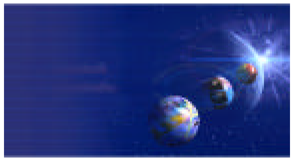
Web Components

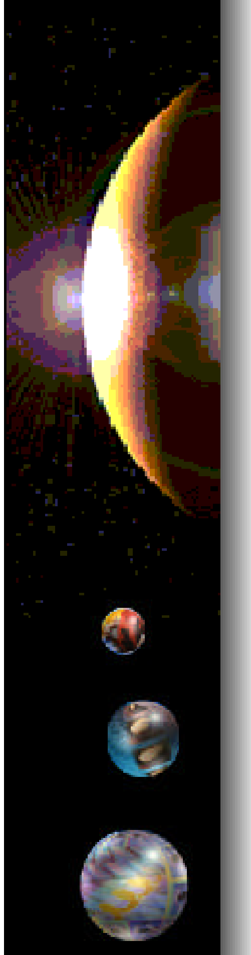
- Text Entry
- Label
- Button
- Image Button
- Combo Box
- Selection Box
- Check Box
- Text Area
- Radio Button Group
- Image
- Hyperlink
- Table

AGENDA

- What is in WDSc 5.0?
- e-business Primer
  - ▶ AD Model, traditional and web
- Web Application Primer
  - ▶ JSPs, Servlets, forms, etc
- Mixing Java and RPG
- What is Web Tool for iSeries
- Introducing WDSc for iSeries
  - ▶ RPG Example
  - ▶ iSeries Visual Control Tags (VTCs)
  - ▶ Web Interaction
  - ▶ Publishing
- Conclusion

Web Interaction Wizard

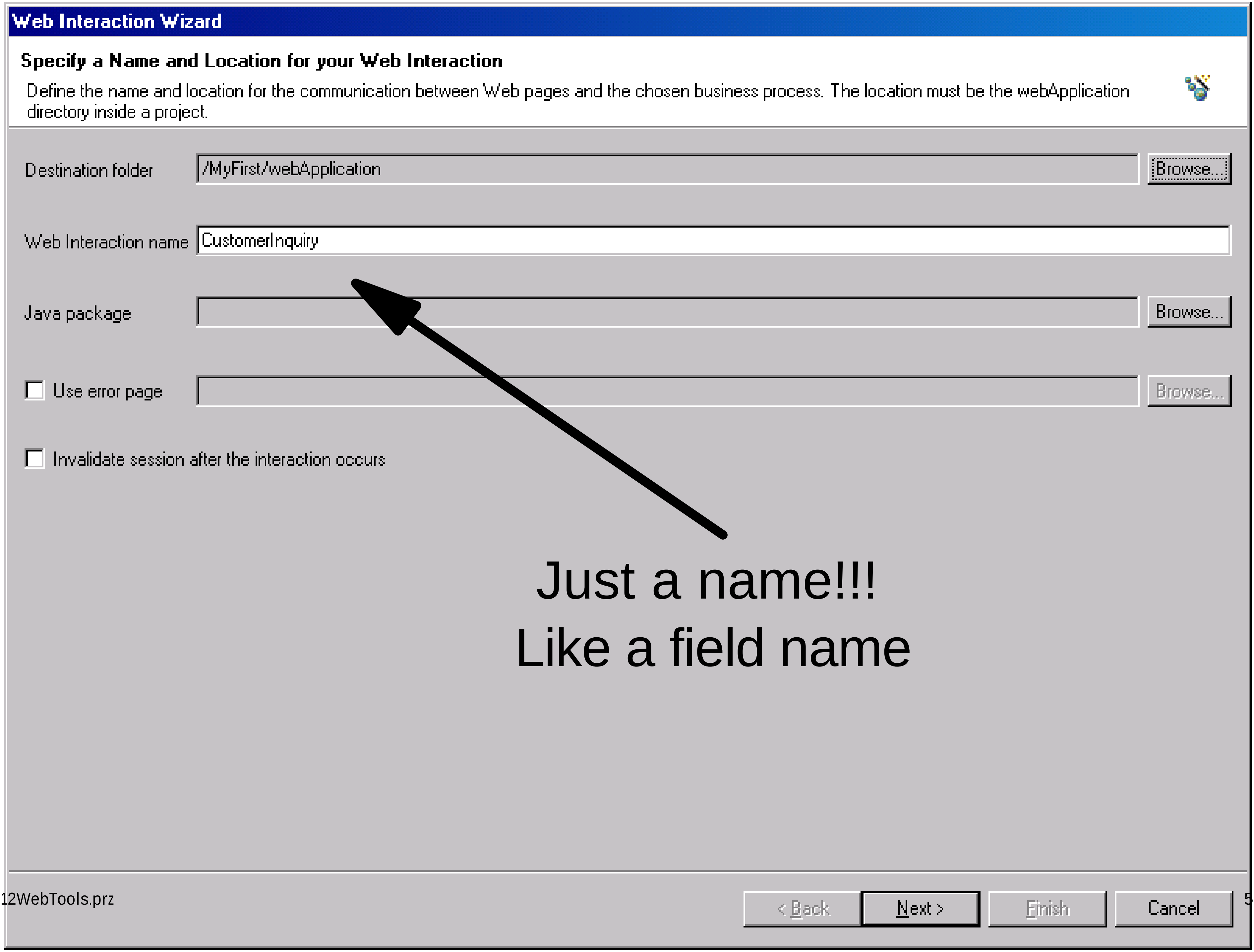




- Two modes to interaction wizard:

1. **Generate input/output Web pages**
   - **Given the parameter description of the API to call**
2. **Generate mappings**
   - **Given the input/output pages**
   - **Given the parameter description of the API to call**
   - **Given the mappings**
     - ✓ between input parms & input fields
     - ✓ between ouput parms & output fields

Interaction Wizard



Specify a Name and Location for your Web Interaction

Define the name and location for the communication between Web pages and the chosen business process. The location must be the webApplication directory inside a project.

Destination folder: /MyFirst/webApplication

Web Interaction name: CustomerInquiry

Java package:

□ Use error page

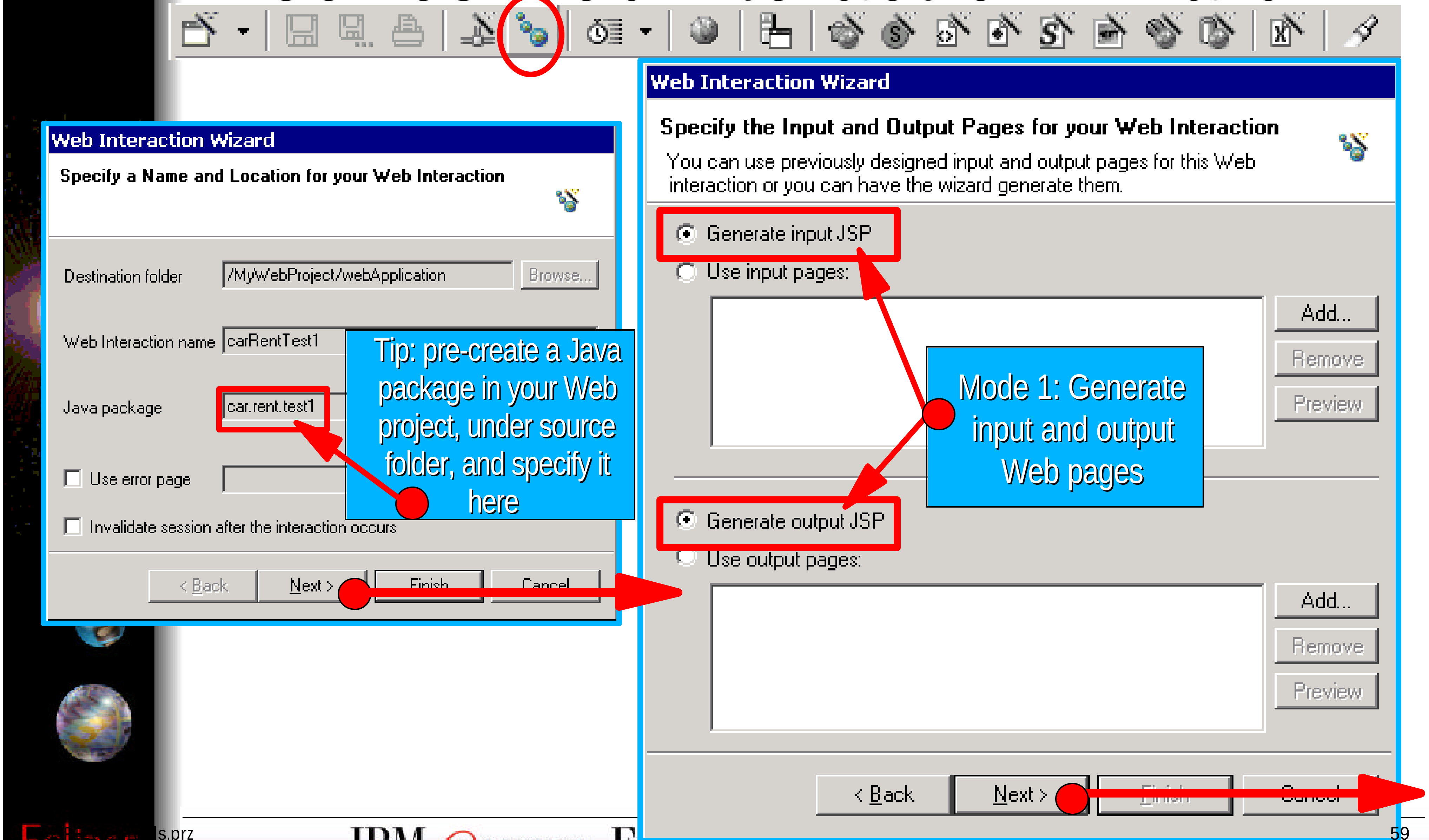
□ Invalidate session after the interaction occurs

Just a name!!!
Like a field name

WDSc: iSeries Web Tools

iSeries AD, IBM Toronto

► iSeries Web Interaction Wizard



Web Interaction Wizard

Specify a Name and Location for your Web Interaction

Destination folder: /MyWebProject/webApplication [Browse...]

Web Interaction name: carRentTest1

Java package: car.rent.test1

□ Use error page

□ Invalidate session after the interaction occurs

Tip: pre-create a Java package in your Web project, under source folder, and specify it here

Web Interaction Wizard

Specify the Input and Output Pages for your Web Interaction

You can use previously designed input and output pages for this Web interaction or you can have the wizard generate them.

● Generate input JSP

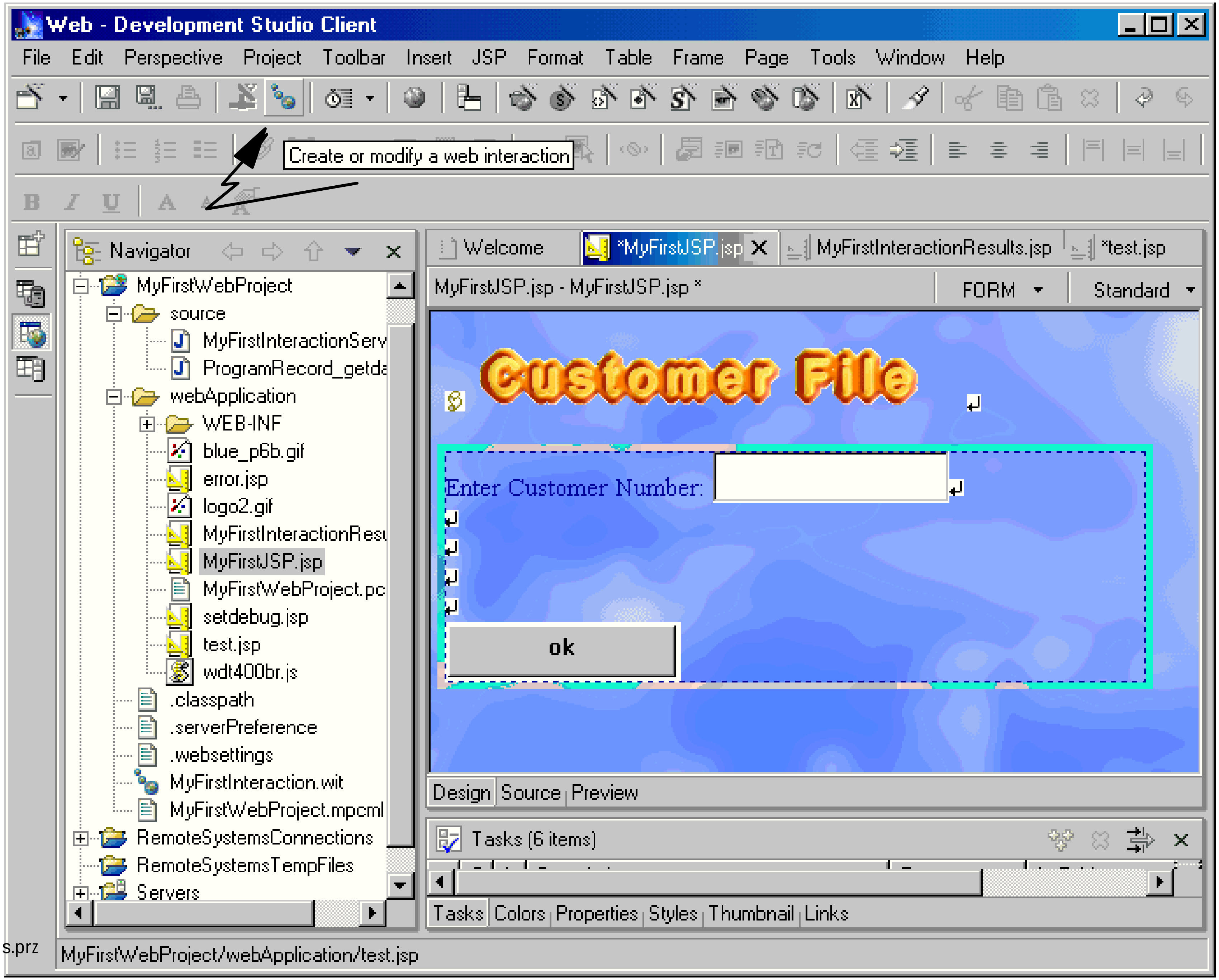
○ Use input pages:

● Generate output JSP

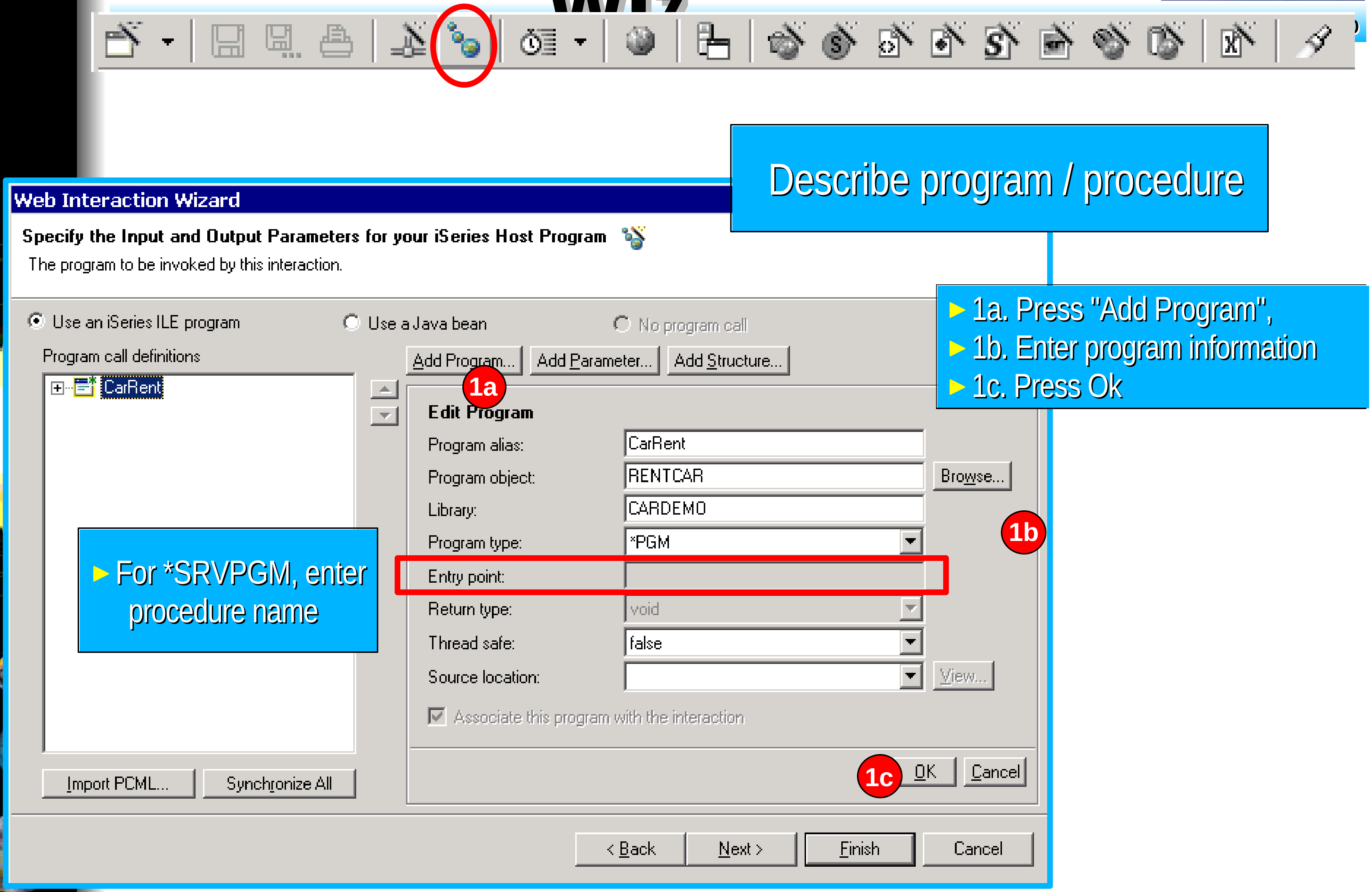
○ Use output pages:

Mode 1: Generate input and output Web pages

Or ... Create your own!



iSeries Web Interaction Wiz



Describe program / procedure

- 1a. Press "Add Program",
- 1b. Enter program information
- 1c. Press Ok

- For *SRVPGM, enter procedure name

Web Interaction Wizard

Specify the Input and Output Parameters for your iSeries Host Program

The program to be invoked by this interaction.

● Use an iSeries ILE program ○ Use a Java bean ○ No program call

Program call definitions

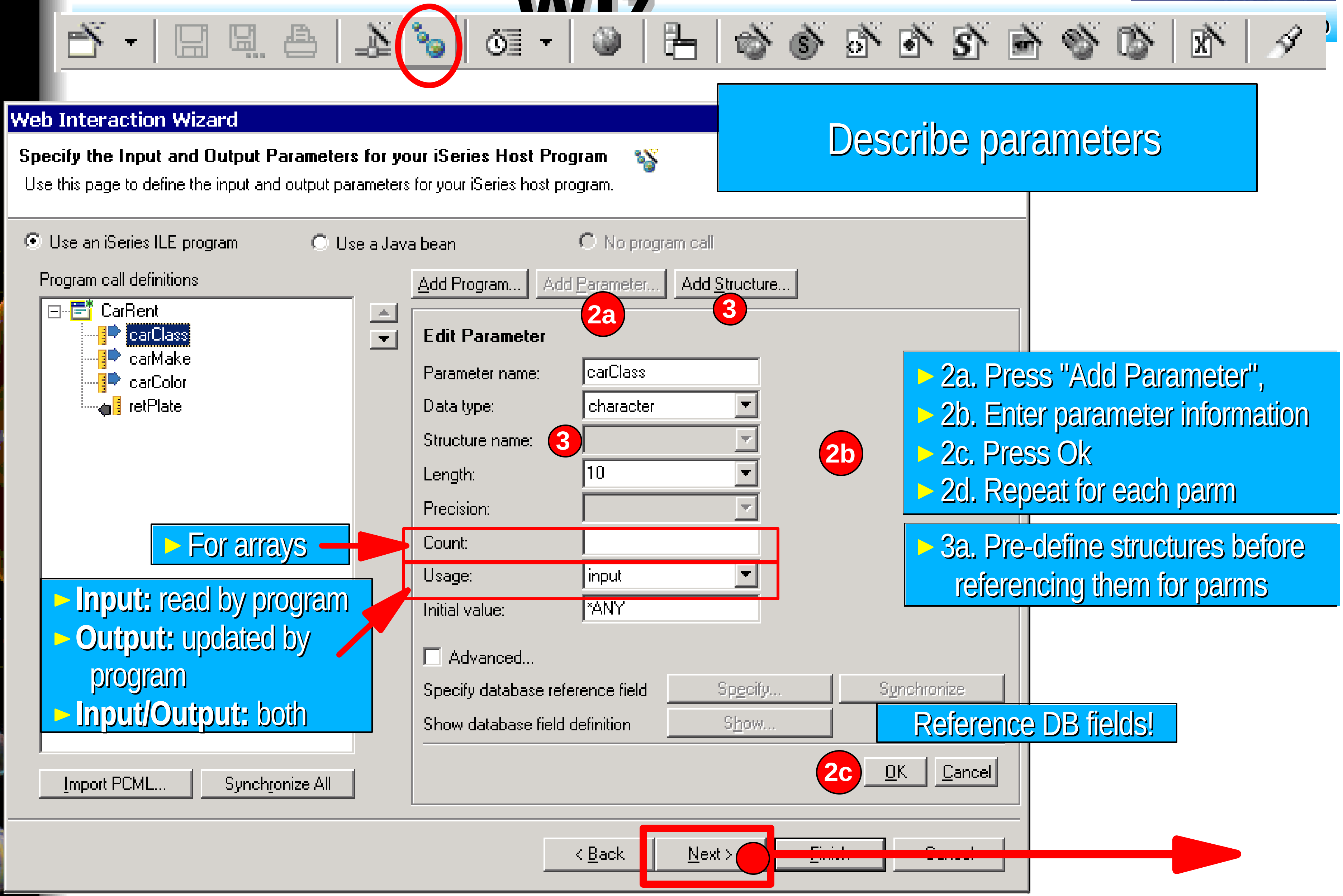
CarRent

Edit Program

Program alias:	CarRent
Program object:	RENTCAR
Library:	CARDEMO
Program type:	*PGM
Entry point:	
Return type:	void
Thread safe:	false
Source location:	

☑ Associate this program with the interaction

iSeries Web Interaction Wiz



Web Interaction Wizard

Specify the Input and Output Parameters for your iSeries Host Program

Use this page to define the input and output parameters for your iSeries host program.

● Use an iSeries ILE program ○ Use a Java bean ○ No program call

Program call definitions

- CarRent
  - carClass
  - carMake
  - carColor
  - retPlate

Edit Parameter

Parameter name:	carClass
Data type:	character
Structure name:	
Length:	10
Precision:	
Count:	
Usage:	input
Initial value:	*ANY

□ Advanced...

Specify database reference field

Show database field definition

Describe parameters

- 2a. Press "Add Parameter",
- 2b. Enter parameter information
- 2c. Press Ok
- 2d. Repeat for each parm

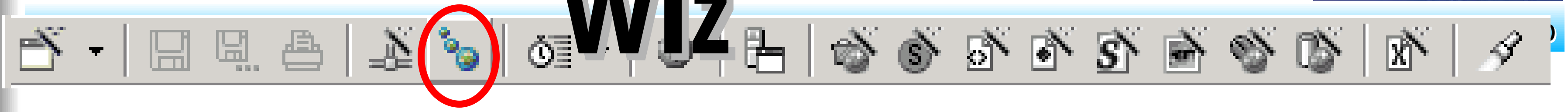
- 3a. Pre-define structures before referencing them for parms

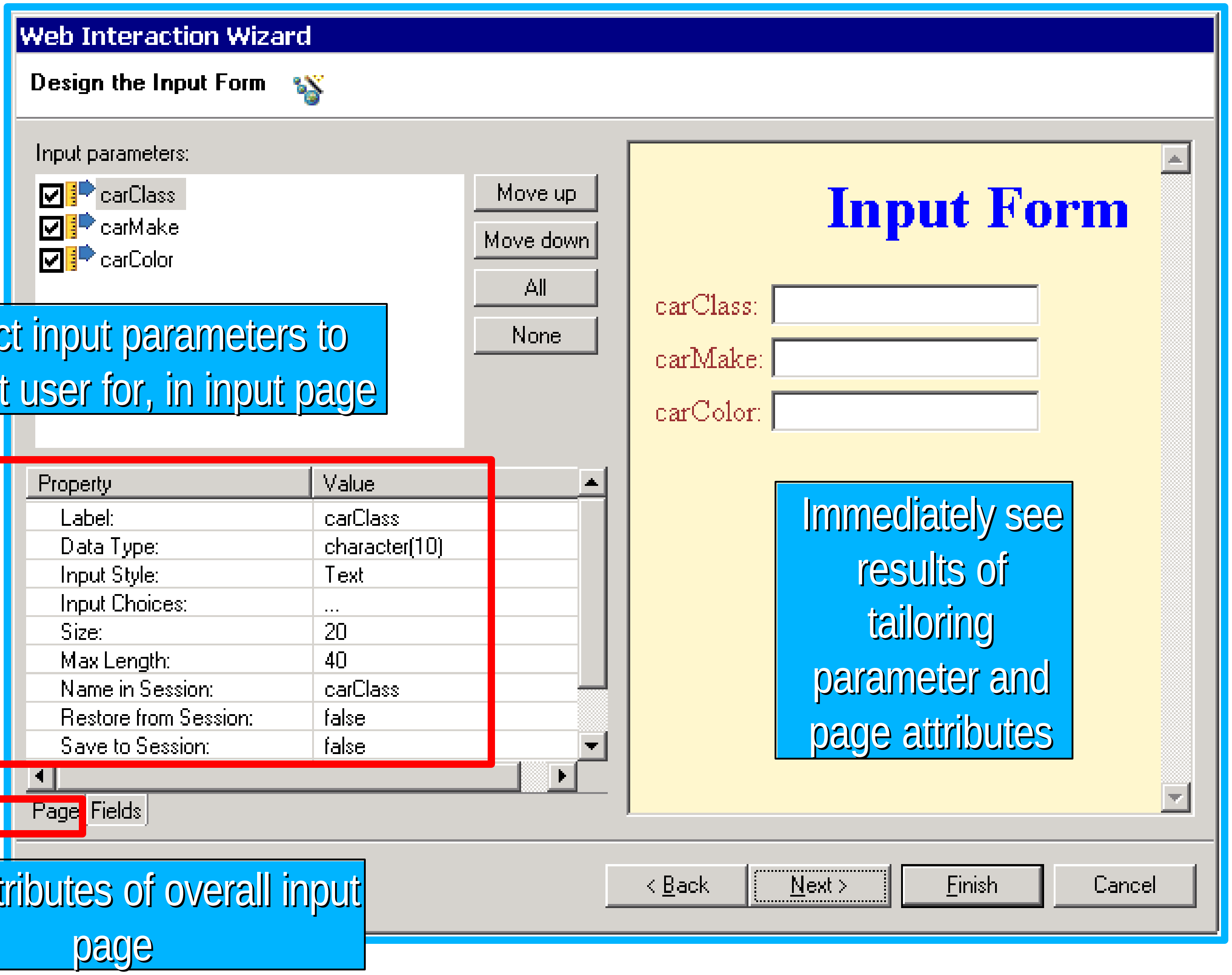
- For arrays

- **Input:** read by program
- **Output:** updated by program
- **Input/Output:** both

Reference DB fields!

iSeries Web Interaction Wiz




Web Interaction Wizard

Design the Input Form

Input parameters:

- ☑ carClass
- ☑ carMake
- ☑ carColor

Property	Value
Label:	carClass
Data Type:	character(10)
Input Style:	Text
Input Choices:	...
Size:	20
Max Length:	40
Name in Session:	carClass
Restore from Session:	false
Save to Session:	false

Input Form

carClass:

carMake:

carColor:

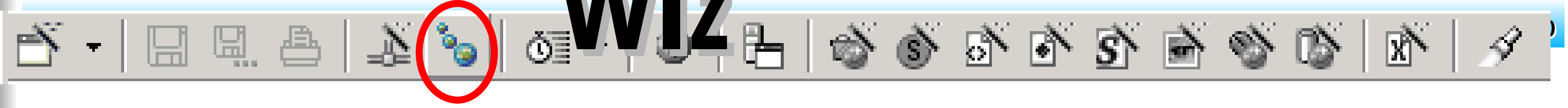
Select input parameters to prompt user for, in input page

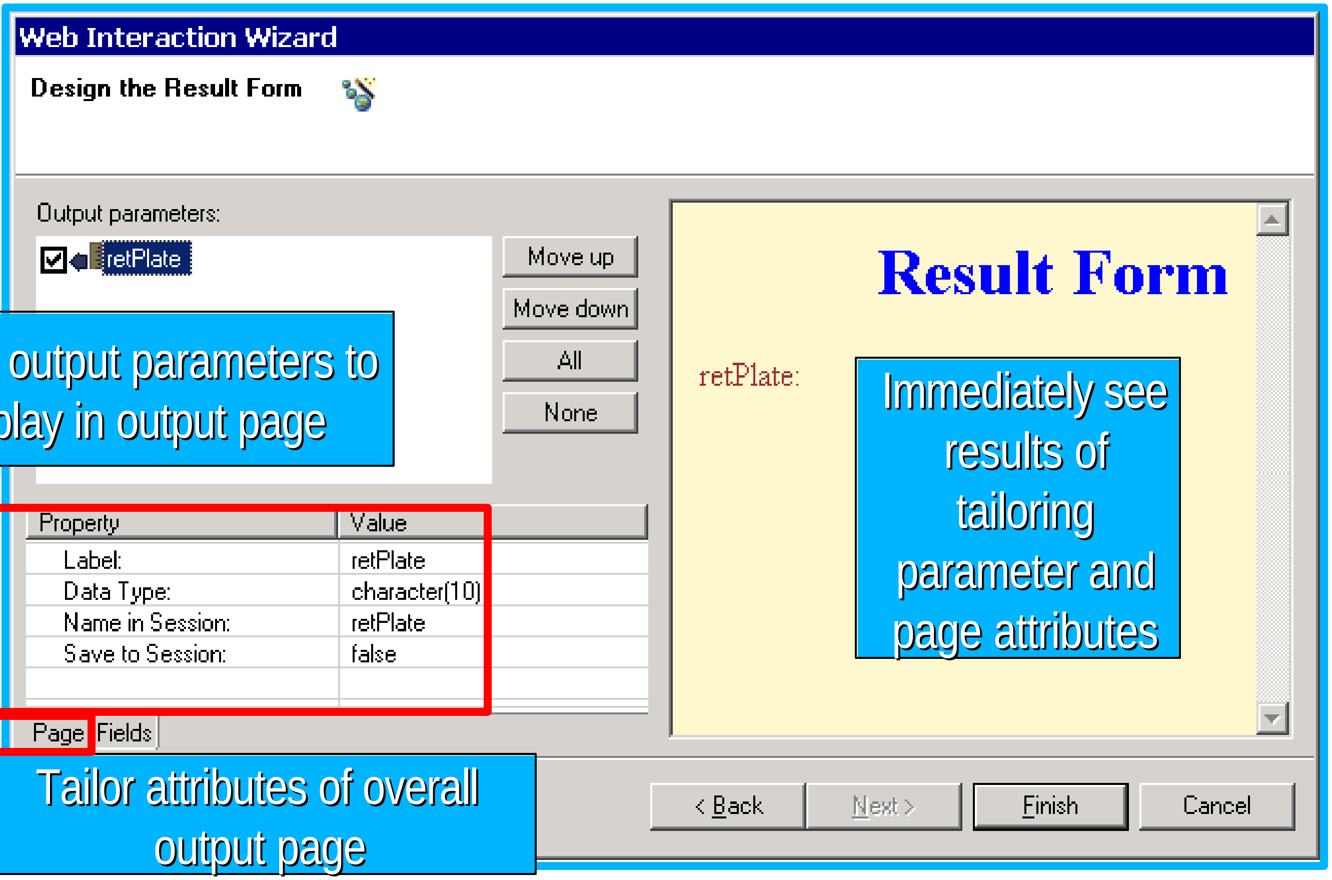
Tailor attributes of generated prompt per parameter

Immediately see results of tailoring parameter and page attributes

Tailor attributes of overall input page

iSeries Web Interaction Wiz





Web Interaction Wizard

Design the Result Form

Output parameters:

☑ retPlate

Property	Value
Label:	retPlate
Data Type:	character(10)
Name in Session:	retPlate
Save to Session:	false

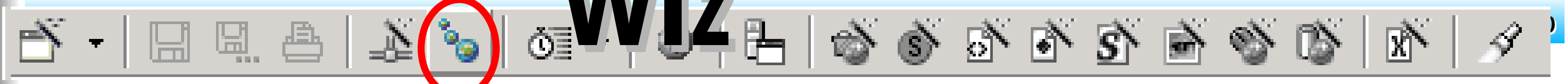
Select output parameters to display in output page

Tailor attributes of generated prompt per parameter

Tailor attributes of overall output page

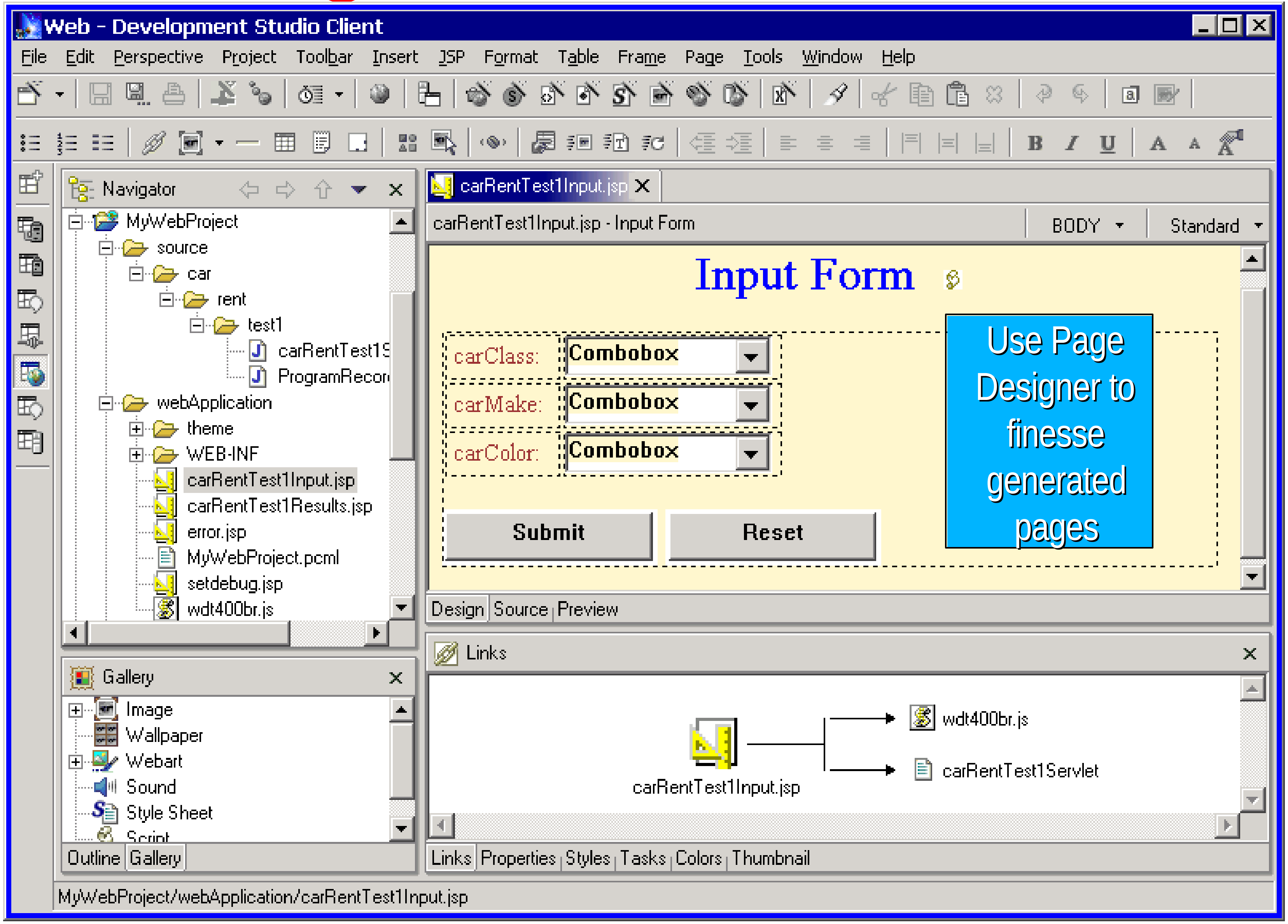
Immediately see results of tailoring parameter and page attributes

iSeries Web Interaction Wiz



Generated files

Use "Run on Server" to test



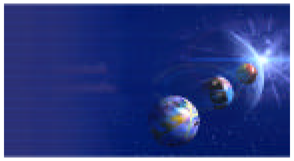
MyWebProject/webApplication/carRentTest1Input.jsp

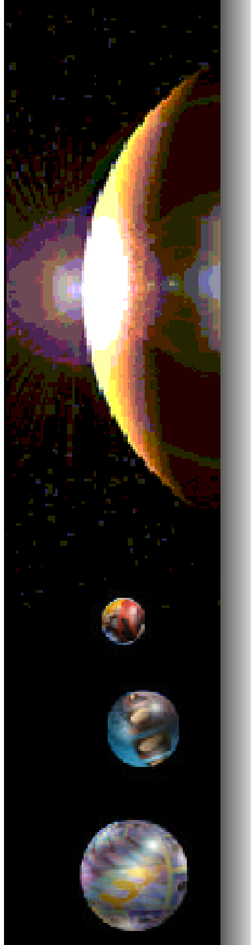
Use Page Designer to finesse generated pages

You are done! What's Next?

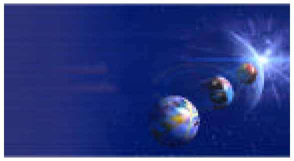
- **What did you do so far?**
  - You used the designer to create the input page
  - The output page was generated for you
  - Created your Files on the iSeries
  - Created your RPG Program
  - Created an 'iSeries Web Interaction' Interaction to link your program parameters to the input page and output page
- **Next?**
  - Run locally to test your application Or

WDSc: Run On Server

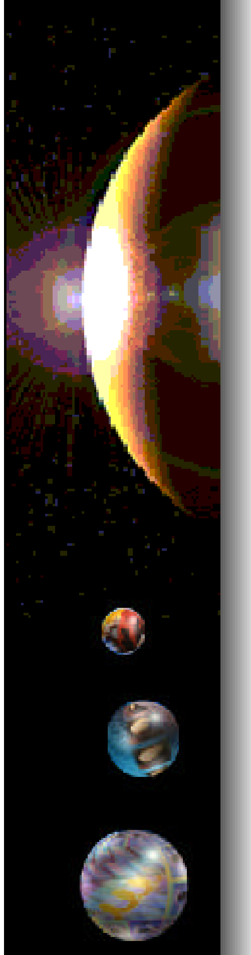




- Run On Server
  - **Now this is VERY COOL!**
- When ready to test your Web app
  - **Right click on initial html or jsp file**
    - or whole project, which implies the index.html file
  - **Select "Run on Project"**
  - **Wait for the magic...**
- Your Web application will run!
  - **Opens Server perspective**
    - Publishes it to built-in copy of WAS
    - Starts built-in copy of WAS
    - Brings up a Web Browser
    - **Runs your application!!**
      - ✓ Tip: you can set breakpoints in your Java code!



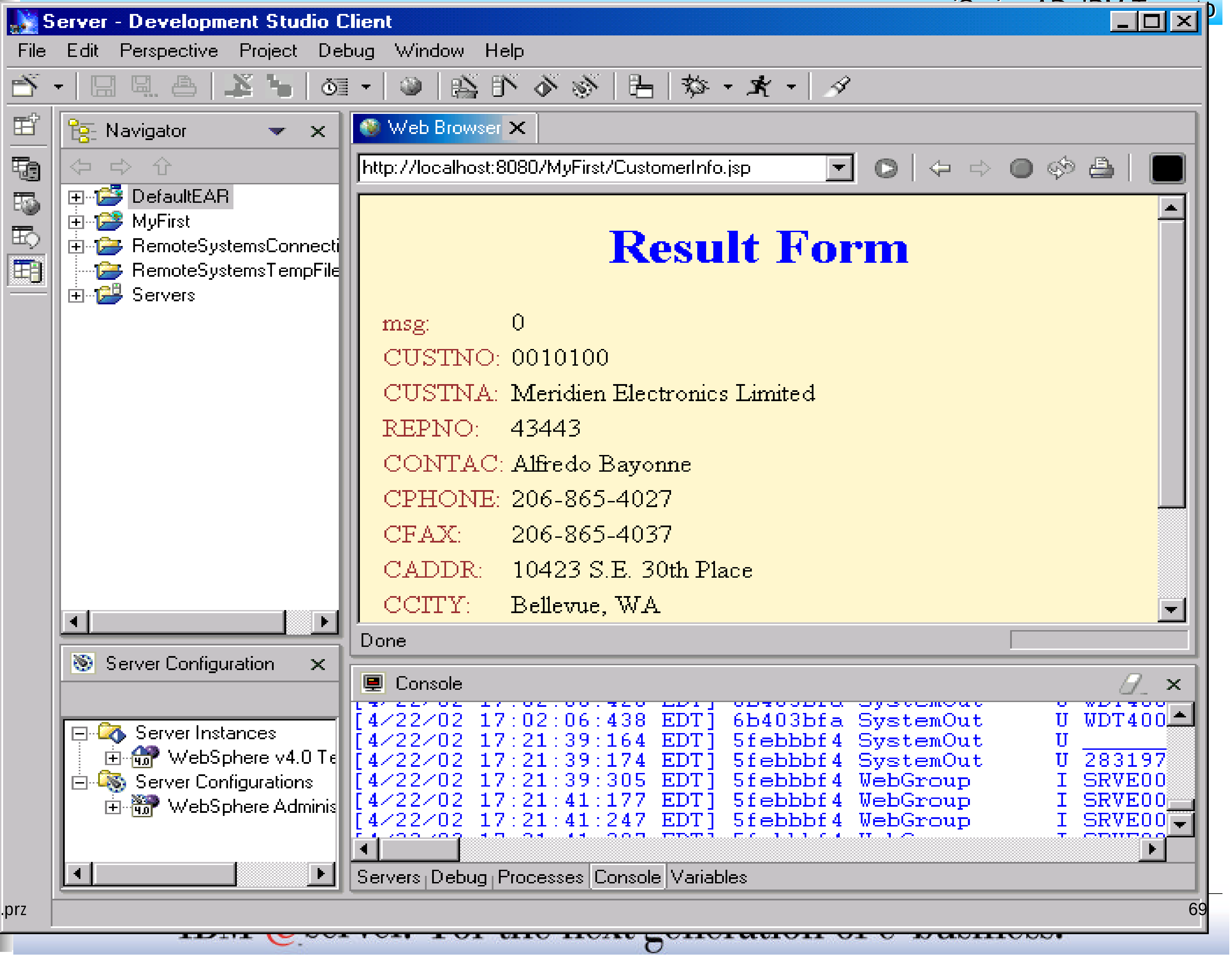
WDSc: WAS Test Env



► WebSphere Test Environment

- **A full copy of WAS 4.0 Single Server Edition is embedded in the IDE**
  - ► Integrated with Server Tools to enable instant and dead-easy testing of Web projects within WDSc.
    - ✓ Standalone all-in-one testing
    - ✓ No dependency on WAS installation or availability
    - ✓ No dependency on an external database
  - ► Provides the ability to debug live server-side code
  - ► Supports configuring multiple Web applications
  - ► Supports multiple servers that can be configured and run at the same time
  - ► Provides access to the profiling feature that is available in the workbench
  - ► Provides the ability to version Server Tools server configurations
  - ► Provides access to the WAS Administration Client

Run it!



http://localhost:8080/MyFirst/CustomerInfo.jsp

Result Form

msg:	0
CUSTNO:	0010100
CUSTNA:	Meridien Electronics Limited
REPNO:	43443
CONTAC:	Alfredo Bayonne
CPHONE:	206-865-4027
CFAX:	206-865-4037
CADDR:	10423 S.E. 30th Place
CCITY:	Bellevue, WA

Done

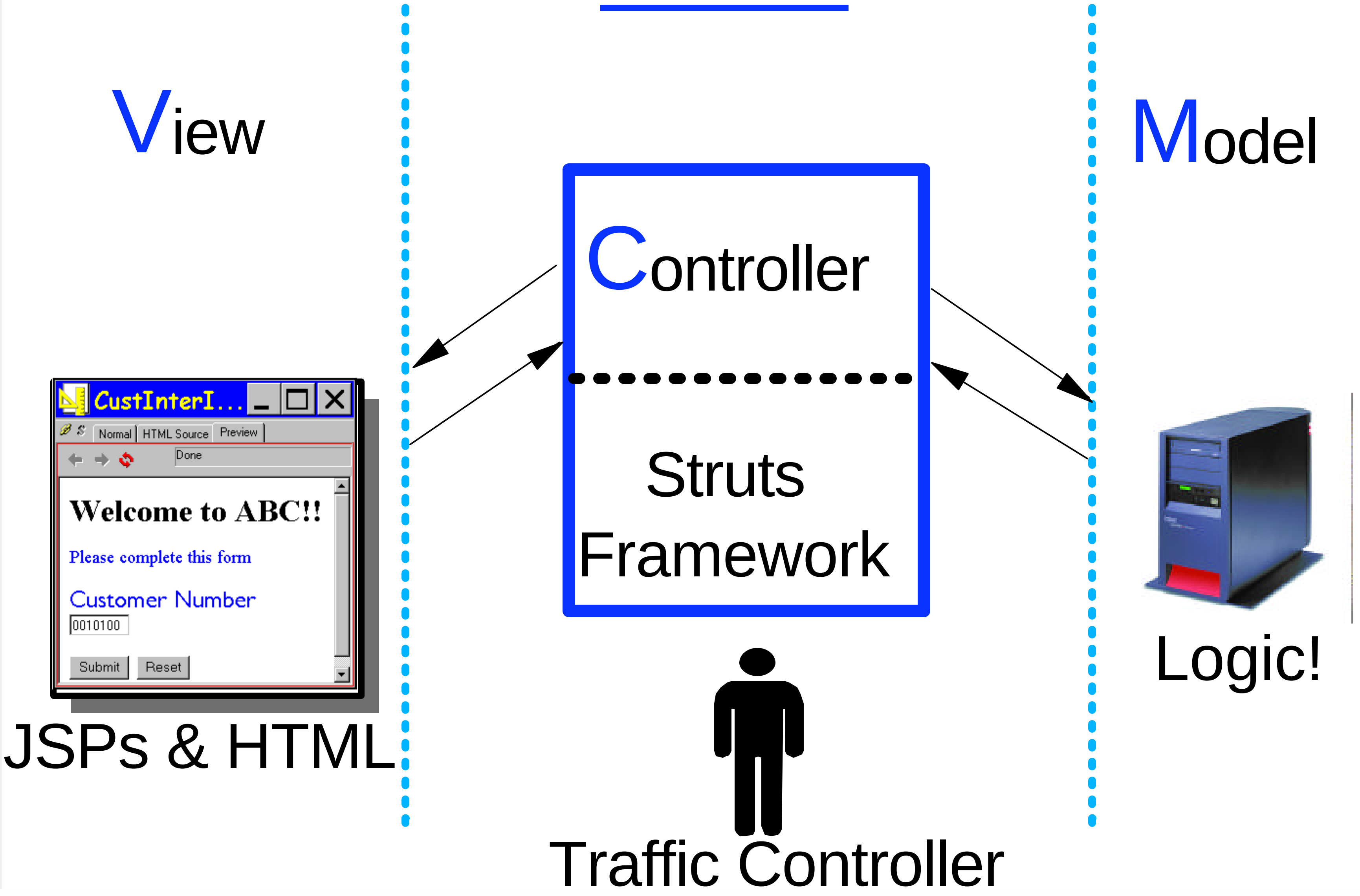
```
[4/22/02 17:02:06:438 EDT] 6b403bfa SystemOut     U WDT400
[4/22/02 17:21:39:164 EDT] 5febbbf4 SystemOut     U
[4/22/02 17:21:39:174 EDT] 5febbbf4 SystemOut     U 283197
[4/22/02 17:21:39:305 EDT] 5febbbf4 WebGroup      I SRVE00
[4/22/02 17:21:41:177 EDT] 5febbbf4 WebGroup      I SRVE00
[4/22/02 17:21:41:247 EDT] 5febbbf4 WebGroup      I SRVE00
```

AGENDA

- Web Interation - Struts based!

Web Wizard Struts based ...

M.V.C

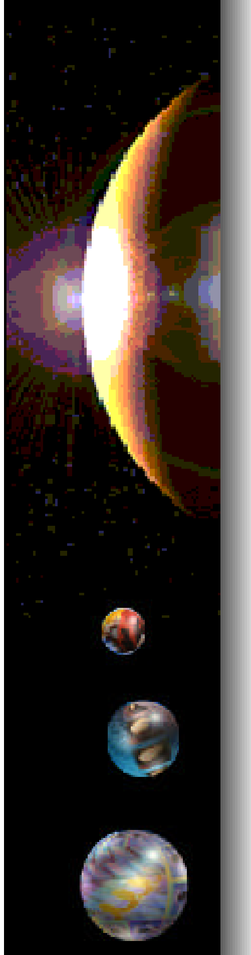


Welcome to ABC!!

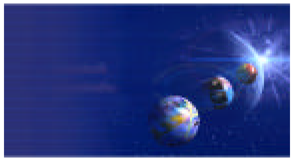
Please complete this form

Customer Number

0010100



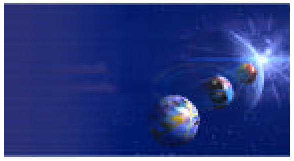
SUMMARY

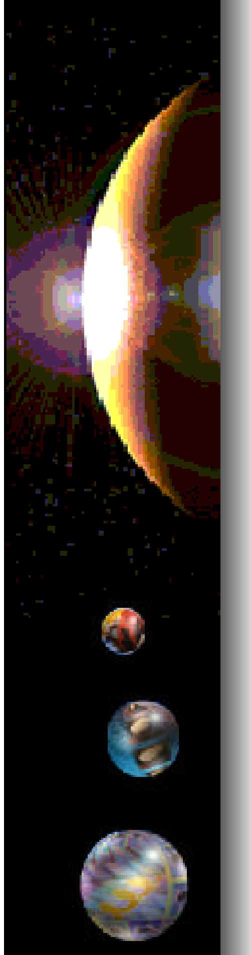


► iSeries Web Tools, at a glance

- **Tools optimized for iSeries developers!**
  - ► **Web Interaction wizard**
    - ✓ You define the parameters to a *PGM/*SRVPGM, wizard generates input JSP prompting for input parm, output JSP showing output parms, and all the glue in-betwee
    - ✓ Or you pre-create the input and/or output pages, and map the input/output fields on the pages to the input/output parameters in the *PGM/*SRVPGM, and it generates the glue to bind them
  - ► **Host Information wizard**
    - ✓ Set runtime information such as library list and sign-on information, to be used by glue generated by all Web Interaction wizards for this Web project
  - ► **Design Time Controls (Web GUI Widgets)**
    - ✓ Web GUI Widgets that support DSPF-like attributes such as error checking by datatype, and edit-code and edit-word

SUMMARY



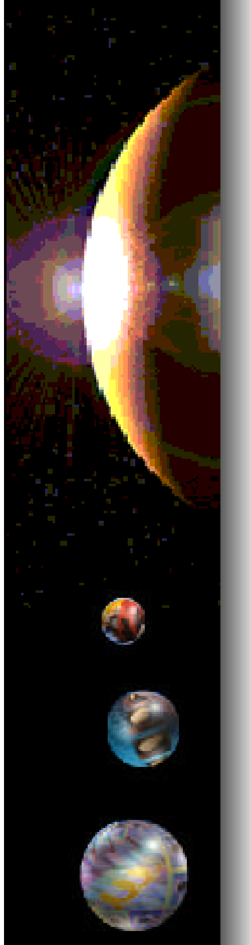


► iSeries Web Tools, at a glance

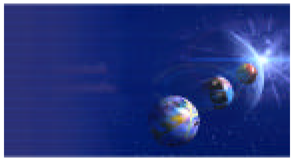
- **Tools optimized for iSeries developers!**
  - ► **Web Interaction wizard**
    - ✓ You define the parameters to a *PGM/*SRVPGM, wizard generates input JSP prompting for input parm, output JSP showing output parms, and all the glue in-betwee
    - ✓ Or you pre-create the input and/or output pages, and map the input/output fields on the pages to the input/output parameters in the *PGM/*SRVPGM, and it generates the glue to bind them
  - ► **Host Information wizard**
    - ✓ Set runtime information such as library list and sign-on information, to be used by glue generated by all Web Interaction wizards for this Web project
  - ► **Design Time Controls (Web GUI Widgets)**
    - ✓ Web GUI Widgets that support DSPF-like attributes such as error checking by datatype, and edit-code and edit-word

Standards

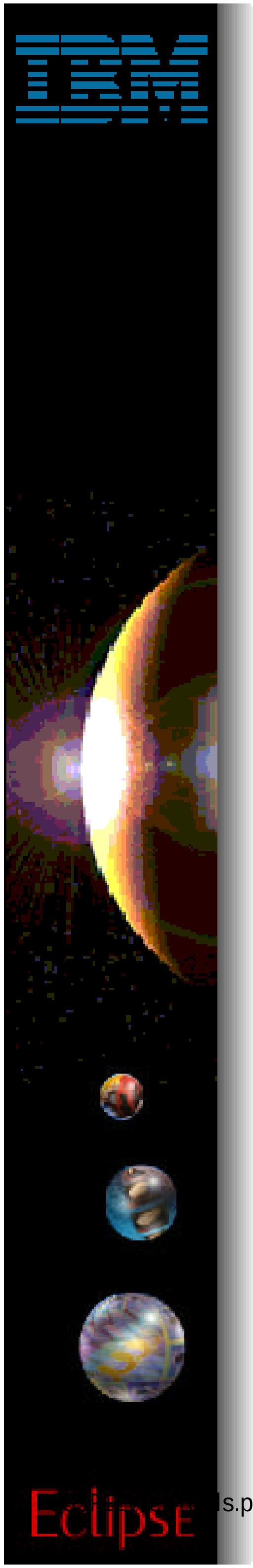
- Industry Standards



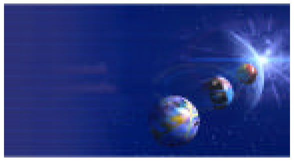
Standards



- Industry Standards Supported
  - **By WSSDa, WSAD and WDSc:**
    - EJB 1.1
    - Servlet 2.2
    - JSP 1.1
    - JRE 1.3
    - Web Services Definition Language (WSDL) 1.1
    - Apache SOAP 2.1
    - XML DTD 1.0 10/2000 Revision
    - XML Namespaces 1/99 Version
    - XML Schema 5/2001 Version
    - HTML 4.01 (other levels should work)
    - CSS2 (PageDesigner displays a subset)

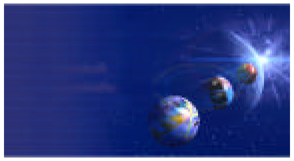


More Information?



Information Sources

- **www.ibm.com/software/ad/wds400**
  - As of June 4th
- **www.eclipse.org**
  - Eclipse and information about eclipse
- **www.ignite400.org**
  - Introduction to eclipse article
- **www.ibm.com/software/info1/websphere/partners/iseries.jsp**
  - WebSphere on iSeries home page for BPs
- **eServer iSeries magazine, July issue**
  - 3 articles on WDSc
- **www.ibm.com/websphere/developer**
  - WebSphere Developer Domain
  - Many articles and tutorials on technology and tools, including eclipse and WSWB and WebSphere Studio configurations



Trademarks & Disclaimers

8 IBM Corporation 1994-2002. All rights reserved.
References in this document to IBM products or services do not imply that IBM intends to make them available in every country.
The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

AS/400
AS/400e
e (logo) business
IBM
IBM(logo)
iSeries
OS/400

Lotus, Freelance Graphics, and Word Pro are registered trademarks of Lotus Development Corporation and/or IBM Corporation.
Domino is a trademark of Lotus Development Corporation and/or IBM Corporation.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.
UNIX is a registered trademark of The Open Group in the United States and other countries.
SET and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.
Other company, product and service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information in this presentation concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Some information in this presentation addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM's current investment and development activities as a good faith effort to help with our customers' future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.