Web Tools for IBM System i5 developers

ibm.com
the power of one

IBM System i5
Technical conference 2006

Claus Weiss
weiss@ca.ibm.com

WebTools agenda

- WDS overview and AD roadmap
- Website creation
- Webpage templates
- Designer for static WebPages
- Designer for Web objects
- Tool for Cascading Style Sheets
- Tools to create WebApplications
- Designer for dynamic WebPages
- Tools to create WebServices

WebSphere Development Studio

RPG | COBOL | C/C++ | PDM SEU SDA RLU

Current 5722-WDS customers with software subscription for V5R3, to upgrade to WDSc V6.0 use feature #: 2656
Available after GA

Upgrade from WDSC 6.0 to 6.0.1 using Rational Product updater

Unlimited Licenses

New WDSc Lite
Technology preview

iSeries	iSeries	iSeries	iSeries	Web Facing WDHT support	iSeries Projects	+CODE +VisualAge RPG
Java™	Debug	Struts Web	Web Service		RSE	
JSF	EGL Java generation	Trace	Profiling	DB	XML	App Server	HATS Toolkit

www.ibm.com/software/awdtools/iseries

WebSphere Development Studio Client V6.0.1 based on RWD V6

WebSphere Development Studio Client Advanced Edition 6.0.1

Workstation License
order through Passport Advantage
http://www.lotus.com/services/passport.nsf/WebDocs/Passport_Advantage_Home

Upgrade from WDSC 6.0 to 6.0.1 using Rational Product updater

+CODE
+VisualAge RPG

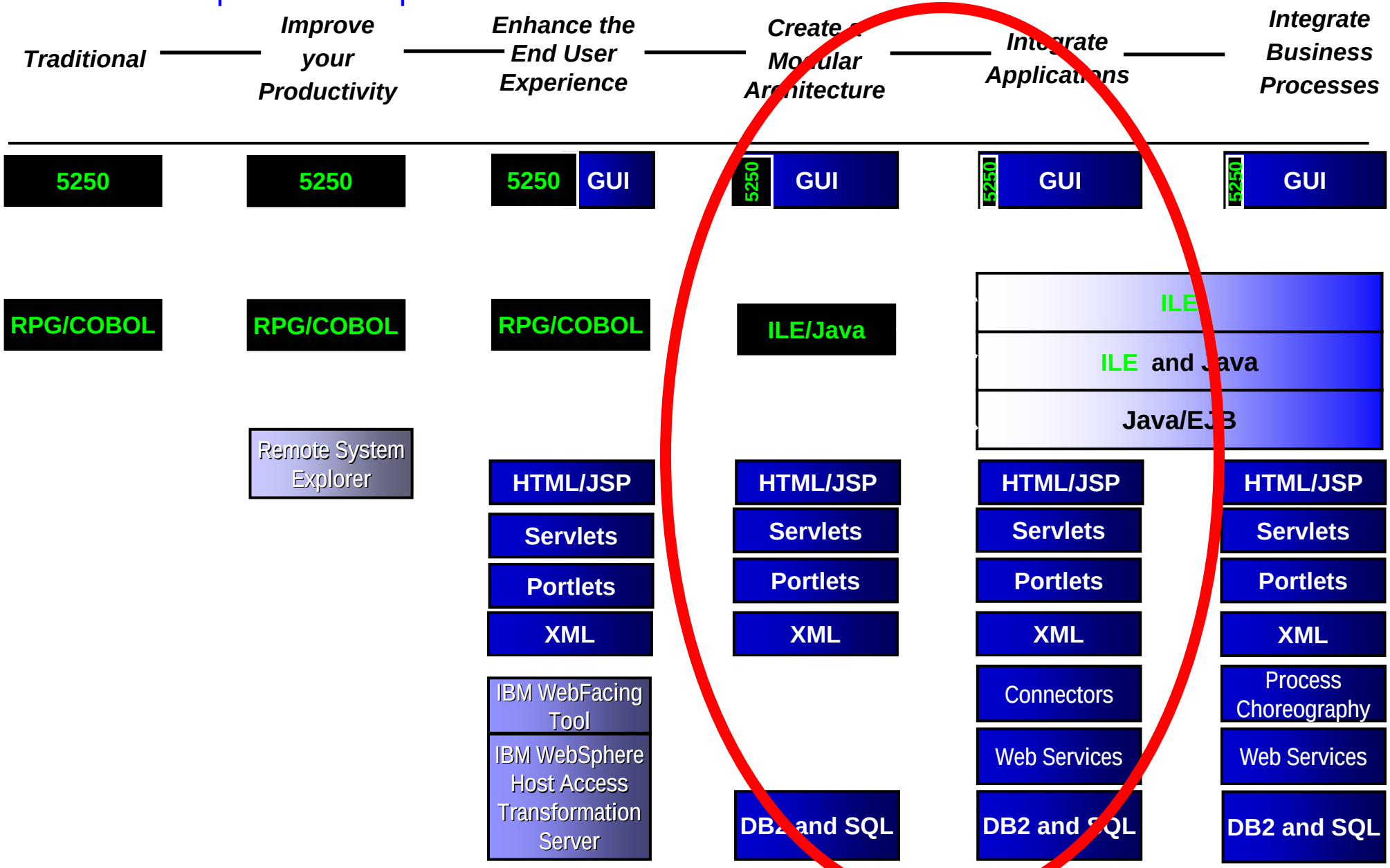
iSeries Java	iSeries Debug	iSeries * Struts Web	iSeries Web Service	Web Facing * **WDHT support**	iSeries Projects RSE		
JSF	EGL Java generation	Trace	Profiling	DB	XML	App Server	HATS Toolkit
	EGL * COBOL generation	EJB * J2EE *	Test * Cases	Portal * Toolkit			

New WDSc Lite
Technology preview

www.ibm.com/software/awdtools/iseries

WebSphere Development Studio Client V6.0.1 based on RAD V6

iSeries Developer Roadmap - Architecture

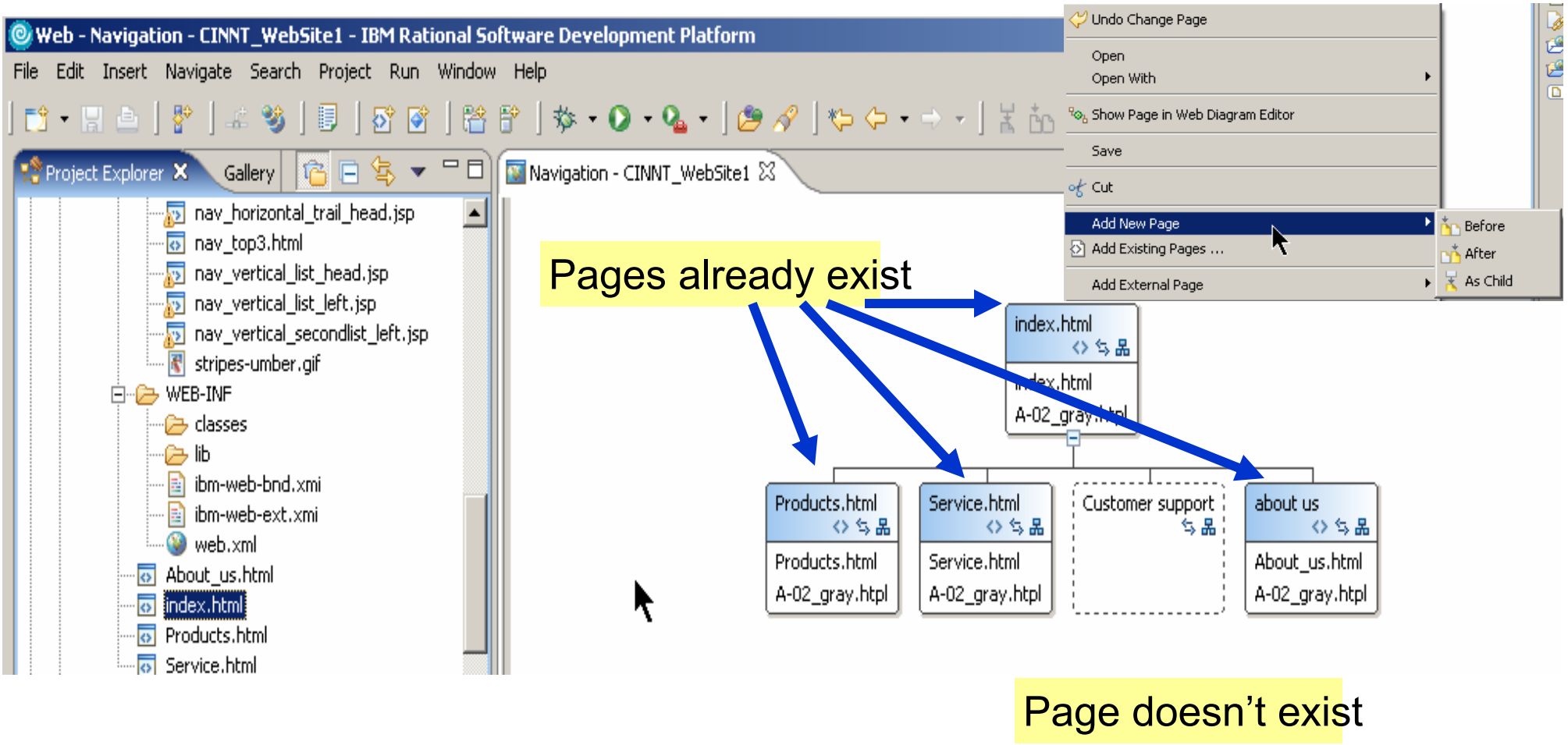


WebTools agenda

- WDS overview and AD roadmap
- Website creation
- Webpage templates
- Designer for static WebPages
- Designer for Web objects
- Tool for Cascading Style Sheets
- Tools to create WebApplications
- Designer for dynamic WebPages
- Tools to create WebServices

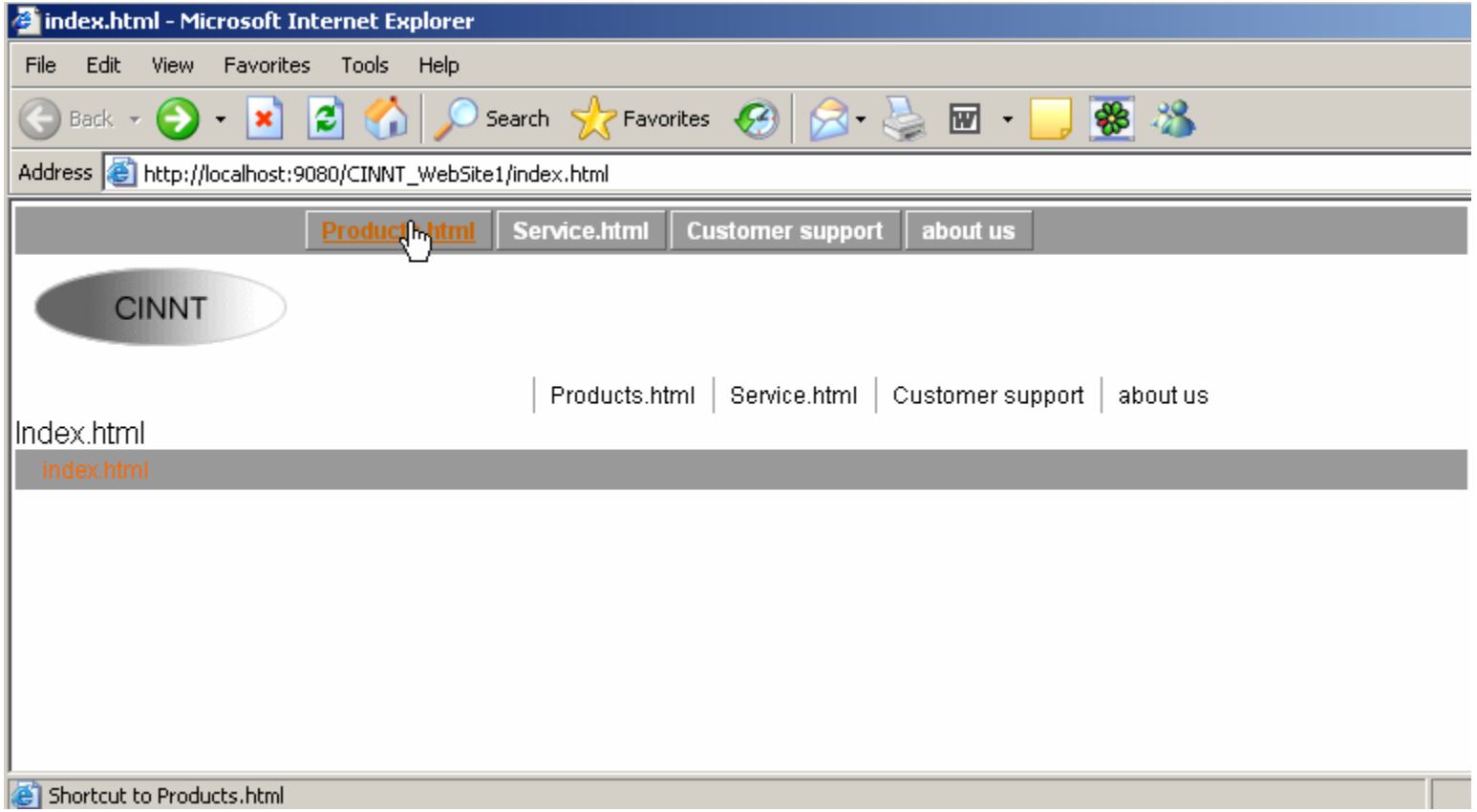
WebSite designer

Design page relations
Bind to real Webpages later

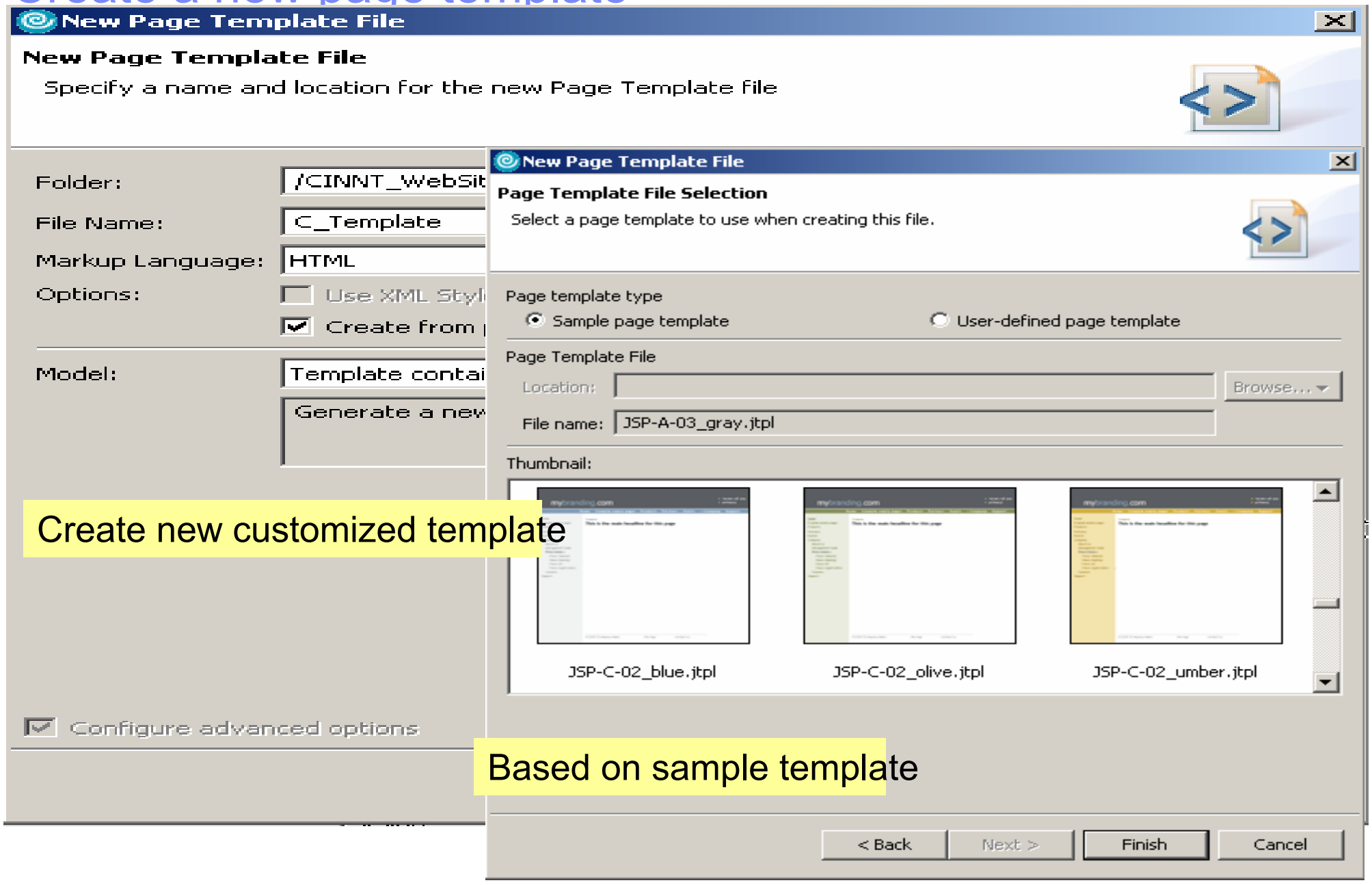


WebPage templates

Template used with Navigation bar
Navigation bar automatically updated from WebSite designer



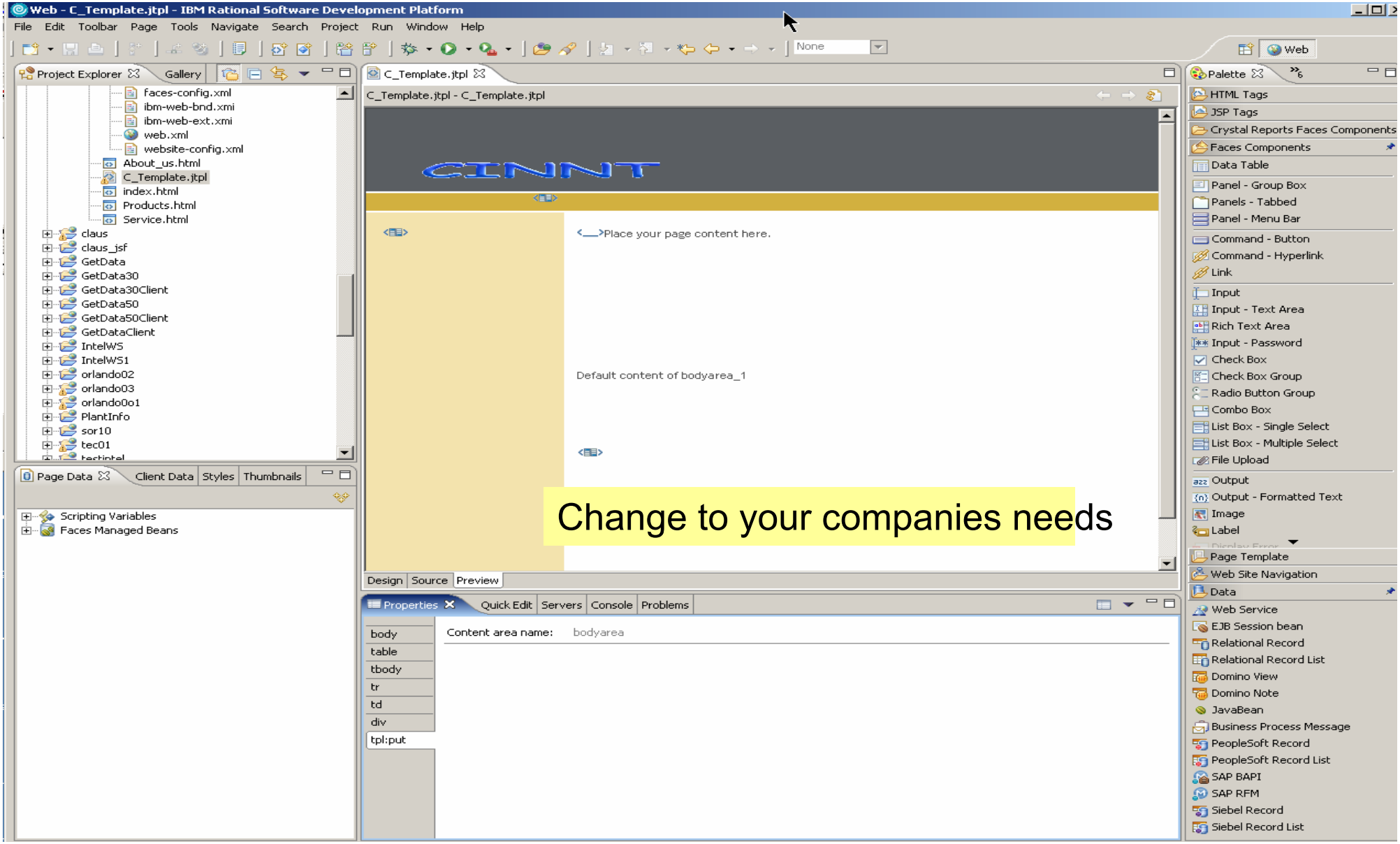
Create a new page template



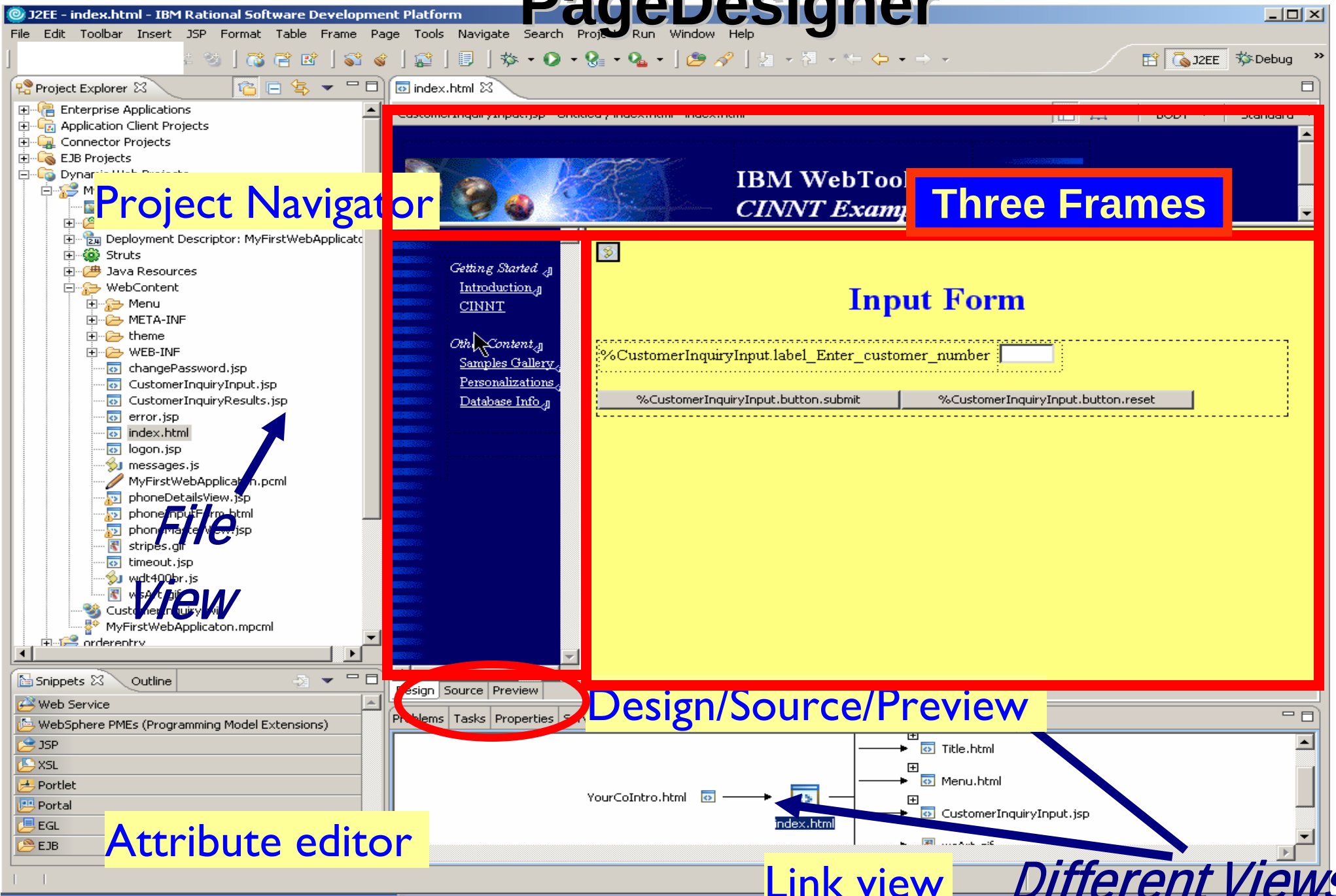
Create new customized template

Based on sample template

Change page template



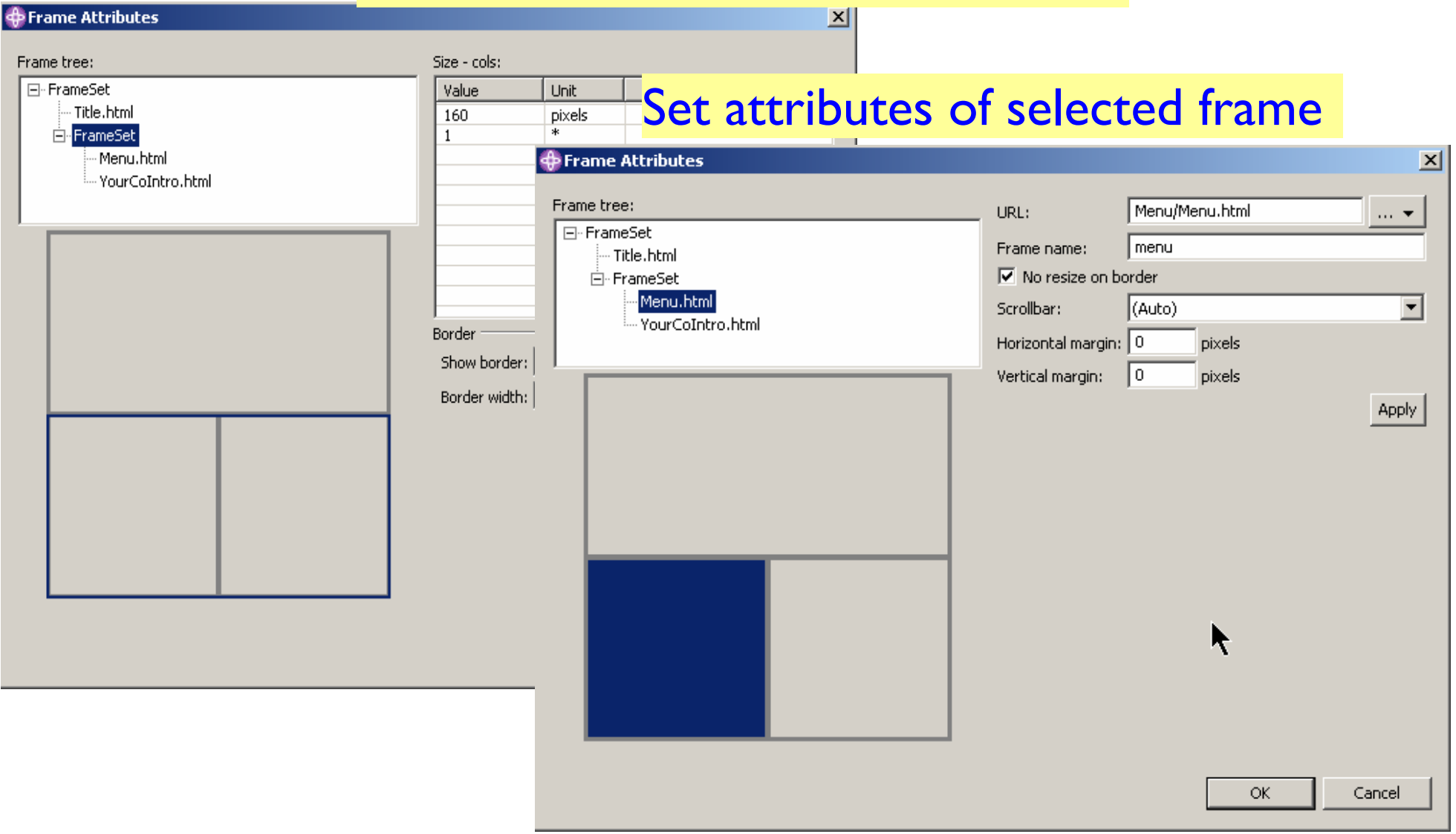
PageDesigner



Frame tools

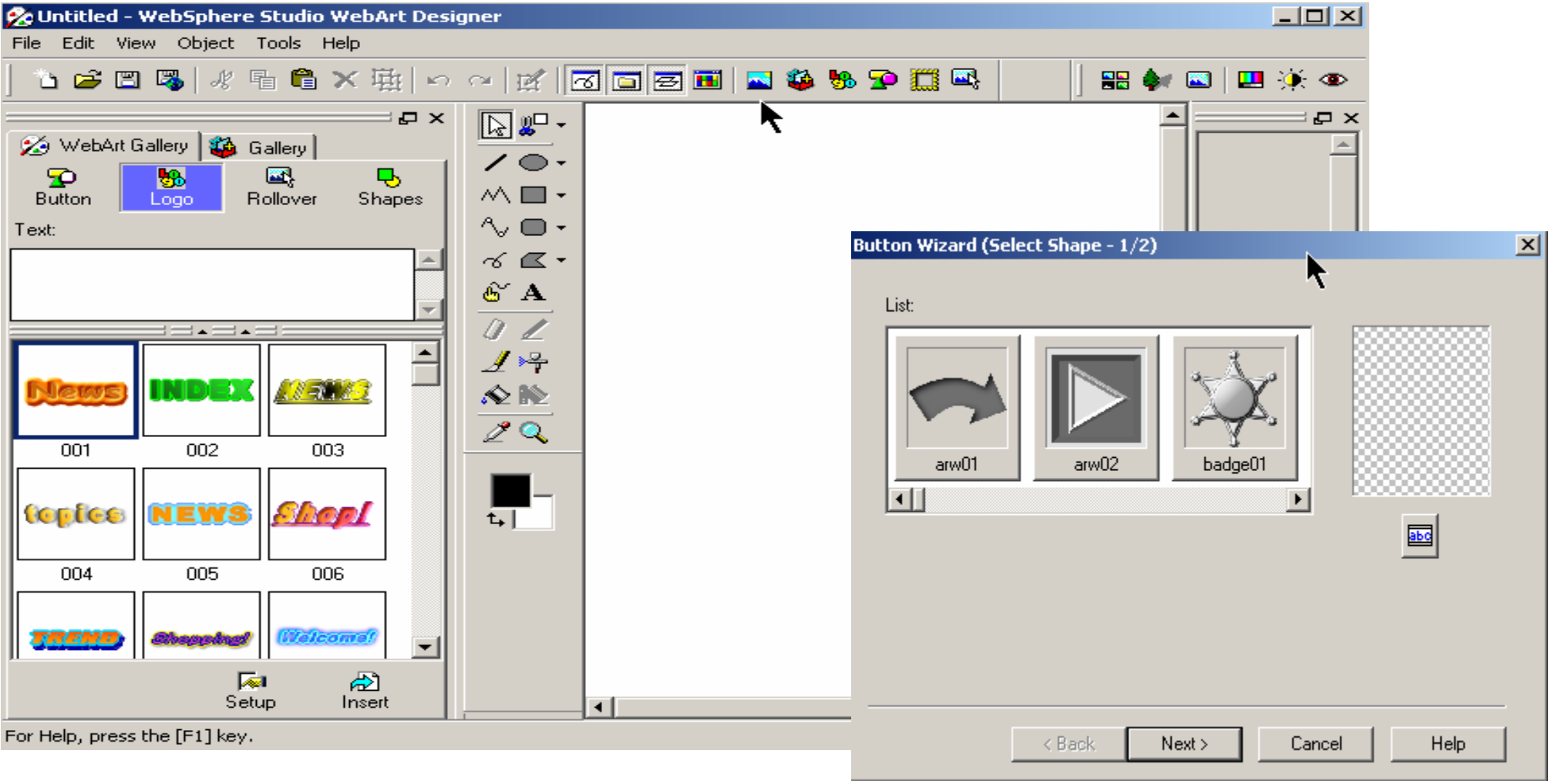
Set attributes of selected Frameset

Set attributes of selected frame

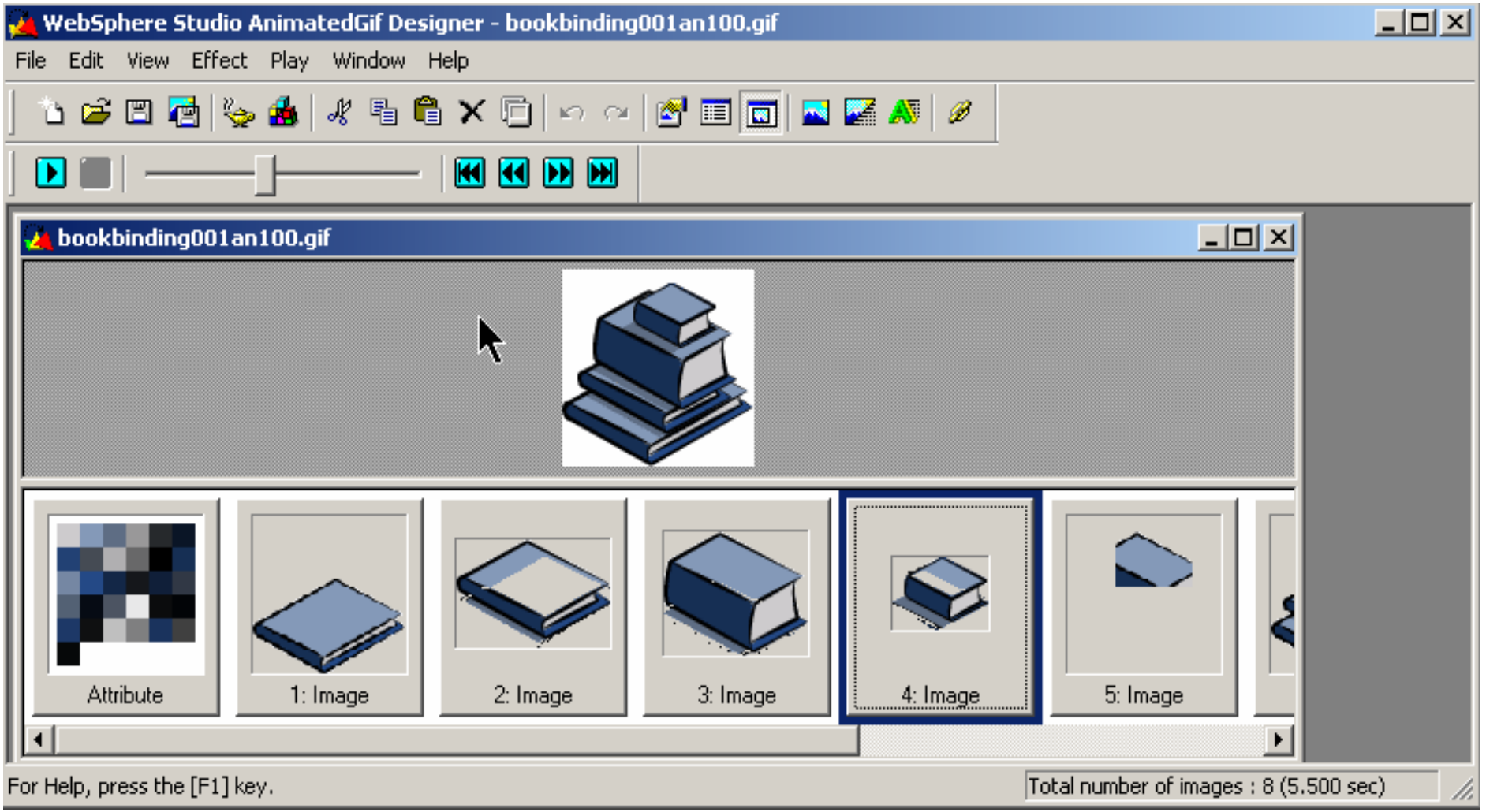


WebArt Designer

Wizards for Logo/Button/PhotoFrame/Rollover

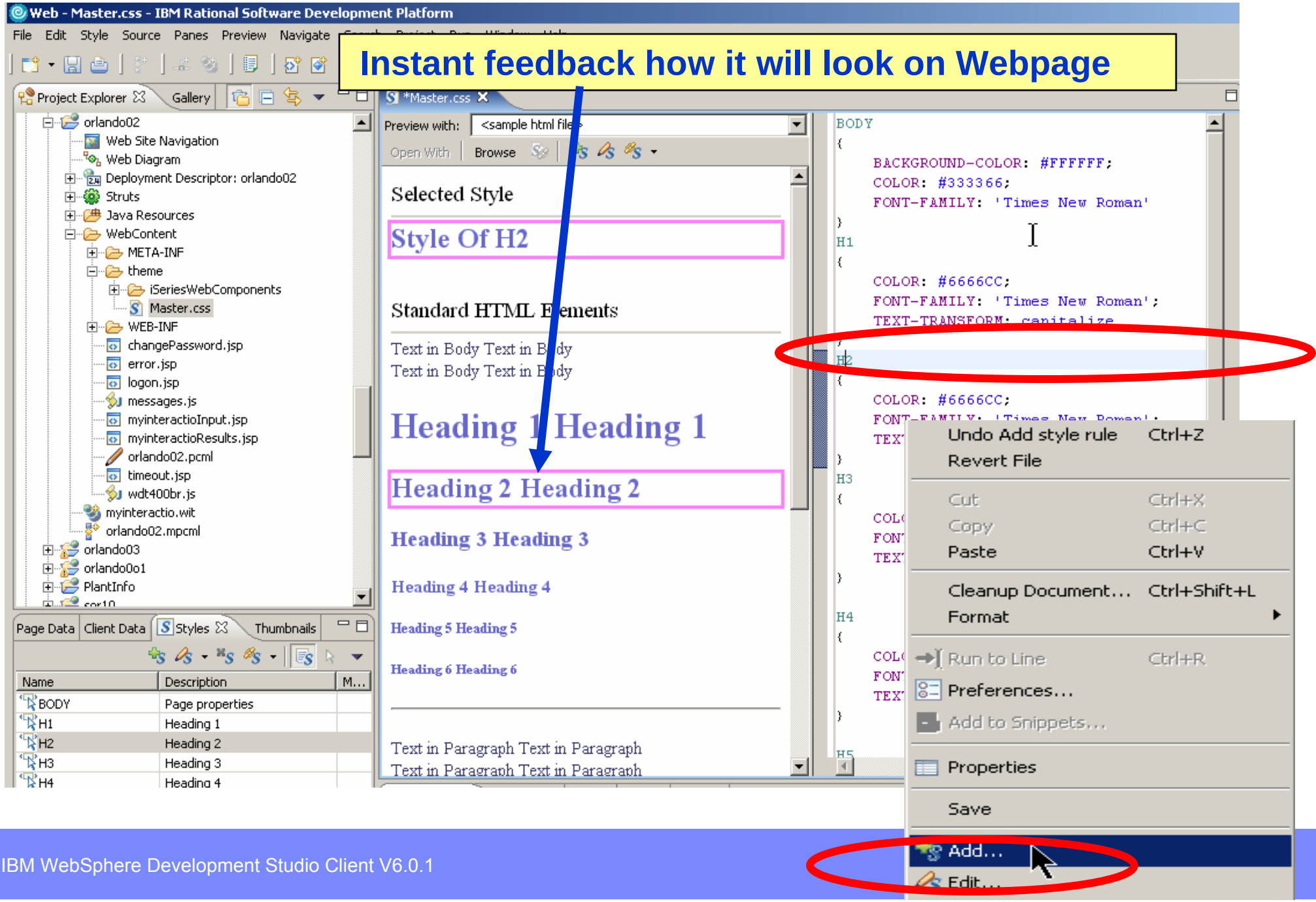


Animation GIF designer

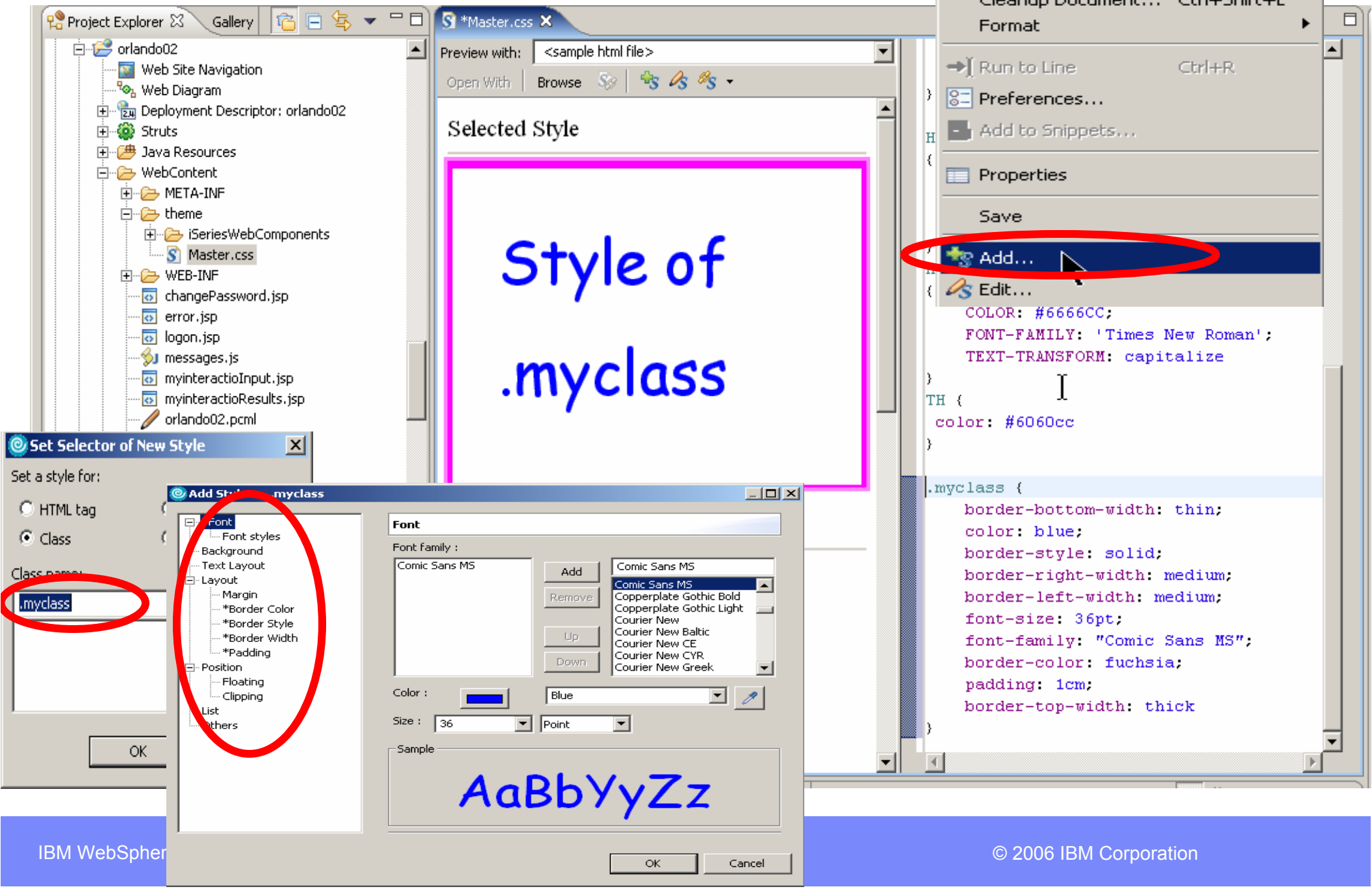


Running the application

Stylesheet editor



Creating your own style classes



WebTools agenda

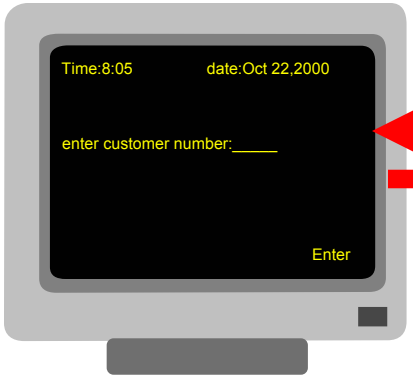
- WDS overview and AD roadmap
- Website creation
- Webpage templates
- Designer for static WebPages
- Designer for Web objects
- Tool for Cascading Style Sheets
- Tools to create WebApplications
  - Designer for dynamic WebPages
- Tools to create WebServices

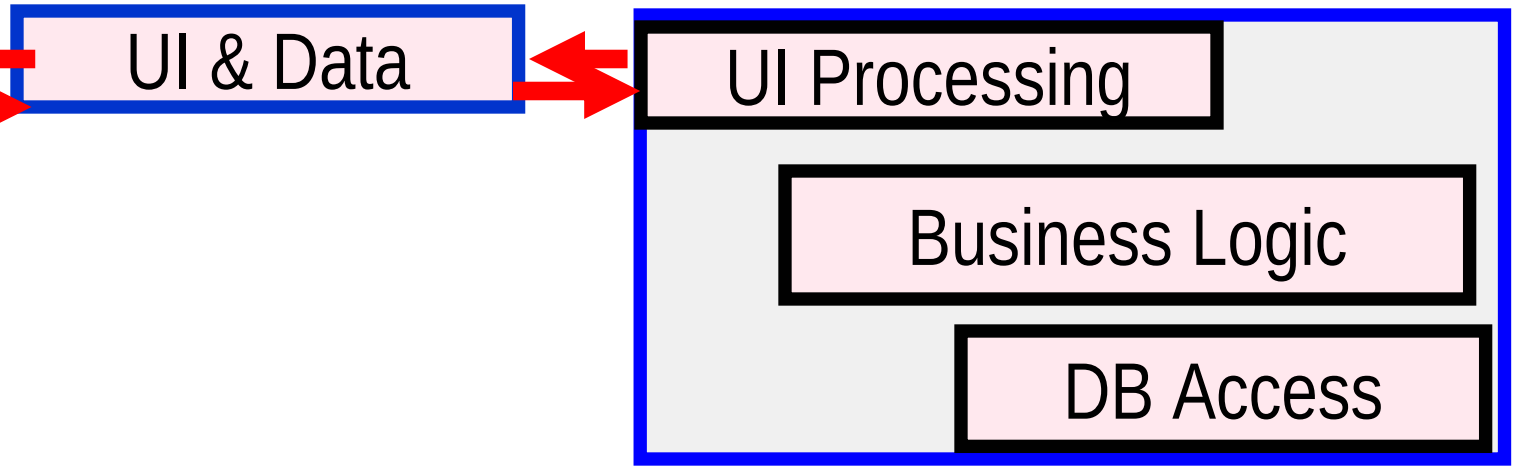
Web applications with dynamic webpages

- **Content of Webpages gets assembled at runtime**
  1. **Applets (application running in a browser)**
  2. **generate HTML at runtime on the fly**
  3. **Use predefined HTML with leaving holes for runtime information**
     - **Substitution variables**
     - **Java Server Page (jsp) standard way in J2EE (Java code on server fills holes)**
     - **Jsp with Struts framework (controller logic strictly separated)**
     - **Java Server Face (jsf) framework that also addresses UI)**

Today's Model

5250 Screens



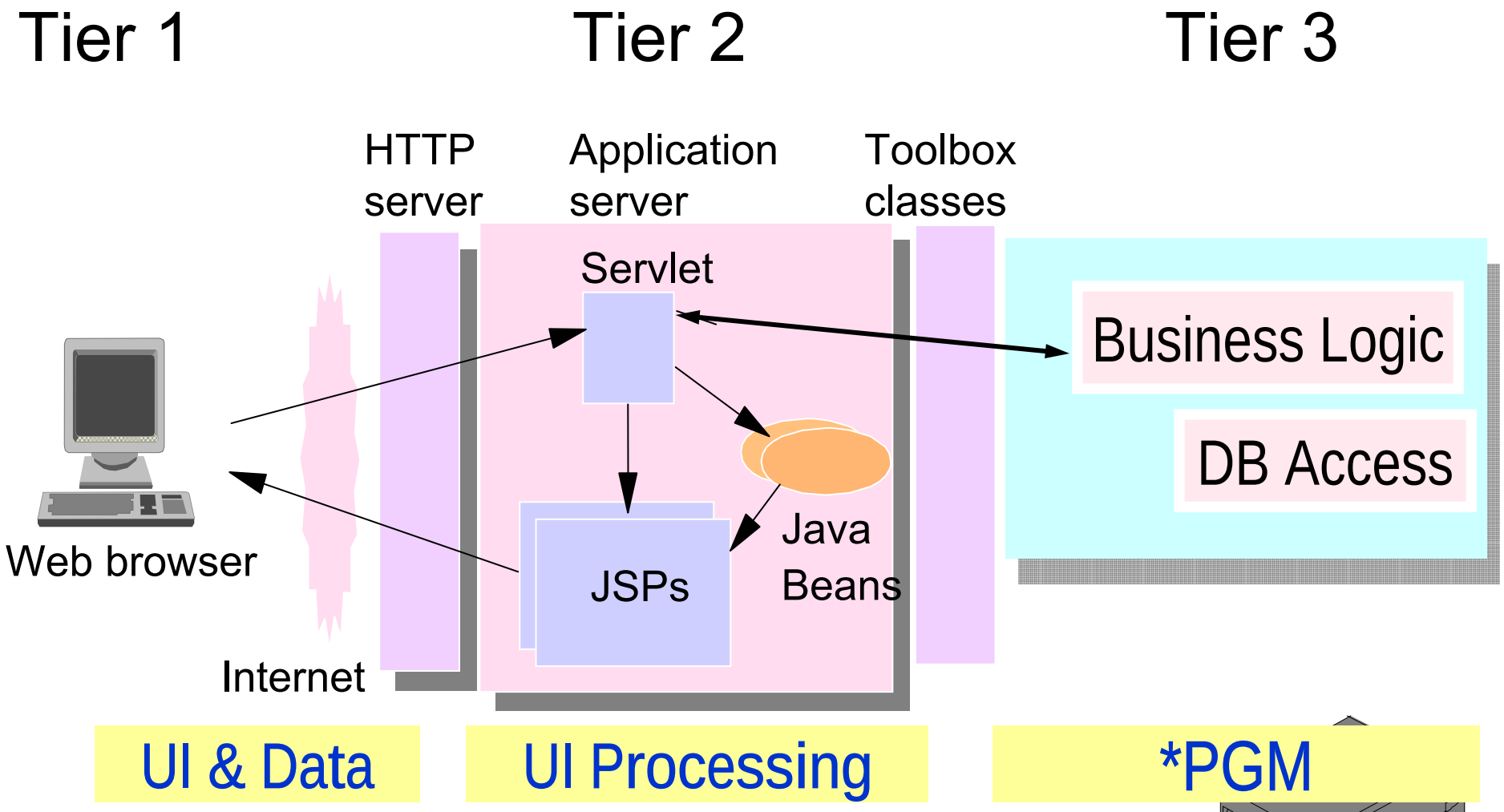


*PGM

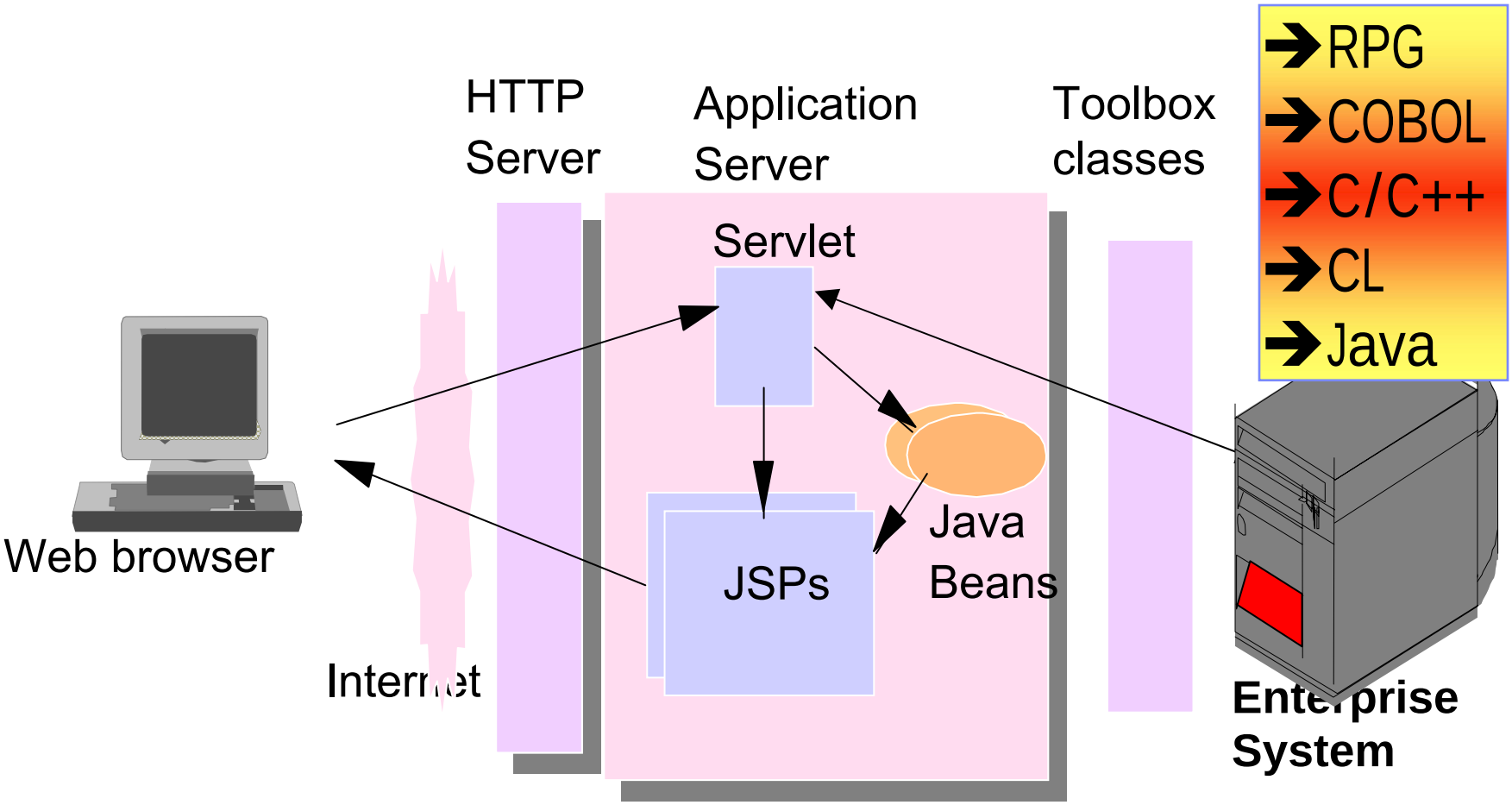
- Program puts up screen, waits for input
- Program processes input, does business logic

System i

Modern Web App Architecture



Tooling for modern web applications



WDSc➔ iSeries webtooling
- ➔Pagedesigner
  - ➔Jsf tooling
- ➔Interaction wizard
- ➔WebServices wizard

➔ iSeries 3 GL tooling
- ➔Remote system explorer
- ➔LPEX source editor
- ➔Integrated Debugger

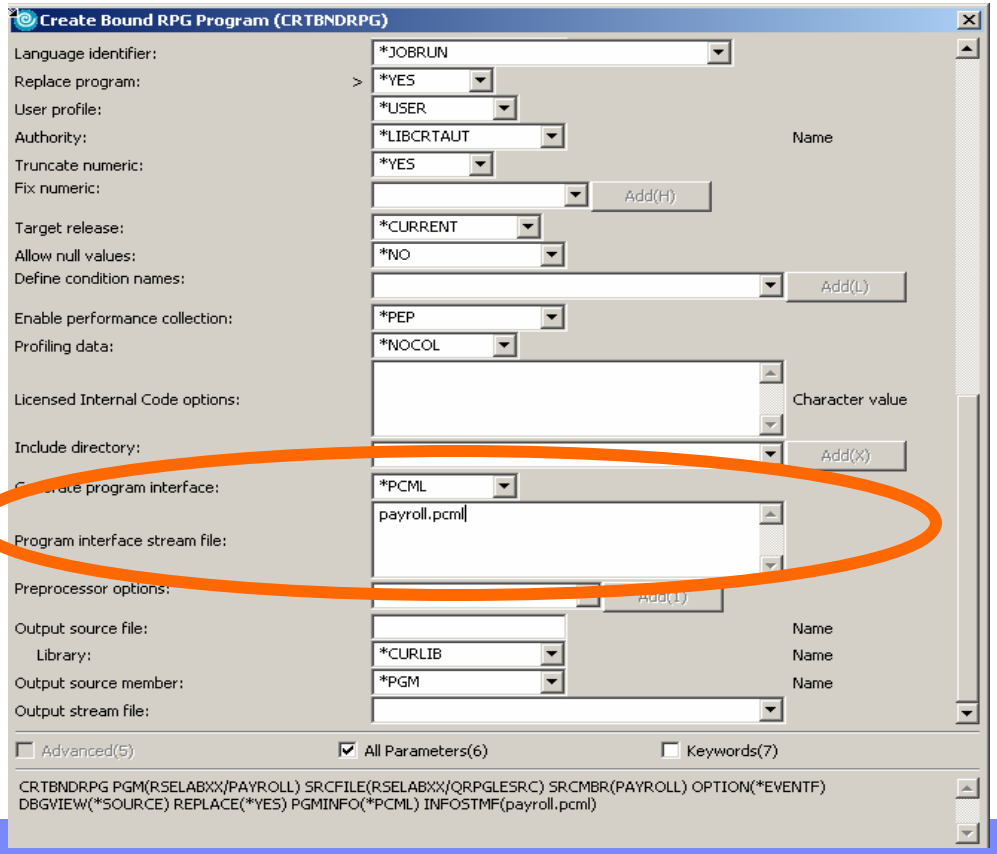
Calling native programs from Java

- A brief look at: IBM System i5 Java toolbox
- Java toolbox is included in WDSc
- Lets look at: Using Program Call Markup Language (PCML)

ILE RPG and COBOL compilers can create PCML for you

➔OS/400 V5R2 and higher:

-New parameters in CRTxxxMOD and CRTBNDxxx



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.

Use PCML to describe program interface

RPG/COBOL compilers create PCML

No need for you to write PCML

```
<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>R
```

Program Call Markup Language (PCML)

Call RPG from Java Servlet

```
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
```

Using PCML to invoke native system I programs

1. First lets look at WebProjects
2. Interaction wizard
3. Jsf tooling
4. Web Services tooling

J2EE Web Application

- Standard Web App folder structure:

```
+Web application folder (root folder)
   +source
      –all non-deployed files (java)
   +WebContent
      –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

J2EE 1.3

Terms: Web Application

- Example Web Application

```
+accounts
  +source
  +WebContent
    –index.html
    +receivable
      –page1.html
    +payable
      –page1.html
    +META-INF
      –MANIFEST.MF
    +theme
      –corporate.css
    +WEB-INF
      –web.xml
      +classes
      +lib
```

www.mydomain.com/accounts

www.mydomain.com/accounts/receivable/page1.html

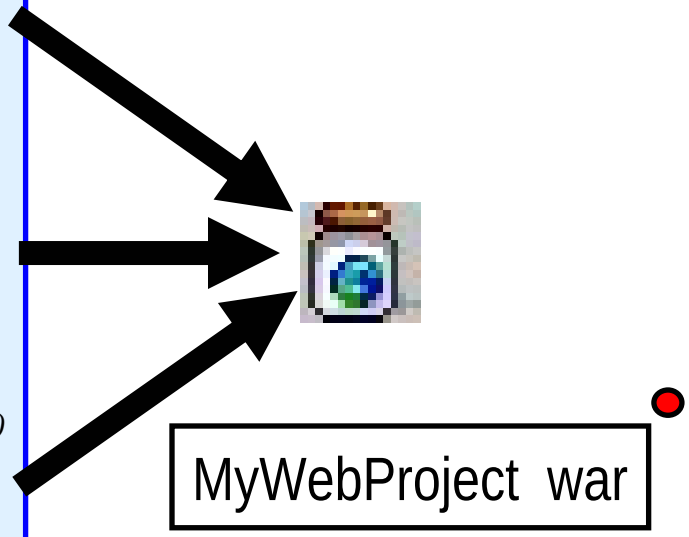
www.mydomain.com/accounts/payable/page1.html

Web Archive Files (WAR)

J2EE 1.3

- **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

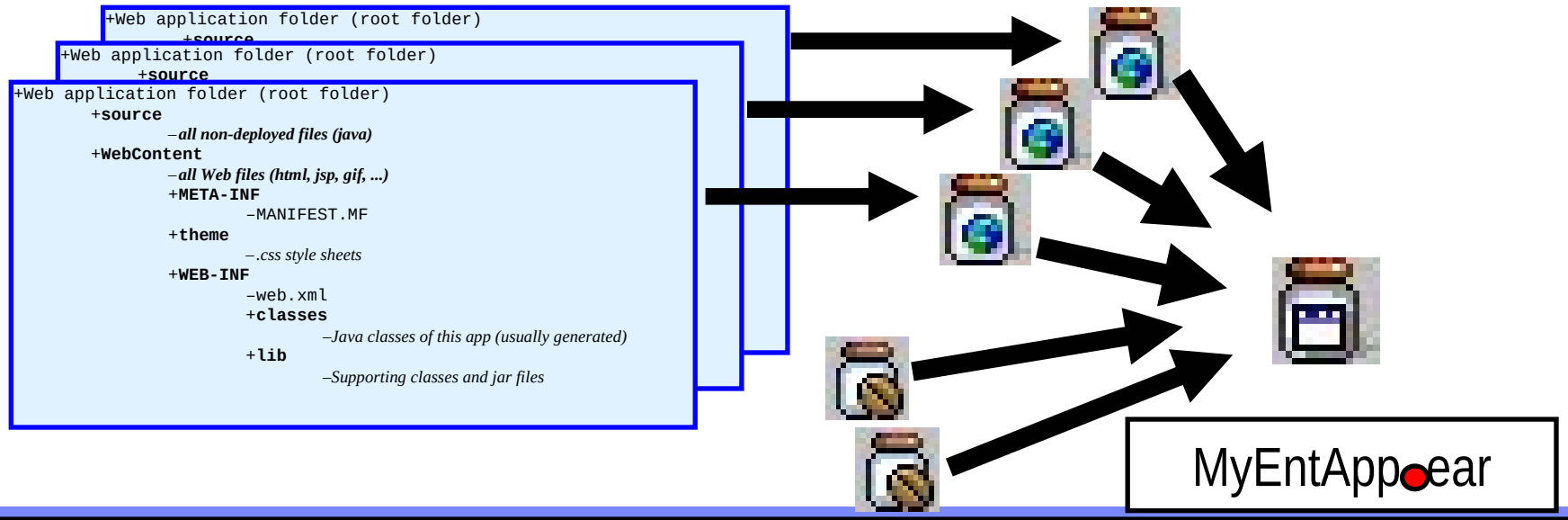
```
+Web application folder (root folder)
     +source
          –all non-deployed files (java)
     +WebContent
          –all Web files (html, jsp, gif, ...)
          +META-INF
               –MANIFEST.MF
          +theme
               –.css style sheets
          +WEB-INF
               –web.xml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



Enterprise Archive Files (EAR)

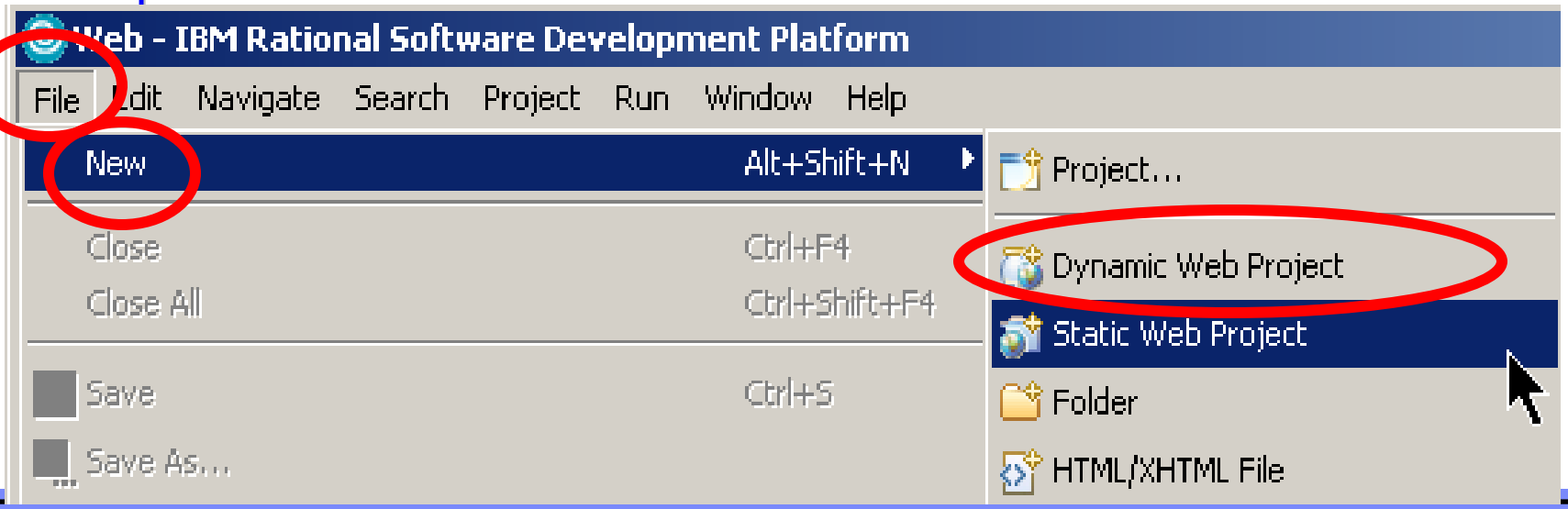
J2EE EJB Spec

- **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)



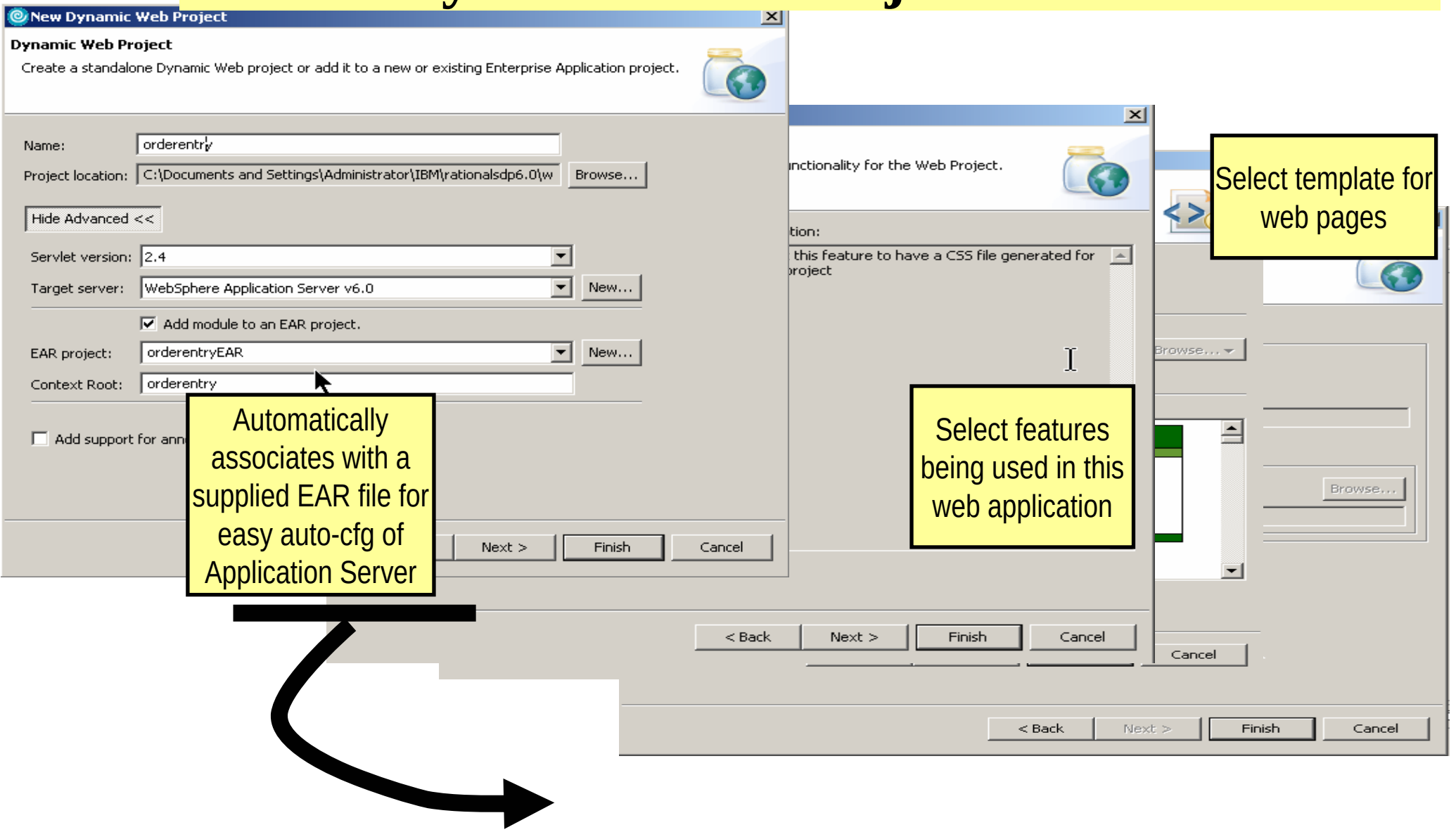
Web Projects

- Create new web applications in a web project
- One of the project types in WDSc
  - With its own web perspective
  - With its own "new" wizard
  - With its own set of tools
- Created with J2EE folder layout
- Created with simple web.xml file
  - Automatically updated as resources are created
  - Has specialized web.xml editor



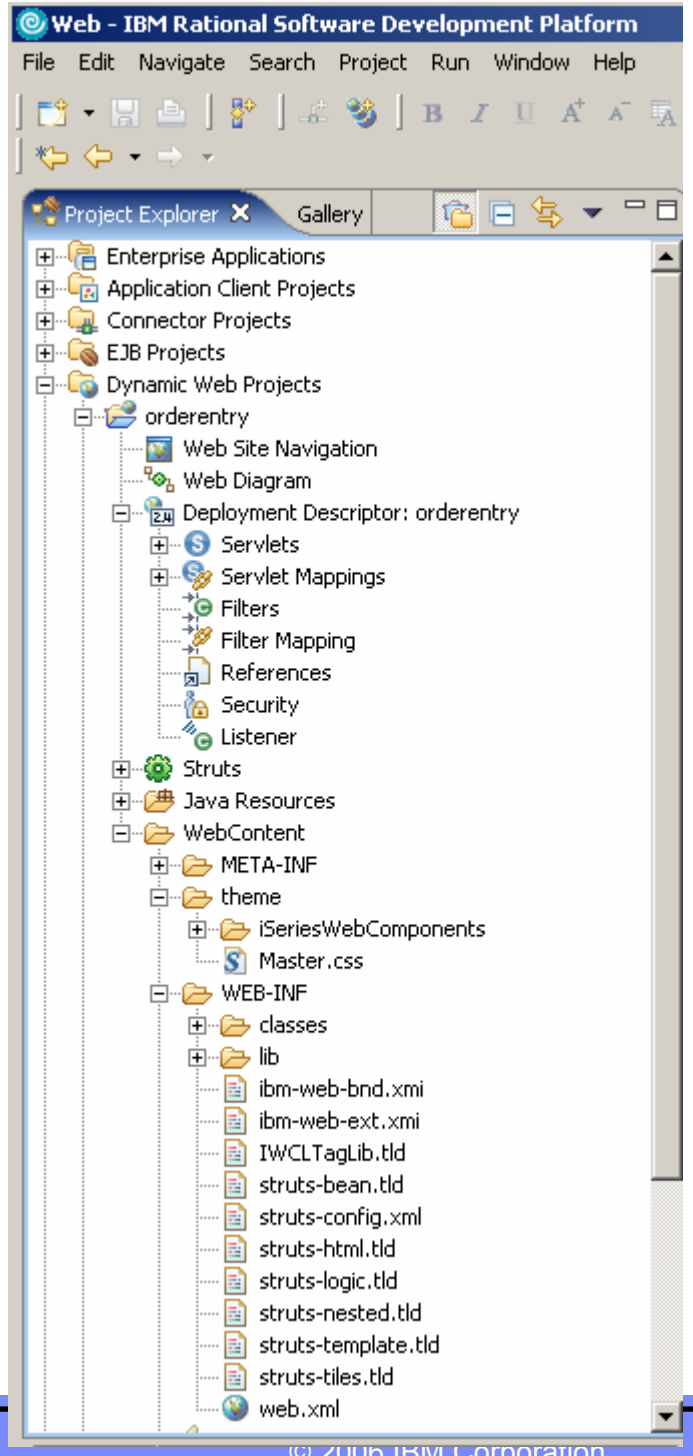
Web projects

New Dynamic Web Project wizard



Web projects

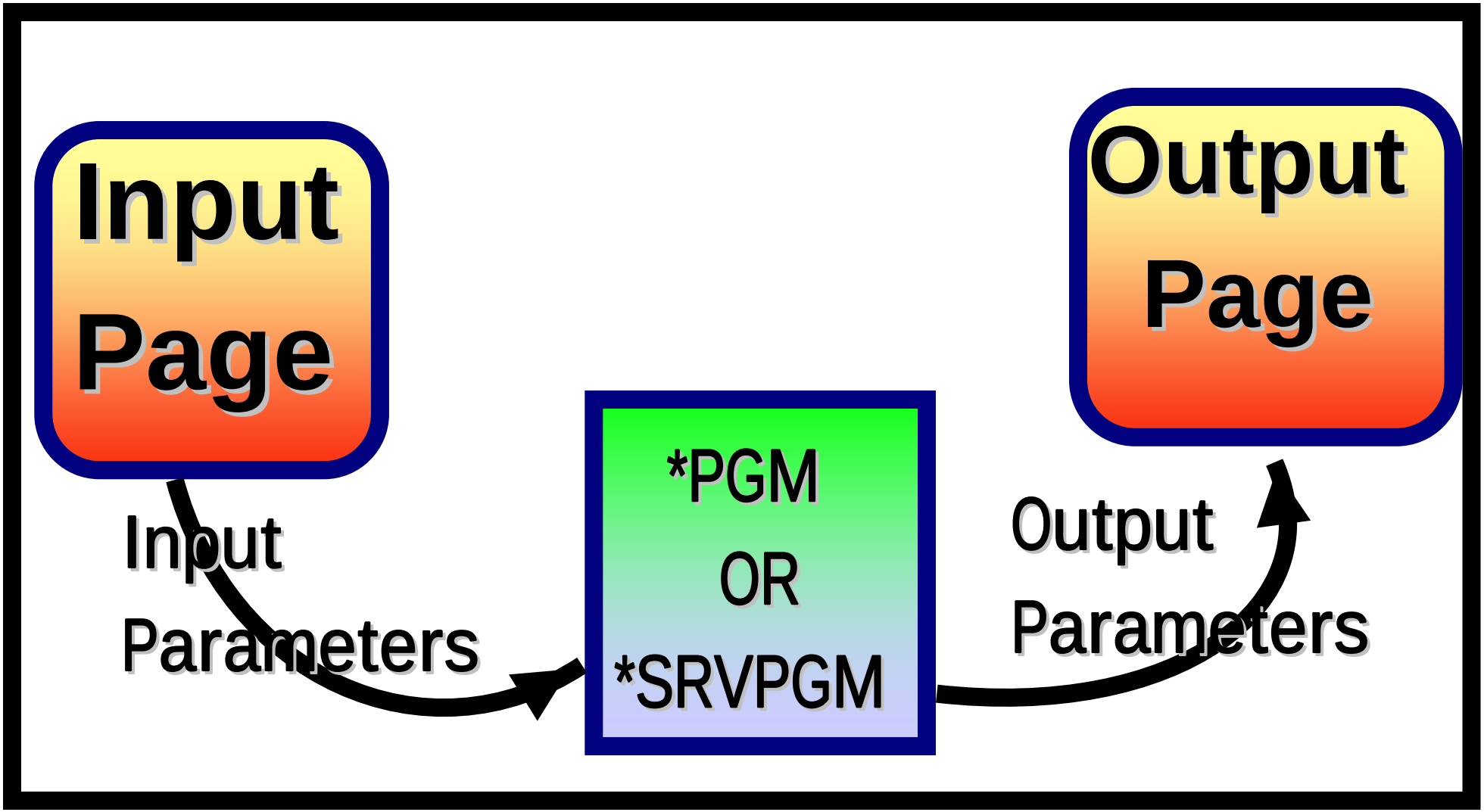
Web application structure is created from information provided in New Dynamic Web Project Wizard



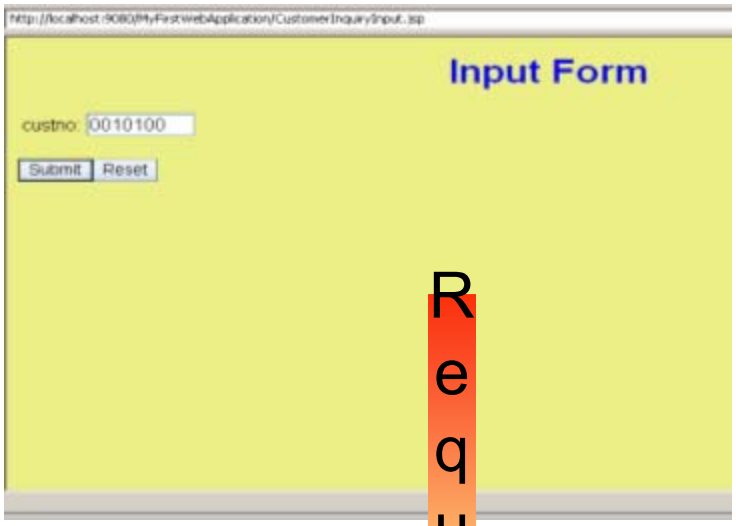
Web Tool for iSeries

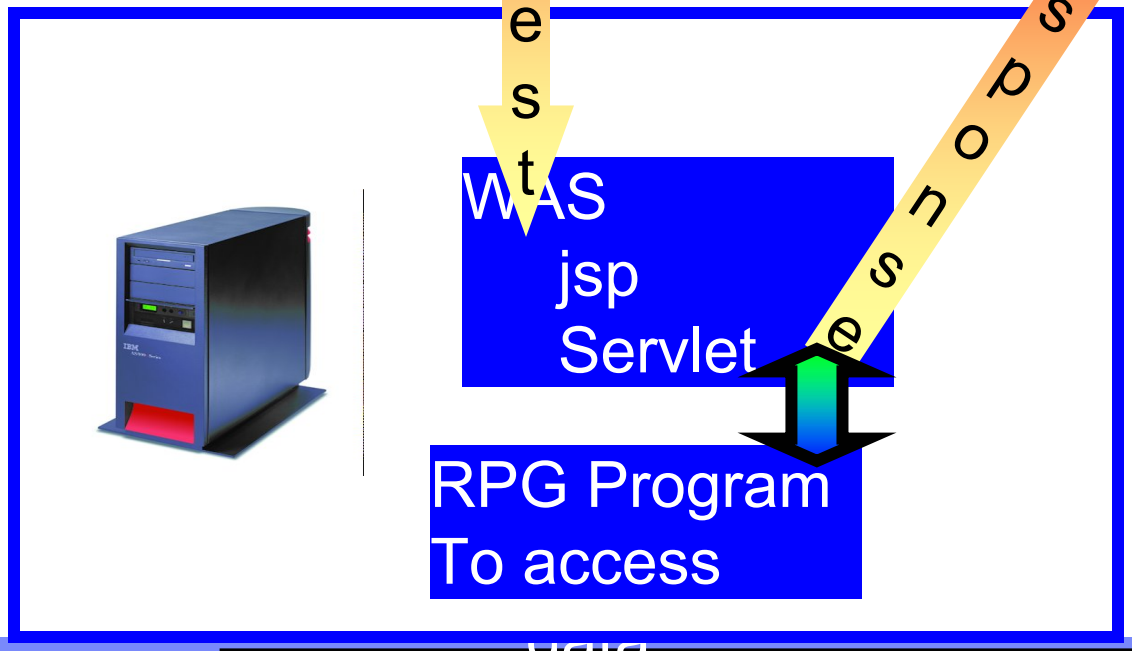
- Wizards to help create iSeries RPG/COBOL web applications
  - Interaction wizard to build web application Struts based
  - JSF tooling
  - WebService wizard to build an iSeries WebService
  - JCA wizard to build an iSeries Java Connector Architecture (JCA) connector

An interaction



Build this simple interaction





Using the Interaction wizard

- Interaction wizard specific for iSeries development (not extendable)
- Steps involved
  - Create a dynamic web project
  - Create Runtime configuration
    - iSeries server
    - User id/password
    - Library list setup
  - Create Interaction
    - Input screen
    - Program interface (PCML)
    - Output screen

New standard framework java server faces (jsf)

- **Wait something better than STRUTS is here**

- A new standard framework fully supported in WDSc
  - Tool support for jsf
    - Page designer
    - Binding support of data beans and UI controls
    - Web diagram editor for jsf
    - Quick edit for small pieces of Java code

Java Server Faces (jsf)

- A specification and reference implementation for web applications
  - Components
  - Events
  - Validators & converters
  - Navigation
  - Back-end-data integration
- Standard
- Allows for extended tool support

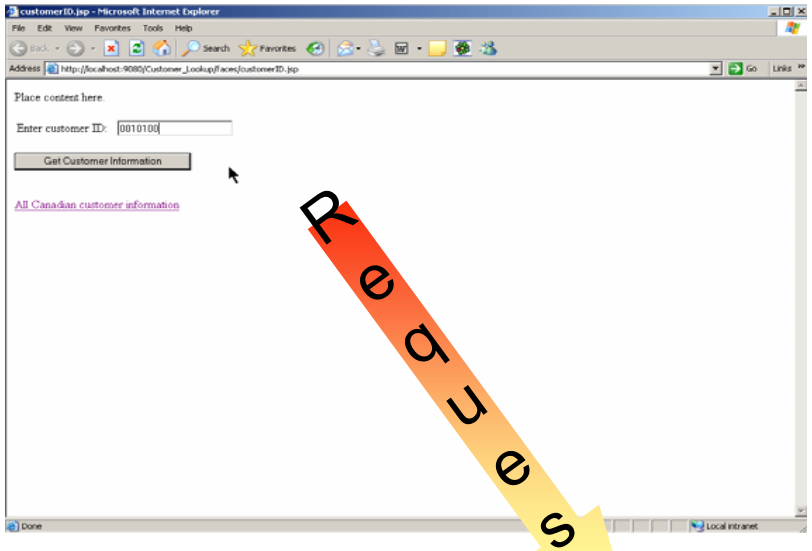
Web Page Development

- JSF contains a collection of UI components to make Web page development easier
  - Each component has extensive properties for customizing look and behavior of component
  - Components can be bound to variables for automatically:
    - Displaying the value of variable
    - Assigning value to a variable on page submit
  - Components have associated “events” which you can write code for
    - JavaScript for client side events
      - onclick, ondblclick, onkeypress, ...
    - Java code for server side events
      - Value Changed, Command, ...

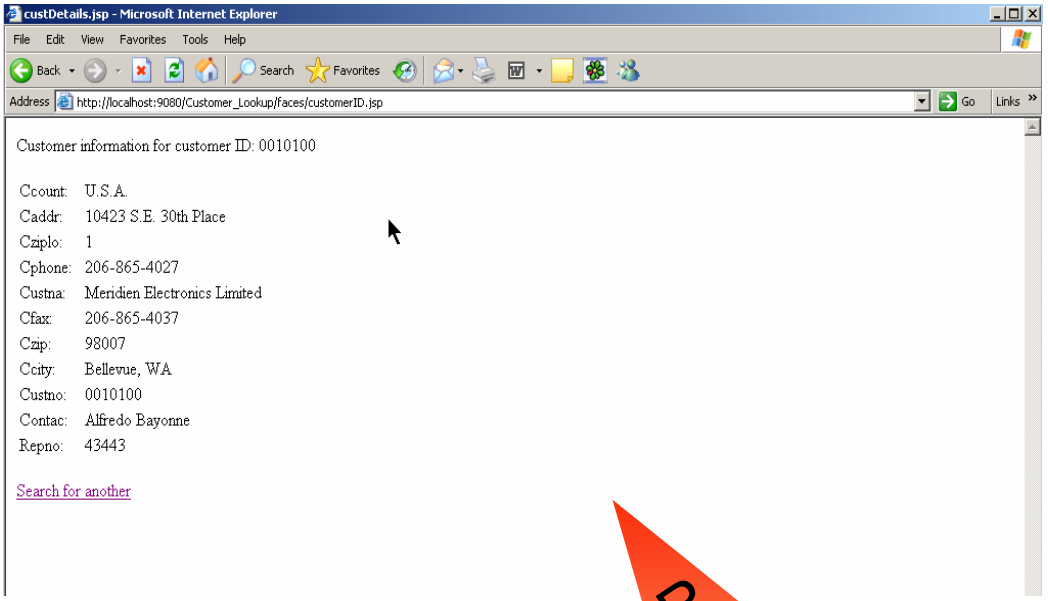
Creating a jsf based web application

- Using
  - Web diagram editor
  - jsf controls and associated tools
  - Program call wizard
  - Jsf Data tools

Build this jsf web application



Request

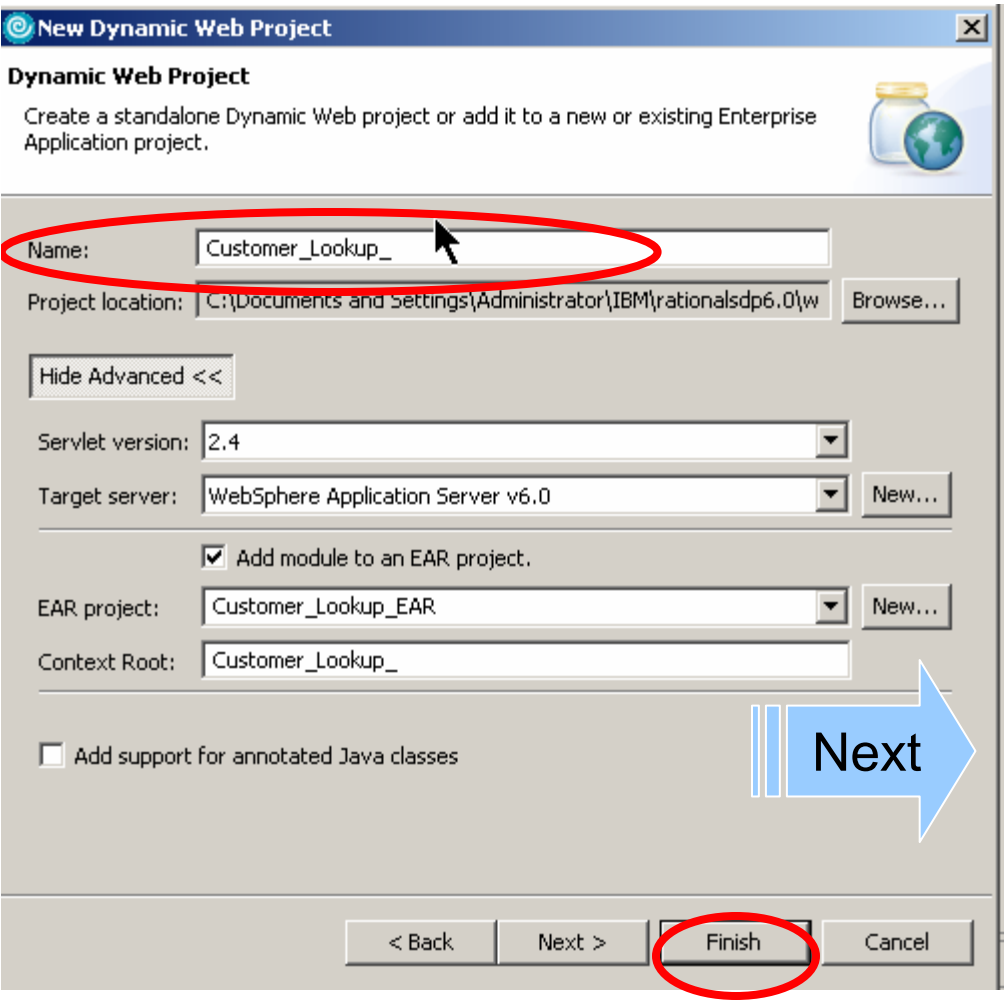


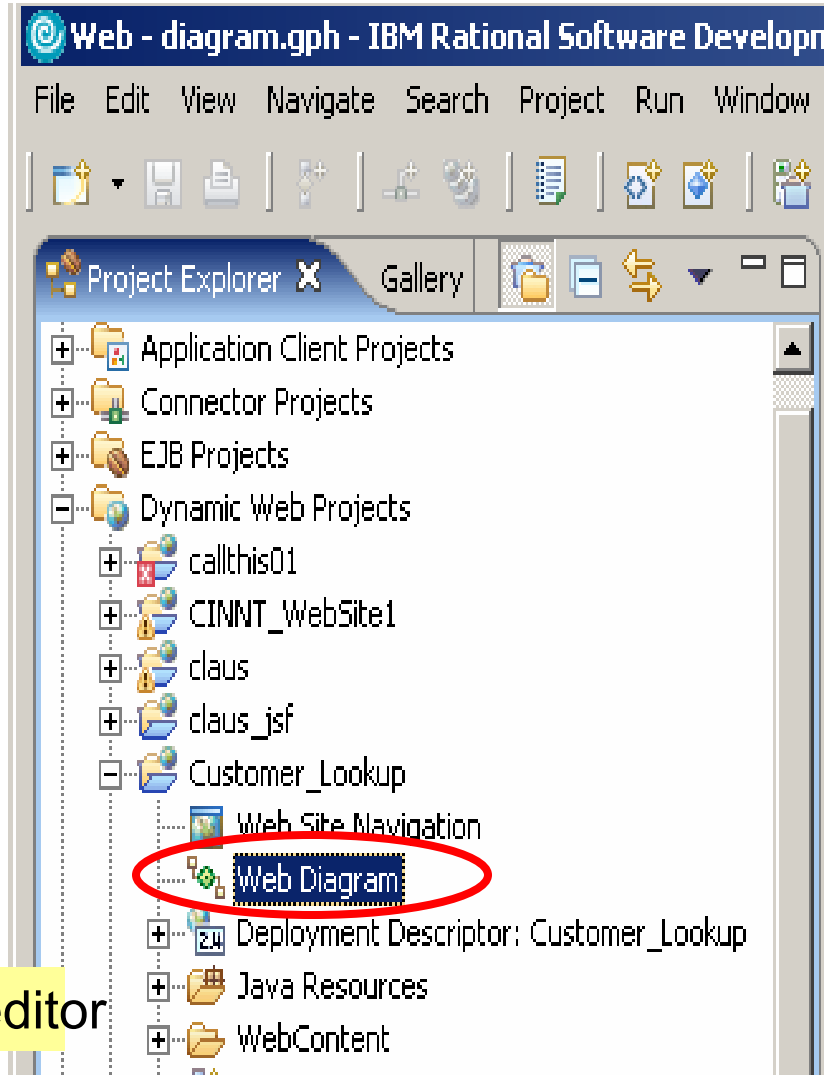
Response

RPG Program
To access
data

jsf
- Validation
- Conversion and put values from UI control into Program call bean
- Use Program call bean to invoke program
- Get return values from Program call bean into bound UI controls

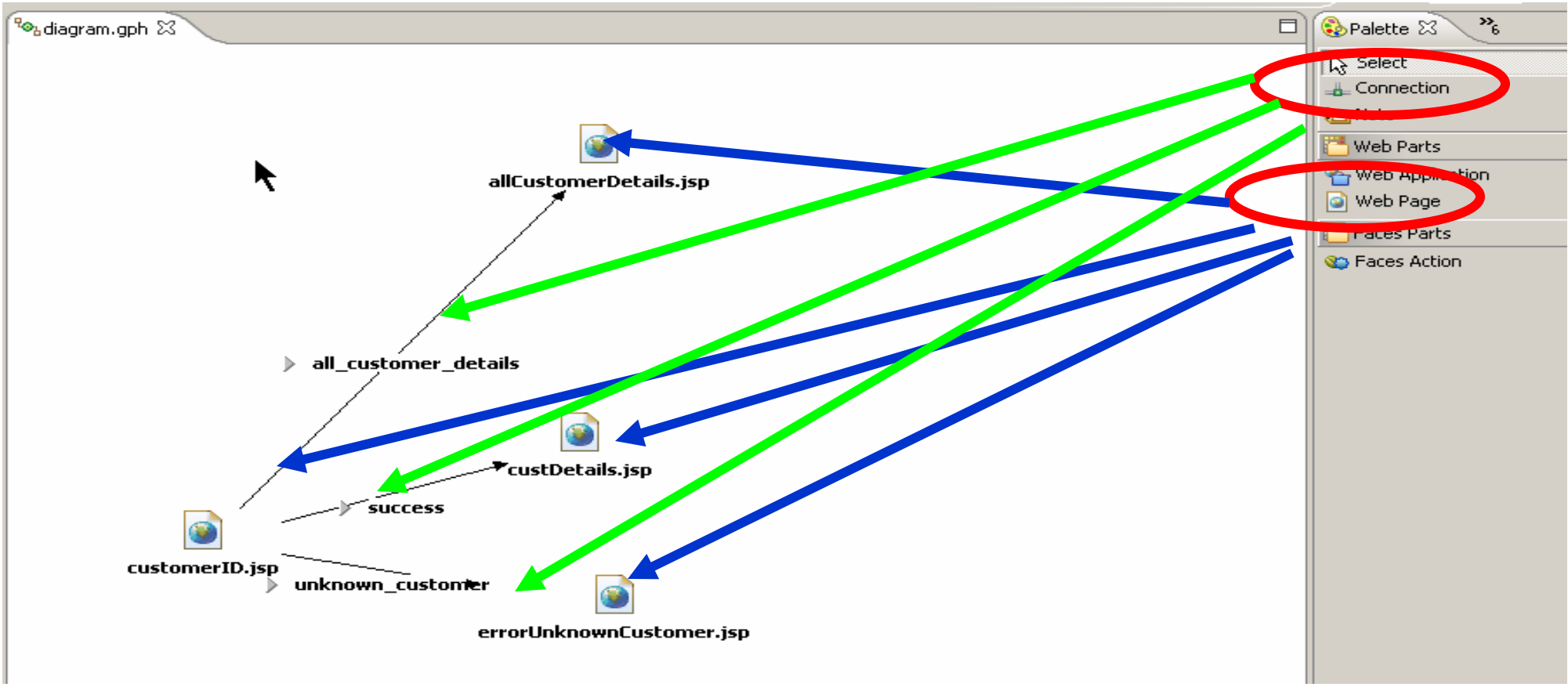
Create a Web project





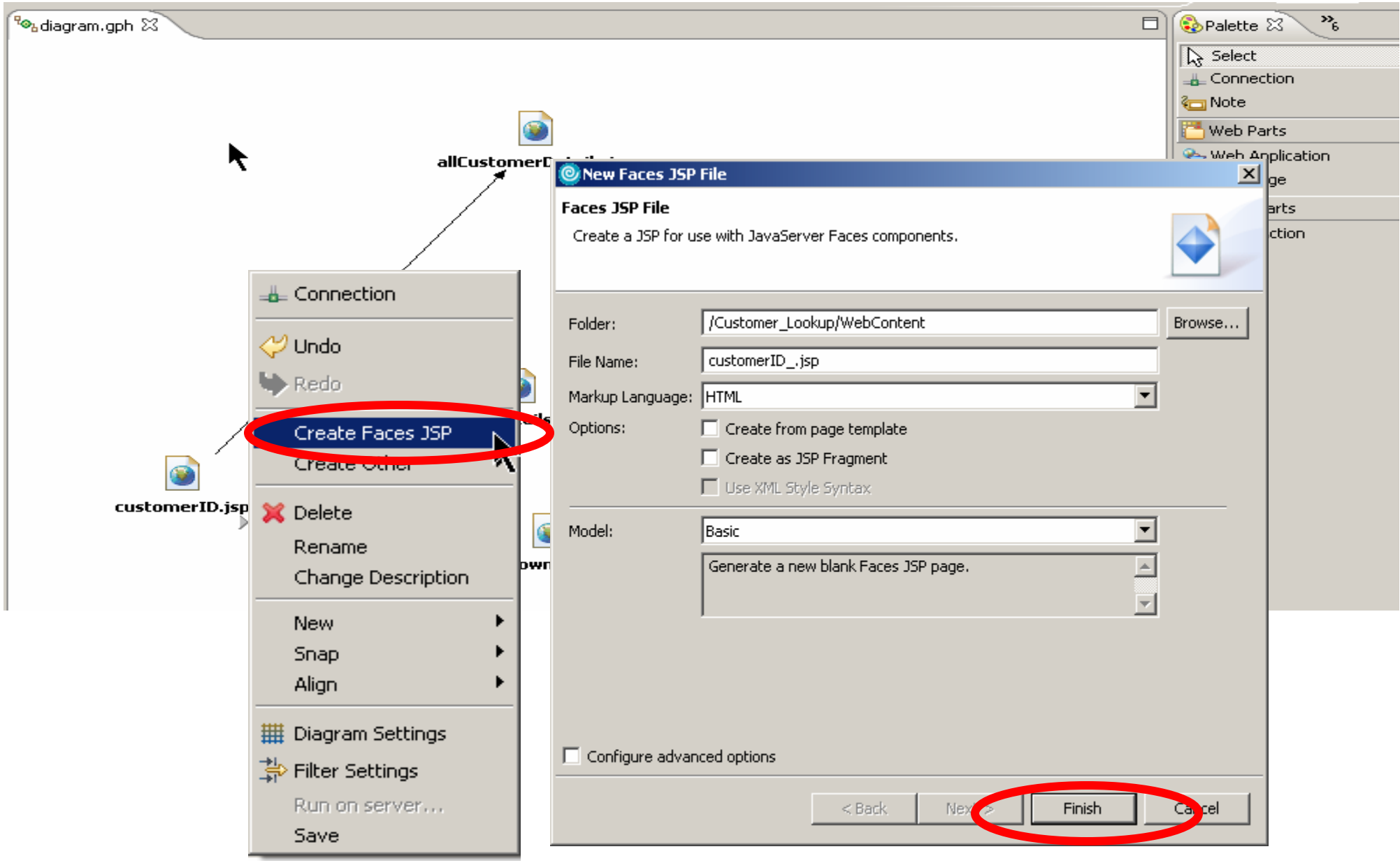
Create project and invoke WebDiagram editor

WebDiagram editor

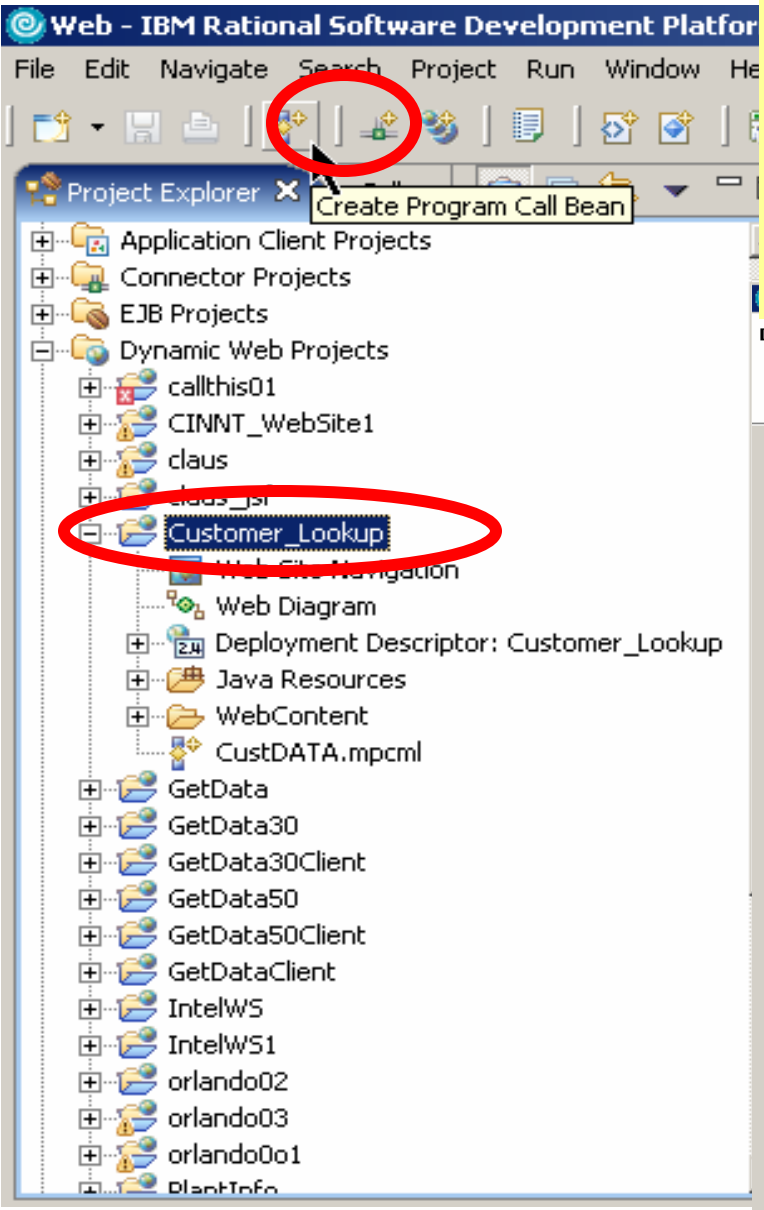


Create design of application layout
web pages and connections between pages with conditions

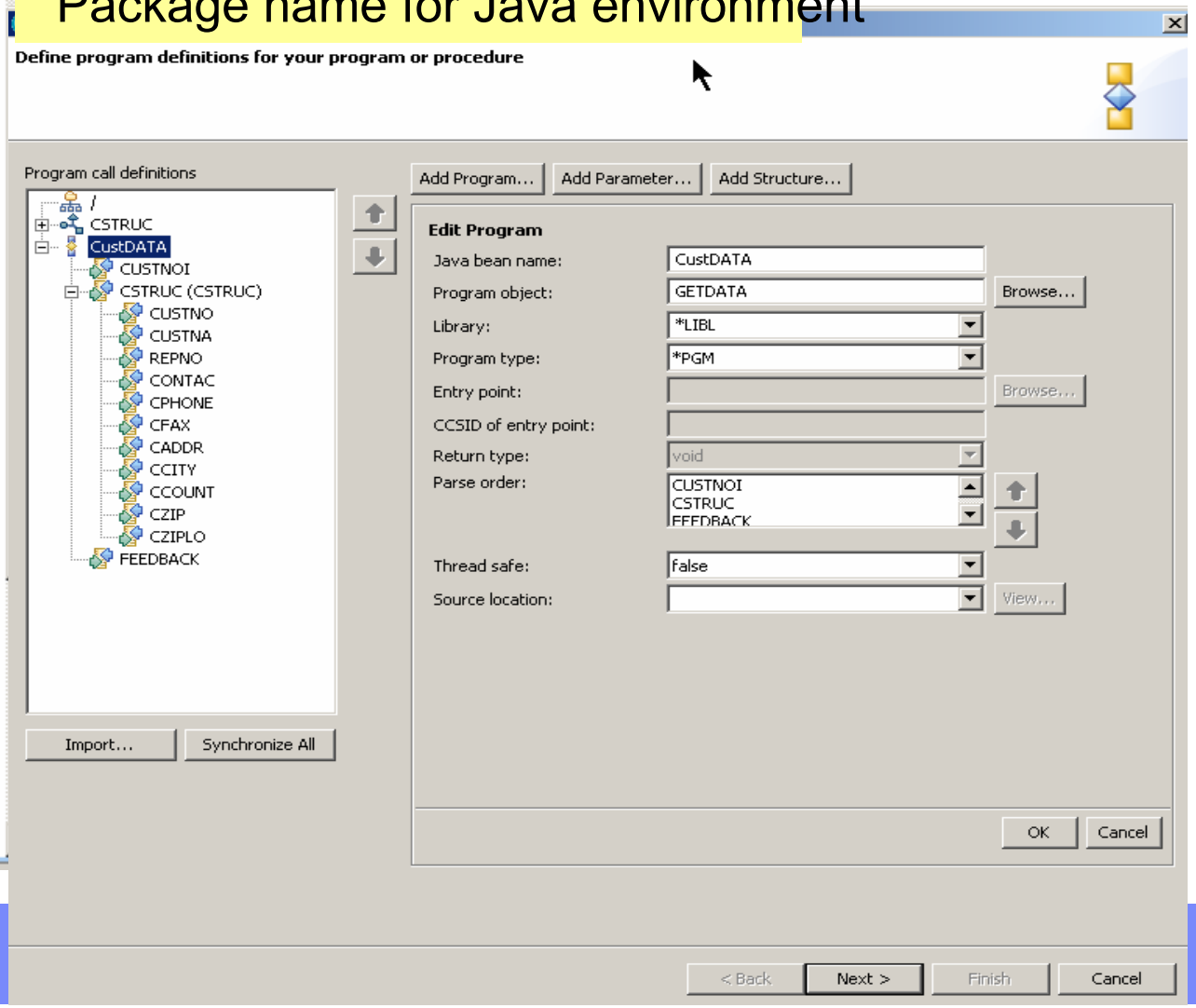
Create webpages from diagram as .jsp files



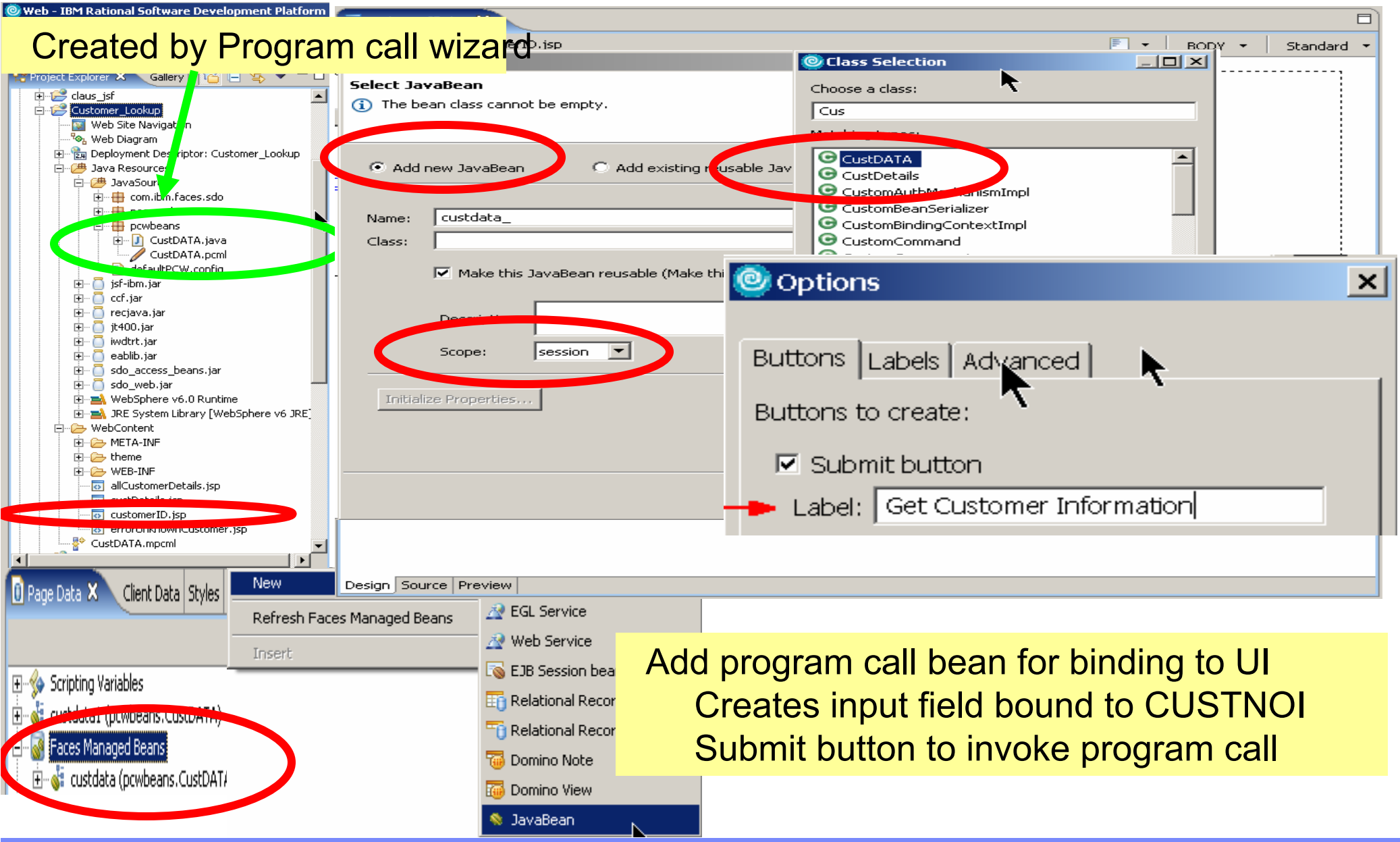
Create program call bean



Select source member
Also create connection info
iSeries name
User id/password
Package name for Java environment



Create Web page design and binding



Add program call bean for binding to UI
- Creates input field bound to CUSTNOI
- Submit button to invoke program call

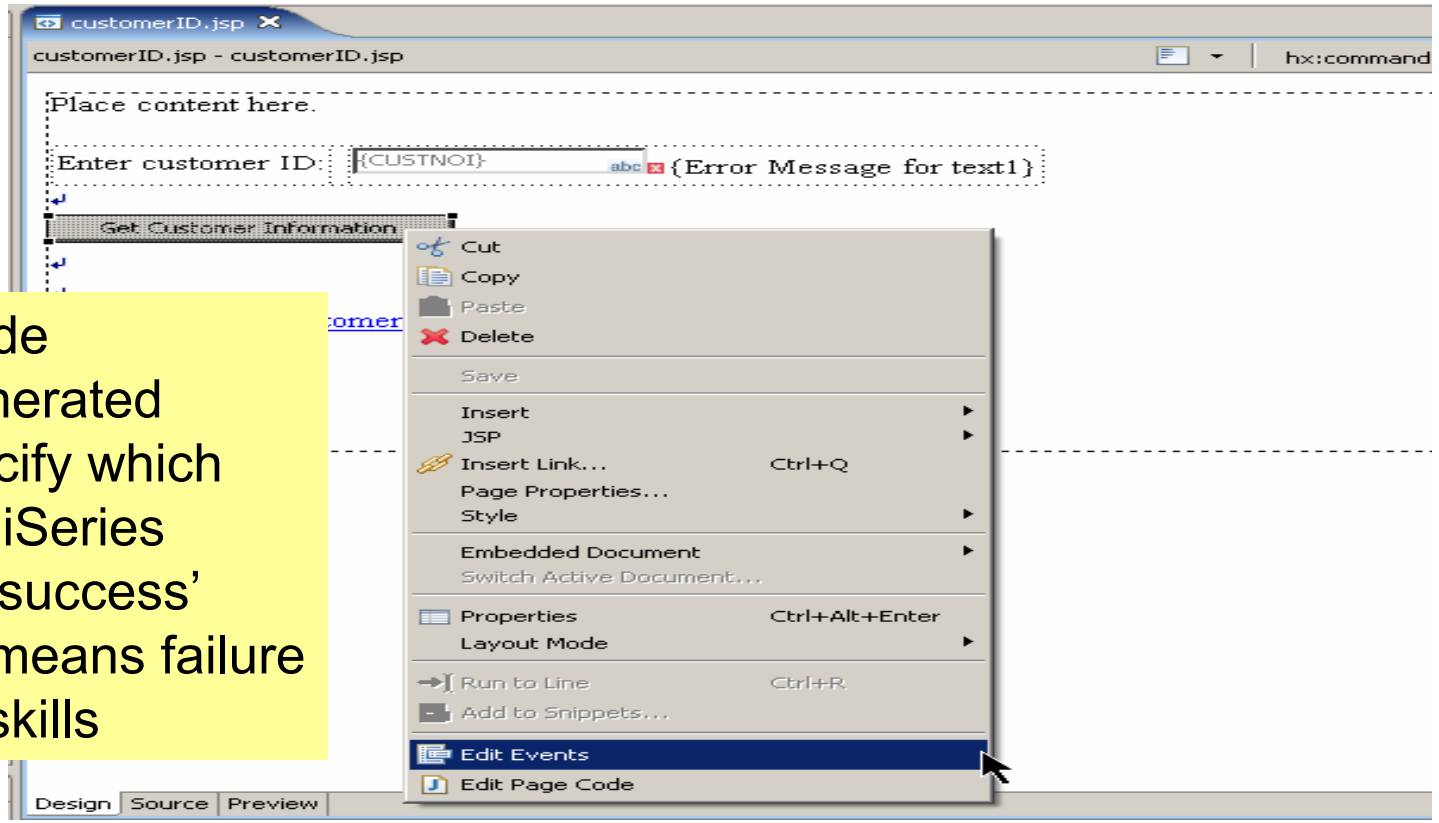
Add rules for submit button and map return code to forward condition

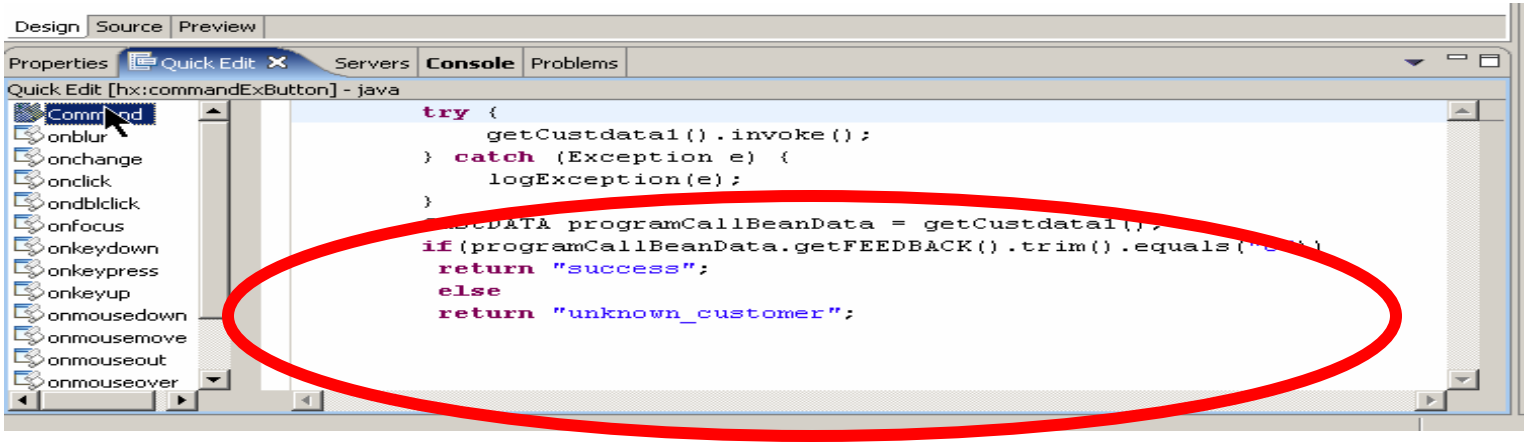
Beginning of code automatically generated

But have to specify which return code from iSeries program means ‘success’

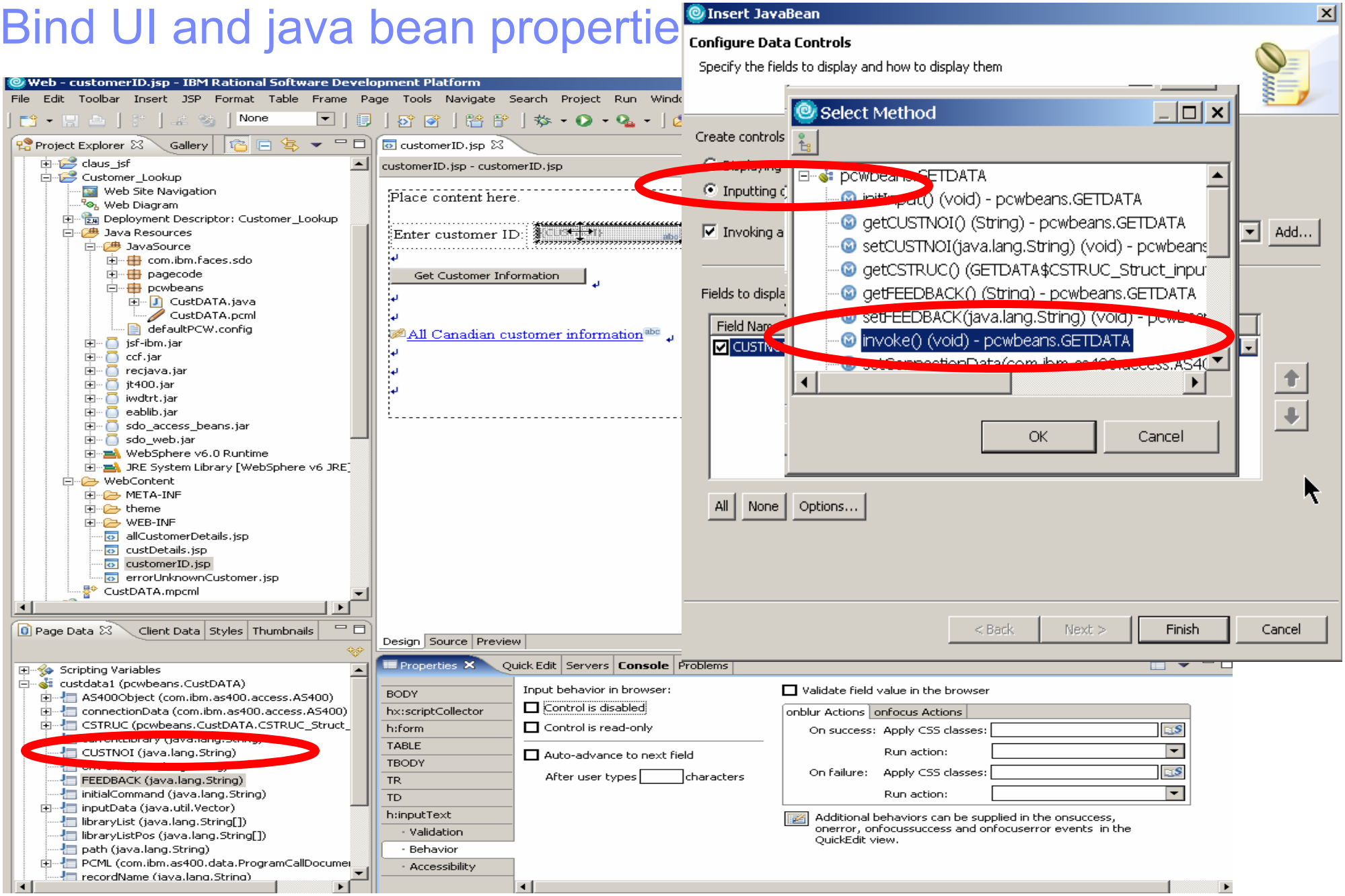
And which one means failure

Very little Java skills required

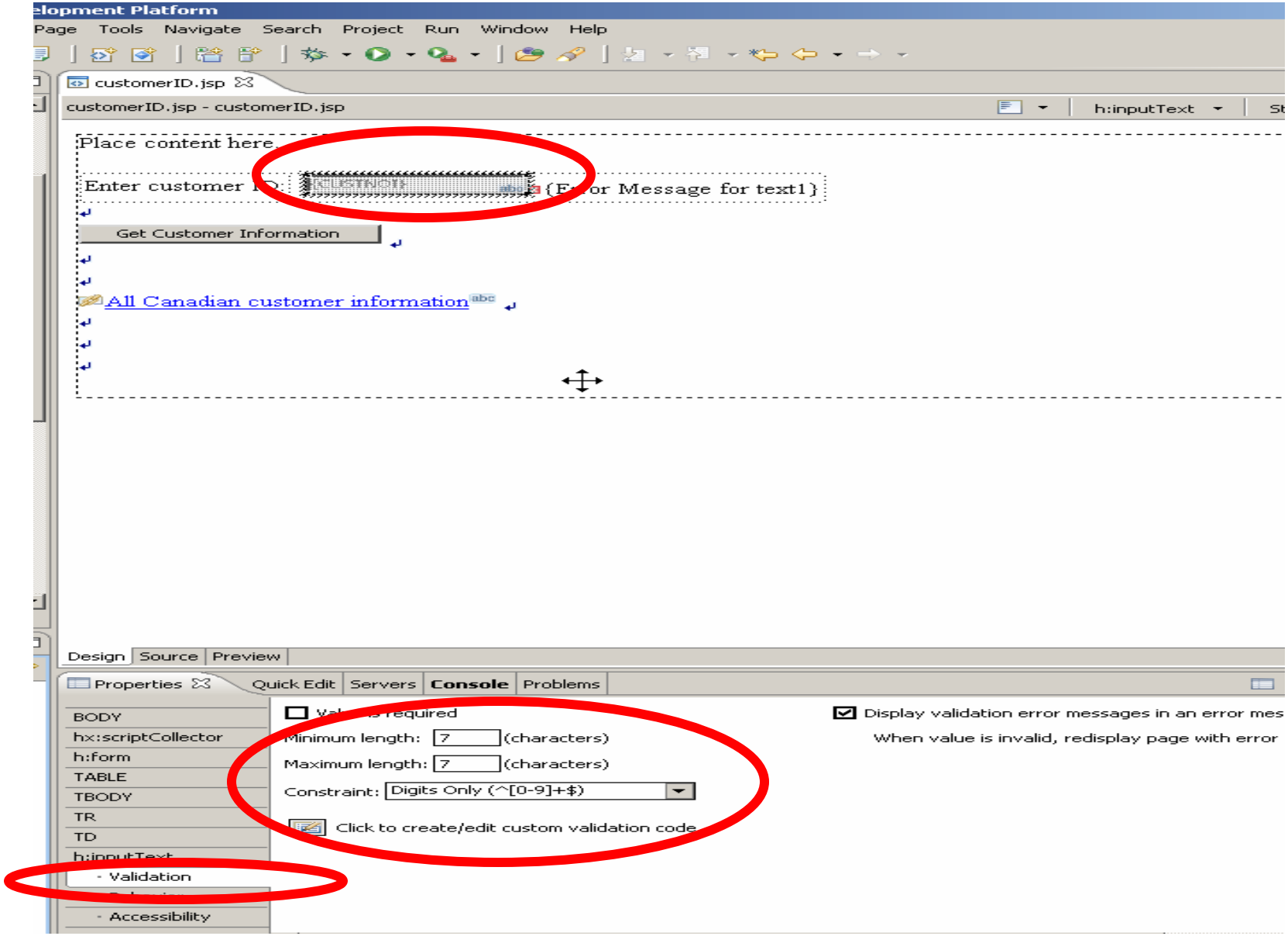




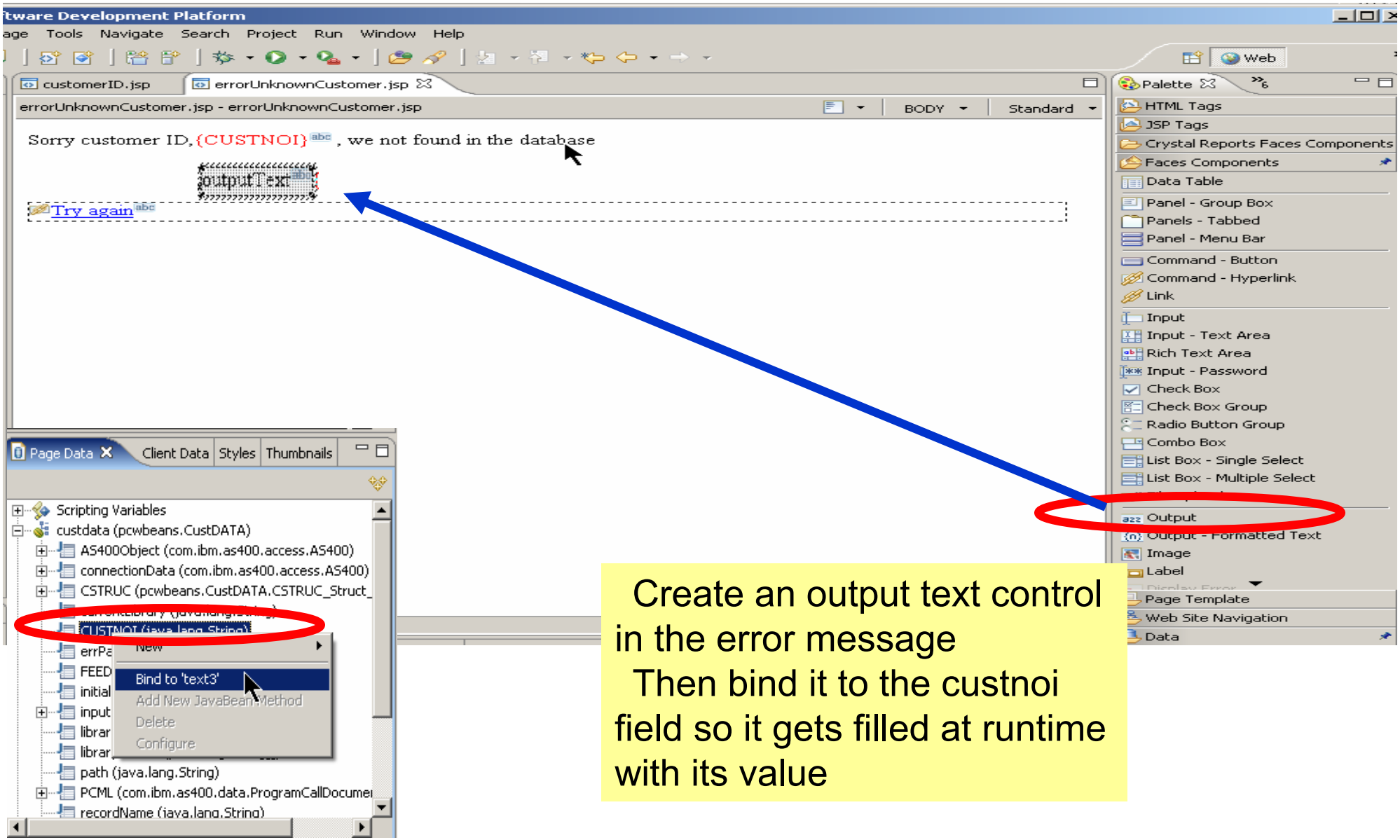
Bind UI and java bean properties



Jsf entry field validation



Creating the error page

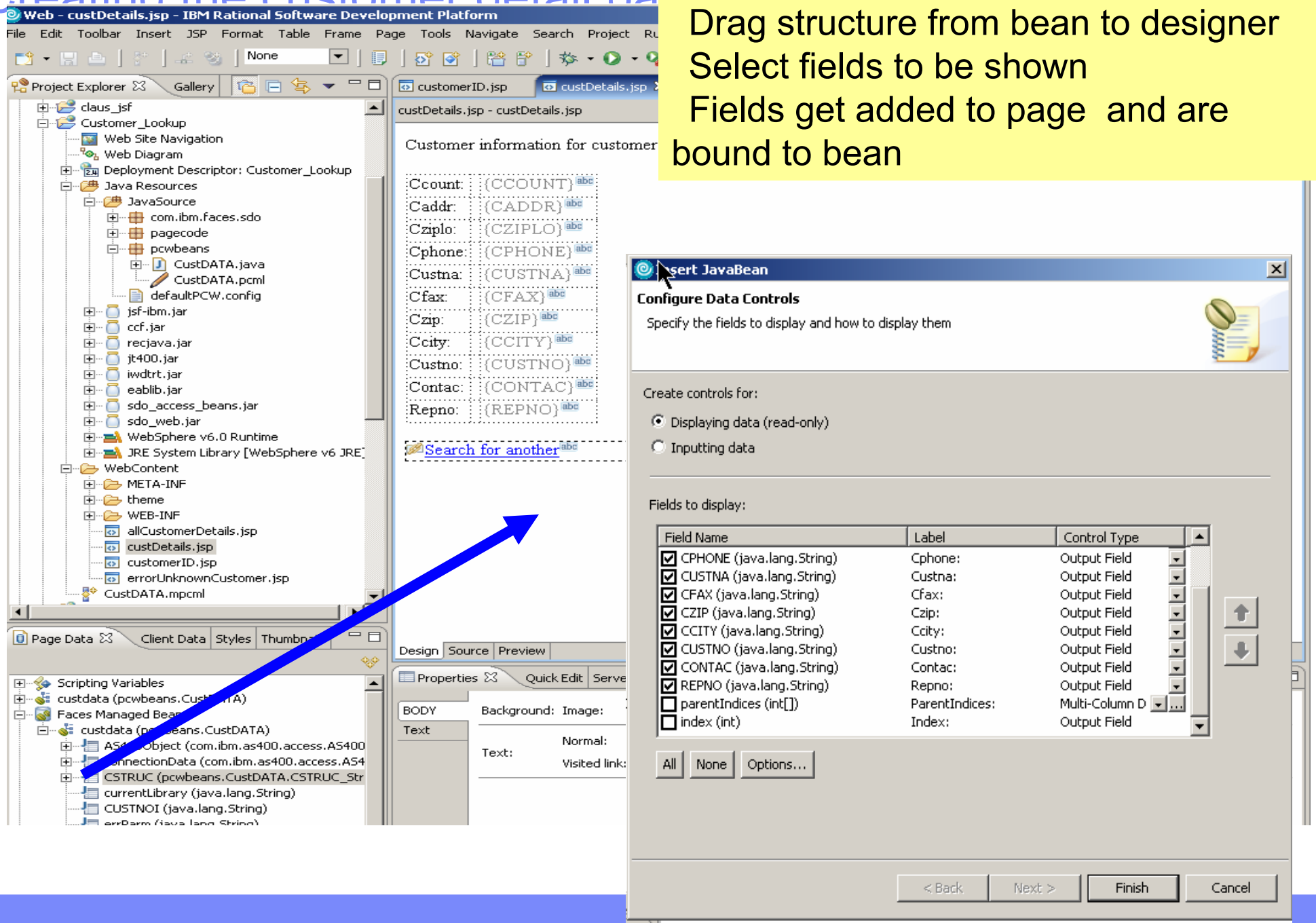


Create an output text control in the error message

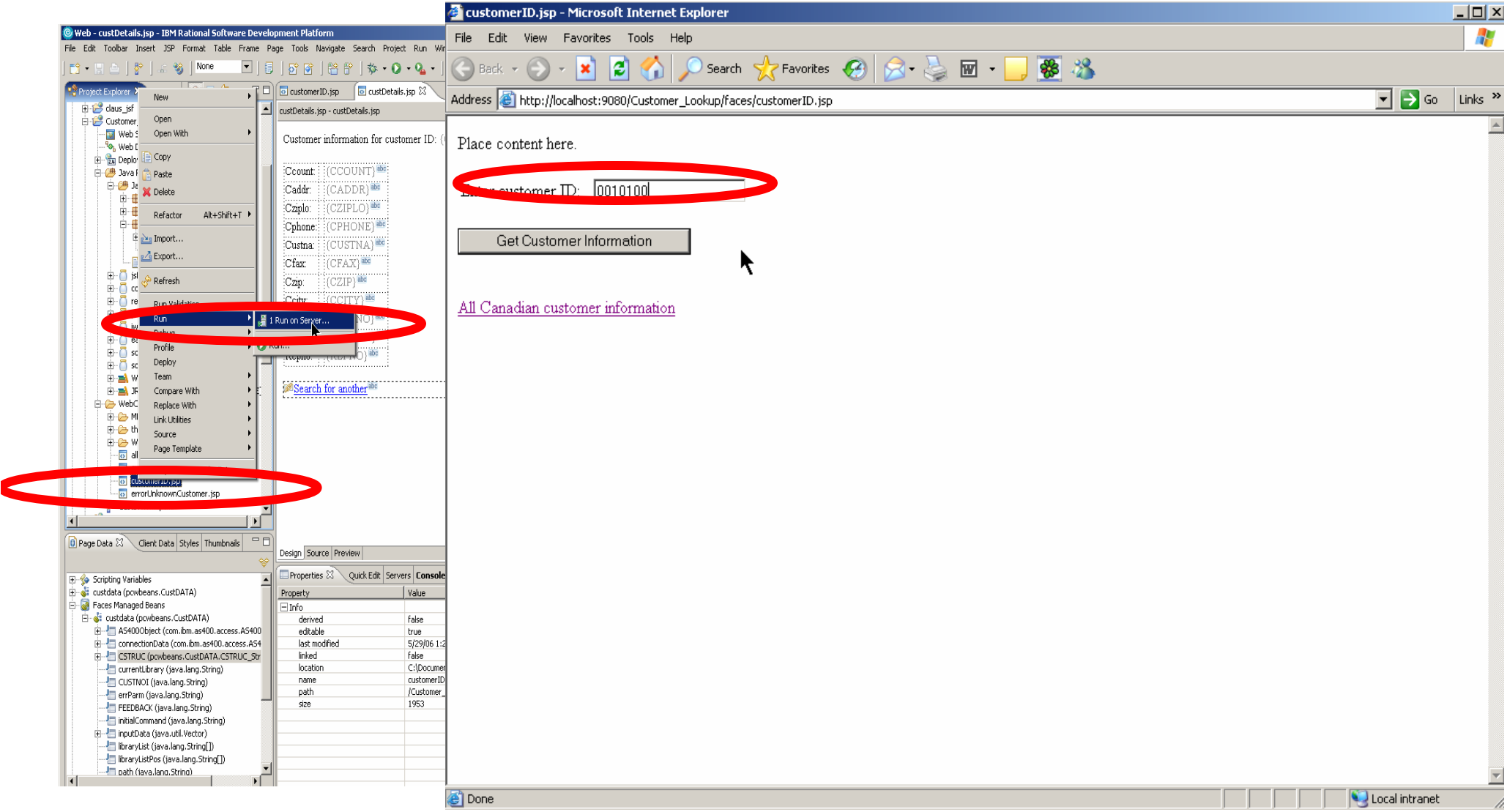
Then bind it to the custnoi field so it gets filled at runtime with its value

Creating the customer detail page

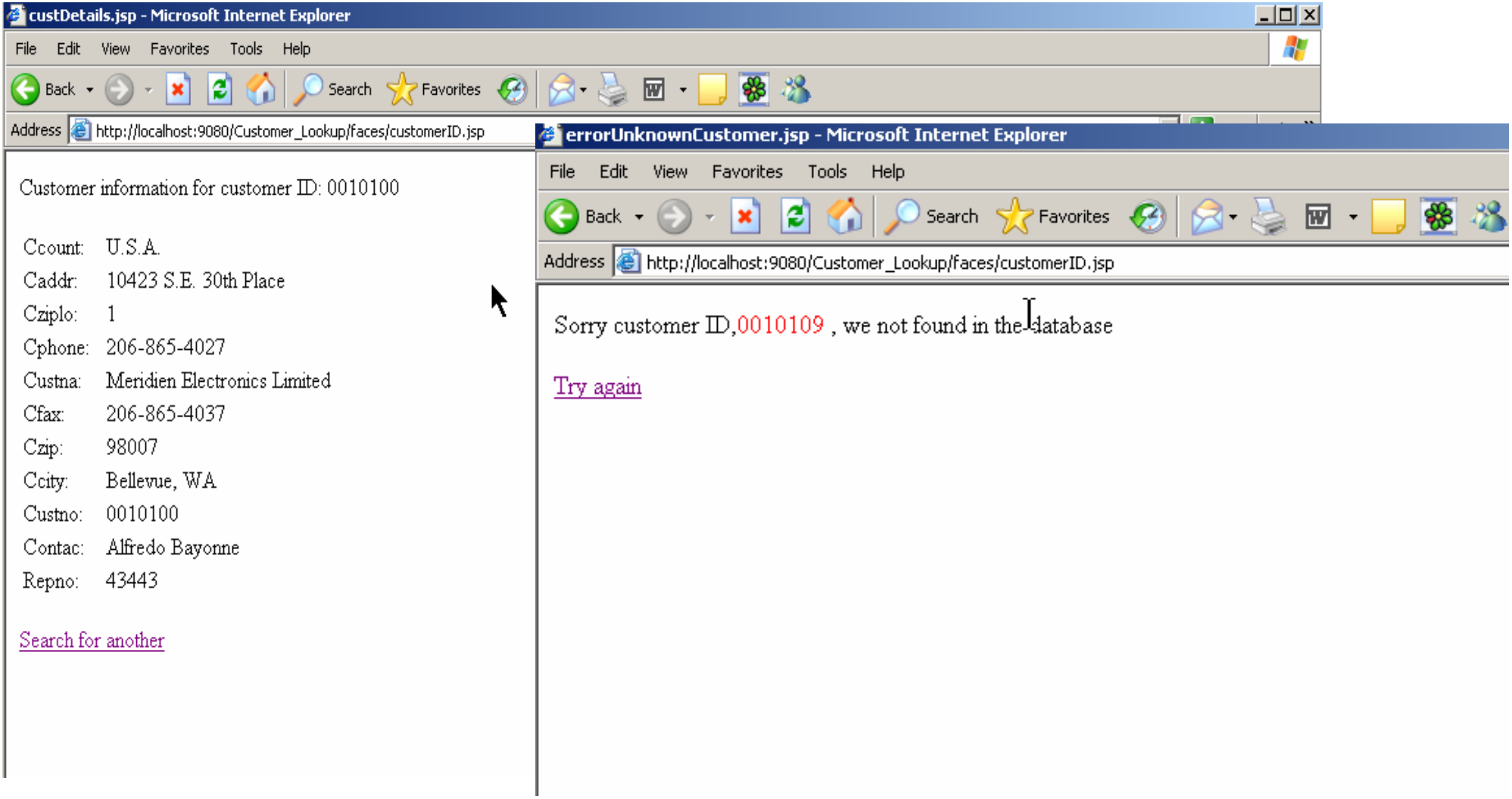
Drag structure from bean to designer
Select fields to be shown
Fields get added to page and are bound to bean



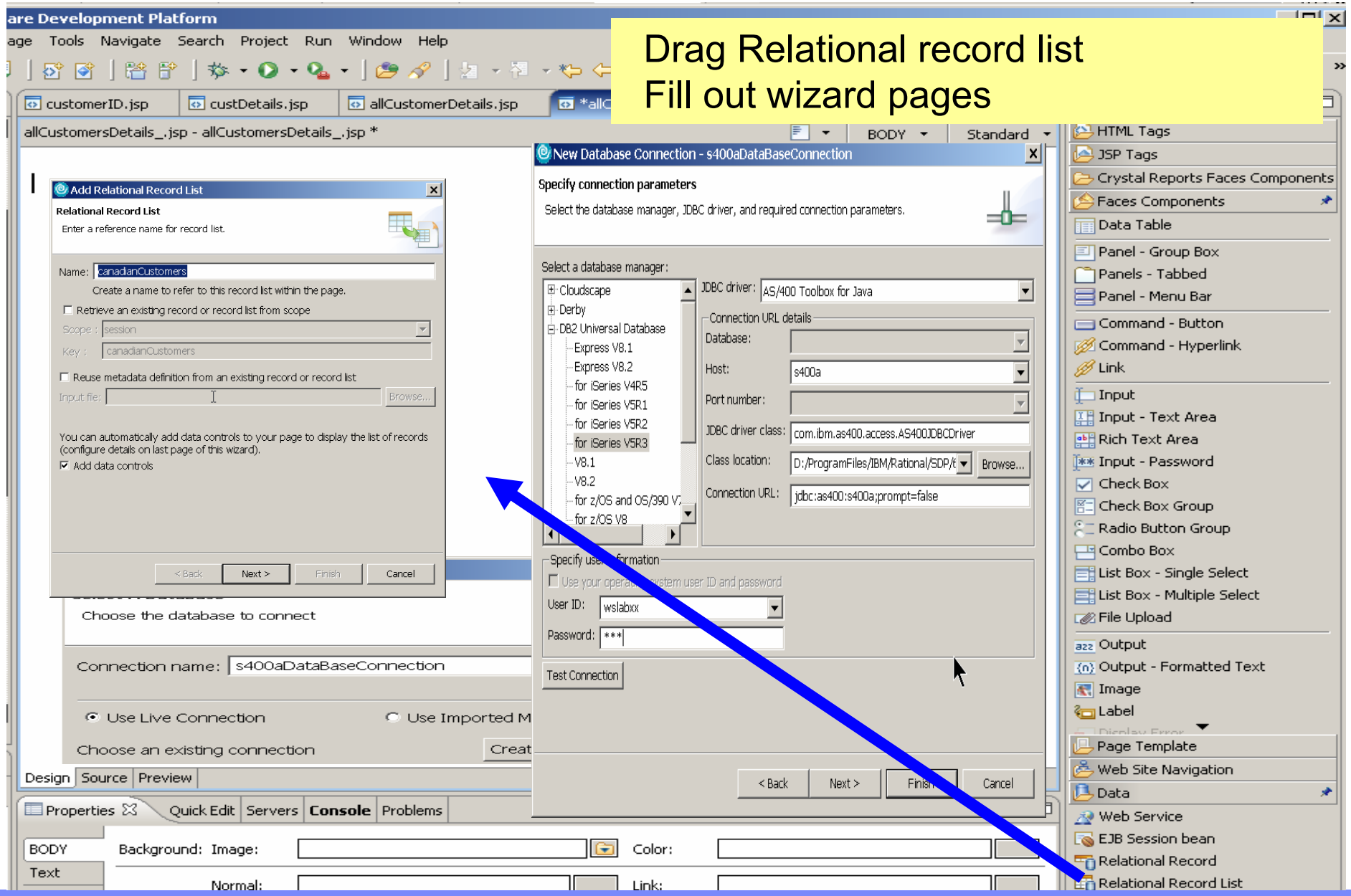
Run it



Result Detail and error pages

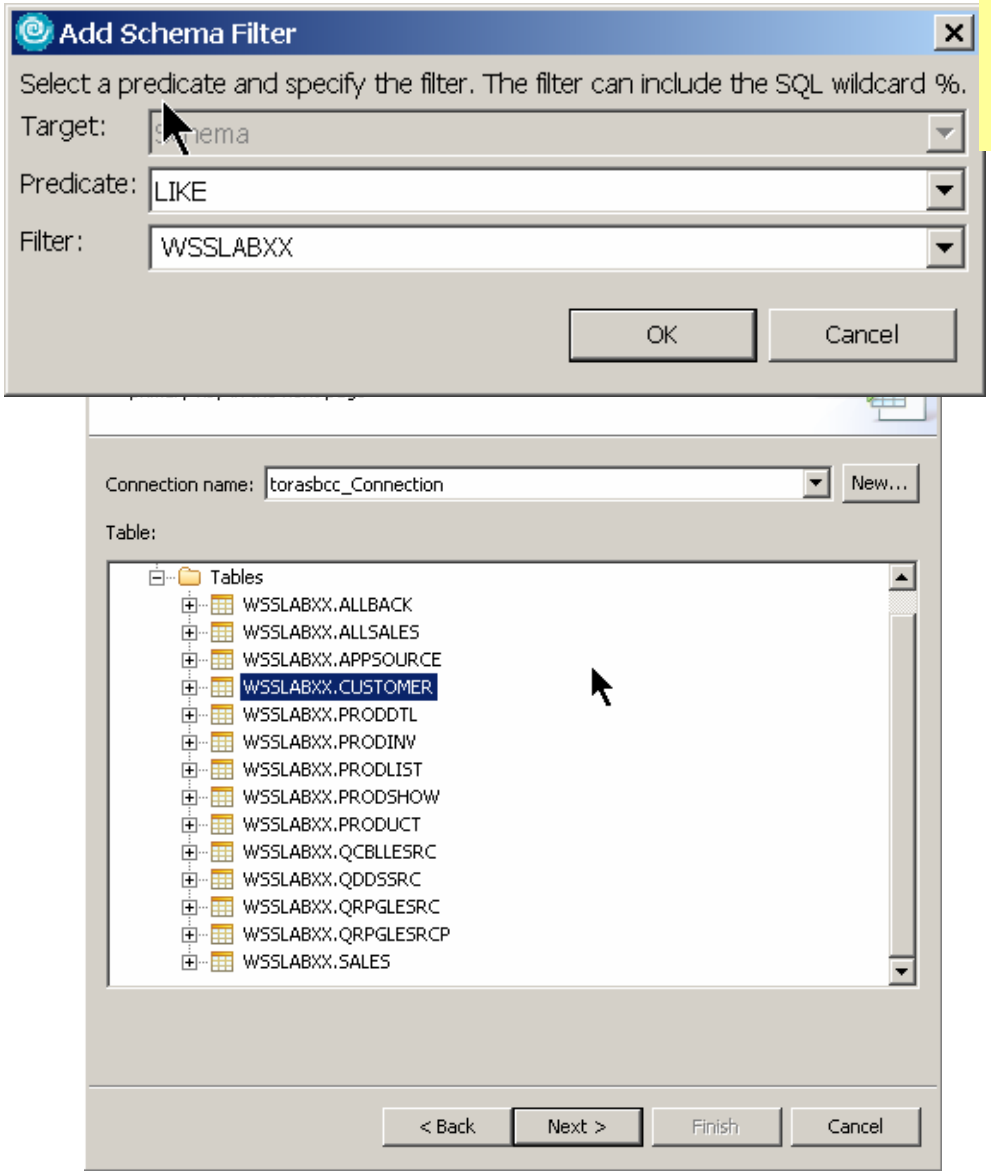


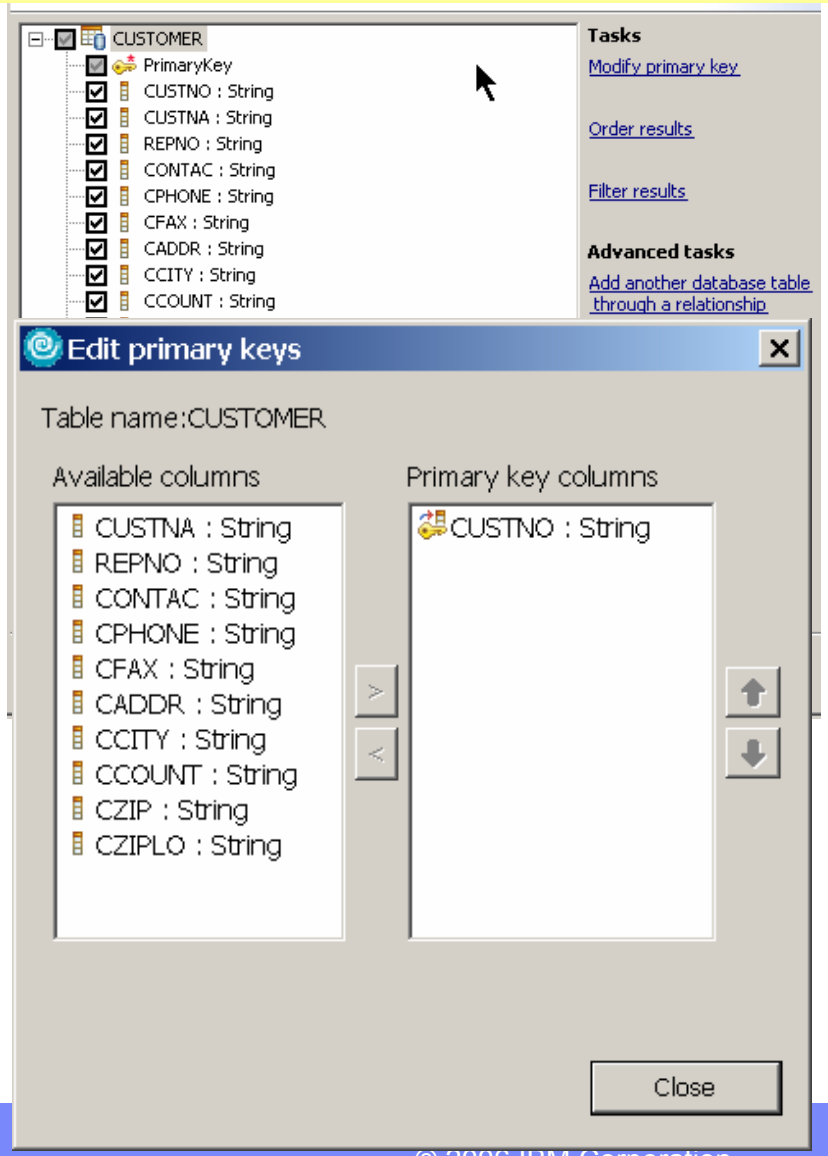
Display list of records using Relational record list control



Identify iSeries data base file

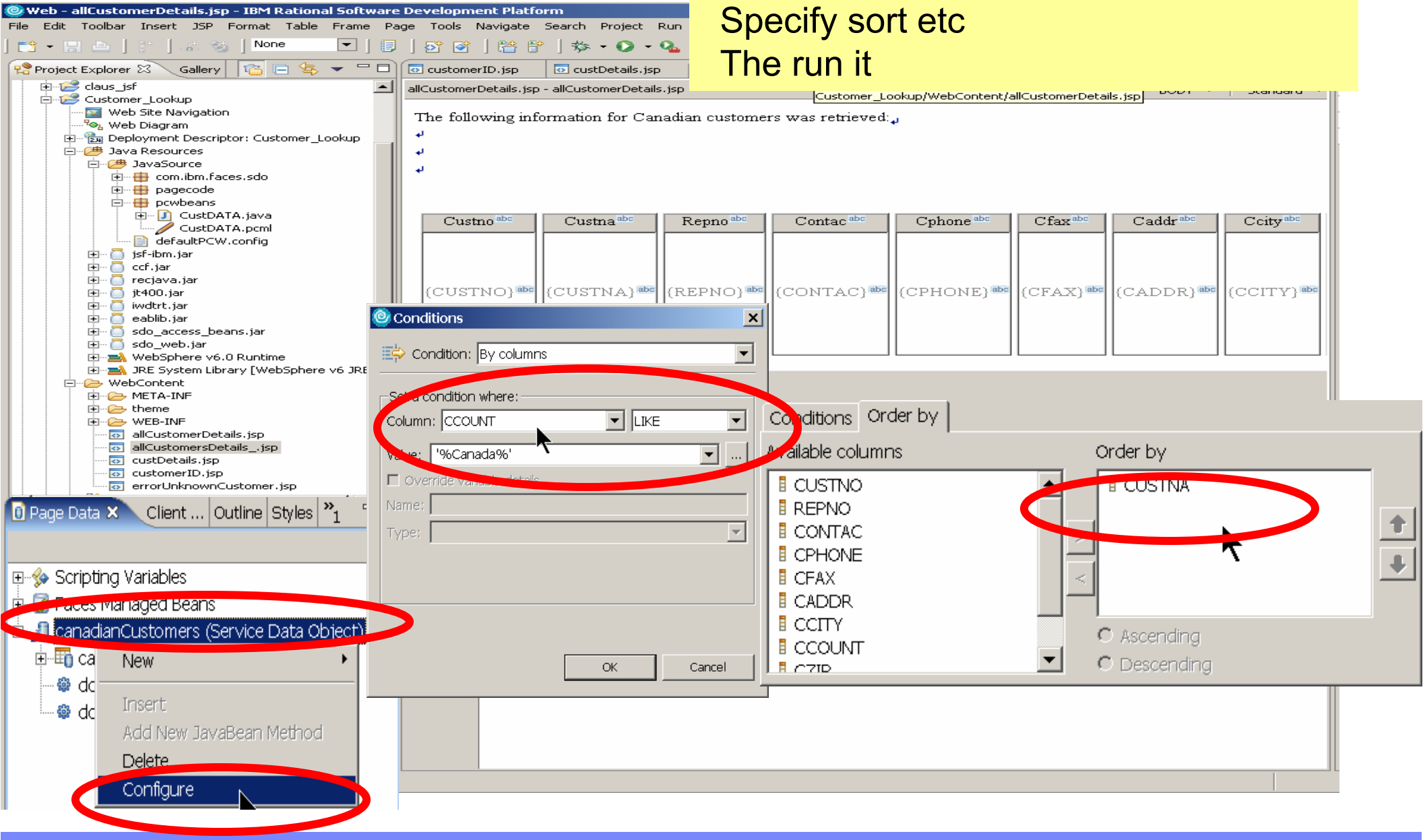
Filter for library
Select database file
Specify primary key
Almost done



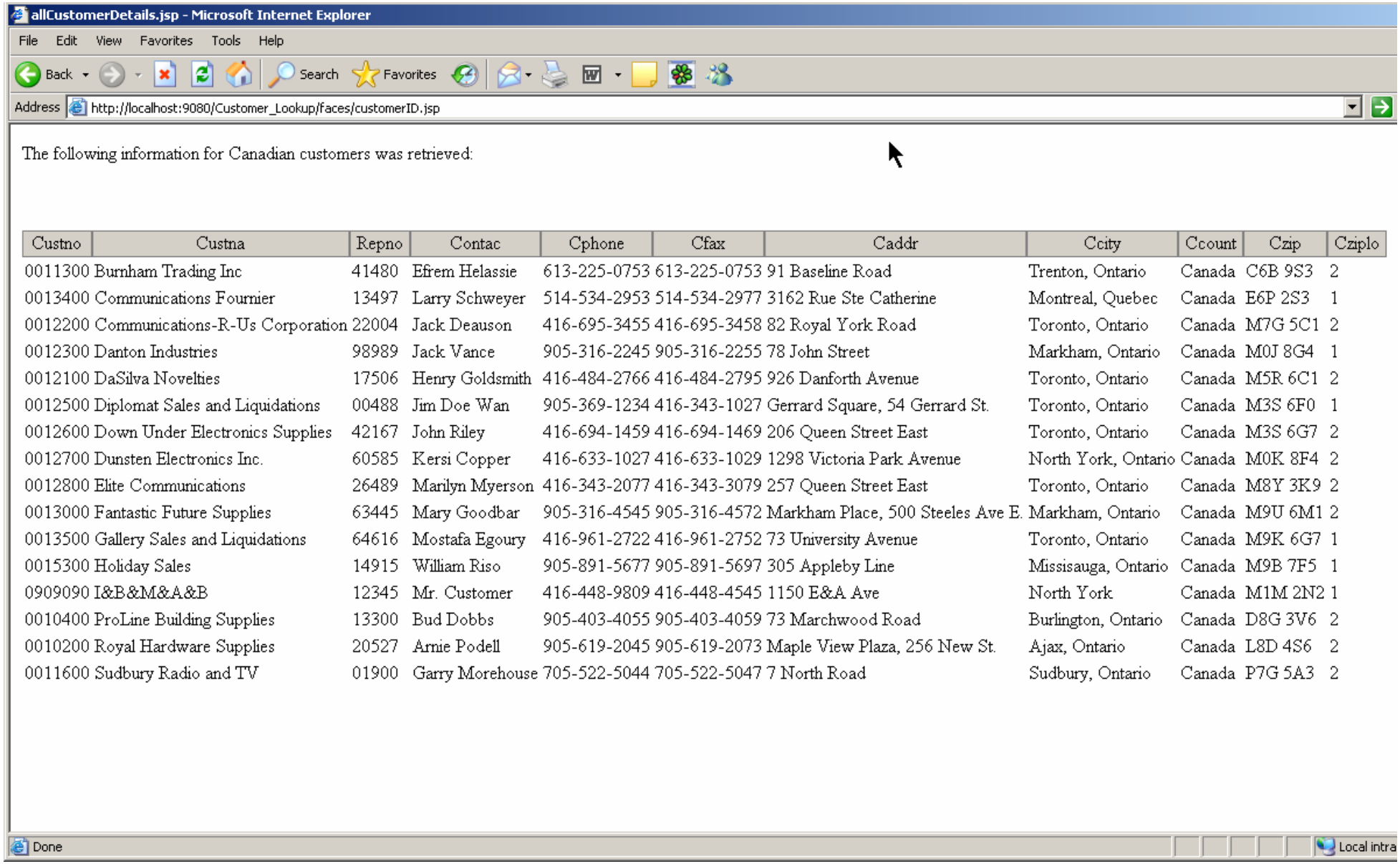


Specify the data to be shown

List page has been created
Specify SQL Query conditons
Specify sort etc
The run it



Running the query



Custno	Custna	Repno	Contac	Cphone	Cfax	Caddr	Ccity	Ccount	Czip	Cziplo
0011300	Burnham Trading Inc	41480	Efrem Helassie	613-225-0753	613-225-0753	91 Baseline Road	Trenton, Ontario	Canada	C6B 9S3	2
0013400	Communications Fournier	13497	Larry Schweyer	514-534-2953	514-534-2977	3162 Rue Ste Catherine	Montreal, Quebec	Canada	E6P 2S3	1
0012200	Communications-R-Us Corporation	22004	Jack Deauson	416-695-3455	416-695-3458	82 Royal York Road	Toronto, Ontario	Canada	M7G 5C1	2
0012300	Danton Industries	98989	Jack Vance	905-316-2245	905-316-2255	78 John Street	Markham, Ontario	Canada	M0J 8G4	1
0012100	DaSilva Novelties	17506	Henry Goldsmith	416-484-2766	416-484-2795	926 Danforth Avenue	Toronto, Ontario	Canada	M5R 6C1	2
0012500	Diplomat Sales and Liquidations	00488	Jim Doe Wan	905-369-1234	416-343-1027	Gerrard Square, 54 Gerrard St.	Toronto, Ontario	Canada	M3S 6F0	1
0012600	Down Under Electronics Supplies	42167	John Riley	416-694-1459	416-694-1469	206 Queen Street East	Toronto, Ontario	Canada	M3S 6G7	2
0012700	Dunsten Electronics Inc.	60585	Kersi Copper	416-633-1027	416-633-1029	1298 Victoria Park Avenue	North York, Ontario	Canada	M0K 8F4	2
0012800	Elite Communications	26489	Marilyn Myerson	416-343-2077	416-343-3079	257 Queen Street East	Toronto, Ontario	Canada	M8Y 3K9	2
0013000	Fantastic Future Supplies	63445	Mary Goodbar	905-316-4545	905-316-4572	Markham Place, 500 Steeles Ave E.	Markham, Ontario	Canada	M9U 6M1	2
0013500	Gallery Sales and Liquidations	64616	Mostafa Egoury	416-961-2722	416-961-2752	73 University Avenue	Toronto, Ontario	Canada	M9K 6G7	1
0015300	Holiday Sales	14915	William Riso	905-891-5677	905-891-5697	305 Appleby Line	Missisauga, Ontario	Canada	M9B 7F5	1
0909090	I&B&M&A&B	12345	Mr. Customer	416-448-9809	416-448-4545	1150 E&A Ave	North York	Canada	M1M 2N2	1
0010400	ProLine Building Supplies	13300	Bud Dobbs	905-403-4055	905-403-4059	73 Marchwood Road	Burlington, Ontario	Canada	D8G 3V6	2
0010200	Royal Hardware Supplies	20527	Arnie Podell	905-619-2045	905-619-2073	Maple View Plaza, 256 New St.	Ajax, Ontario	Canada	L8D 4S6	2
0011600	Sudbury Radio and TV	01900	Garry Morehouse	705-522-5044	705-522-5047	7 North Road	Sudbury, Ontario	Canada	P7G 5A3	2

WebSphere Test Environment

A full copy of WAS 6.0 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

WebTools agenda

- WDS overview and AD roadmap
- Website creation
- Webpage templates
- Designer for static WebPages
- Designer for Web objects
- Tool for Cascading Style Sheets
- Tools to create WebApplications
  - Interaction wizard
  - Java server faces tooling
- Tools to create WebServices

What Are Web Services

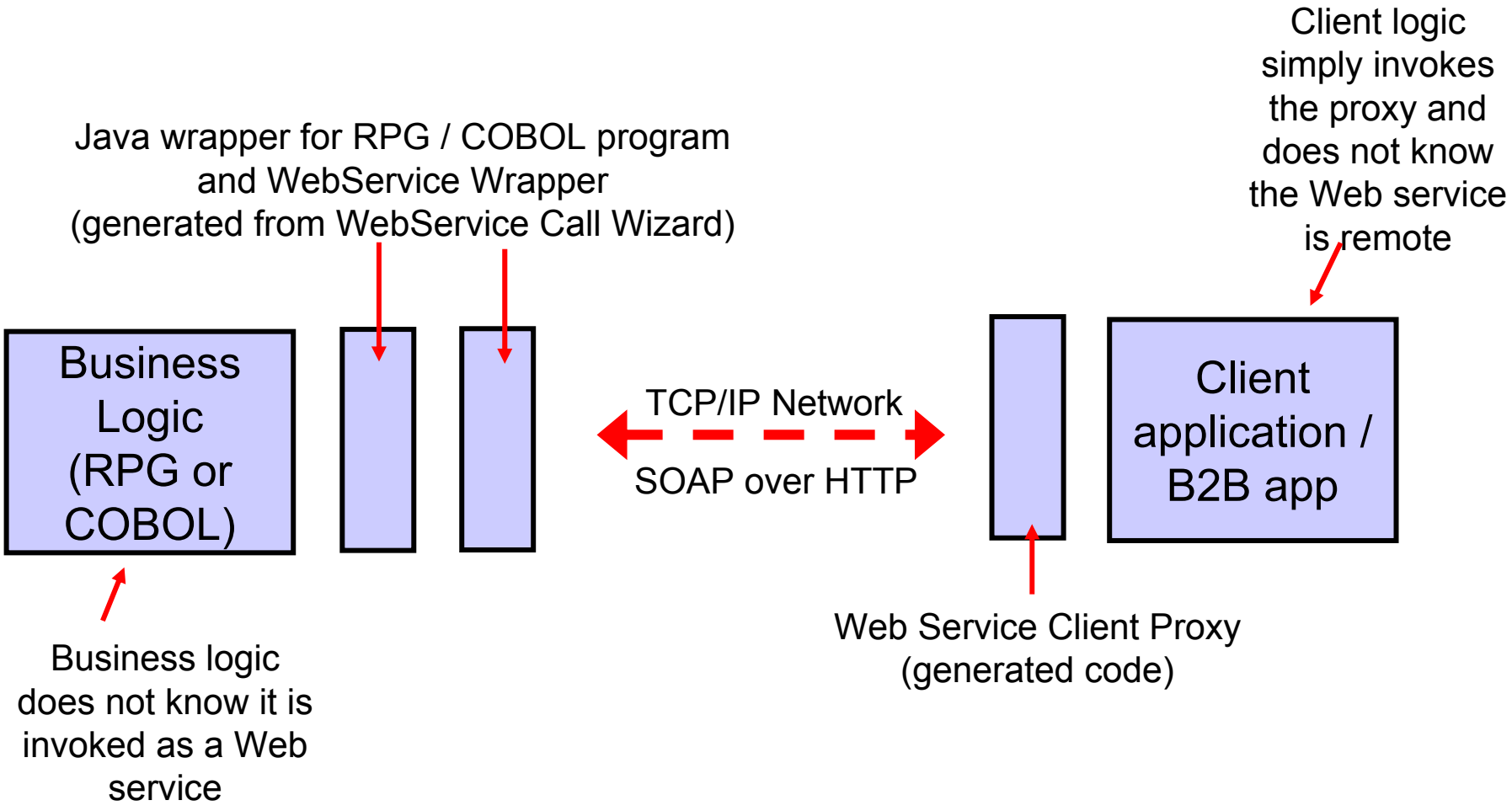
- Function that can be programmatically invoked over a network

- Basically remote procedure calls built on open standards and proven technologies
  - Lots of new standards around Web services to ensure interoperability in heterogeneous environments
  - Underlying implementations built on proven technologies like
    - XML
    - HTTP
    - Messaging middleware
    - Security standards

Why use Web Services

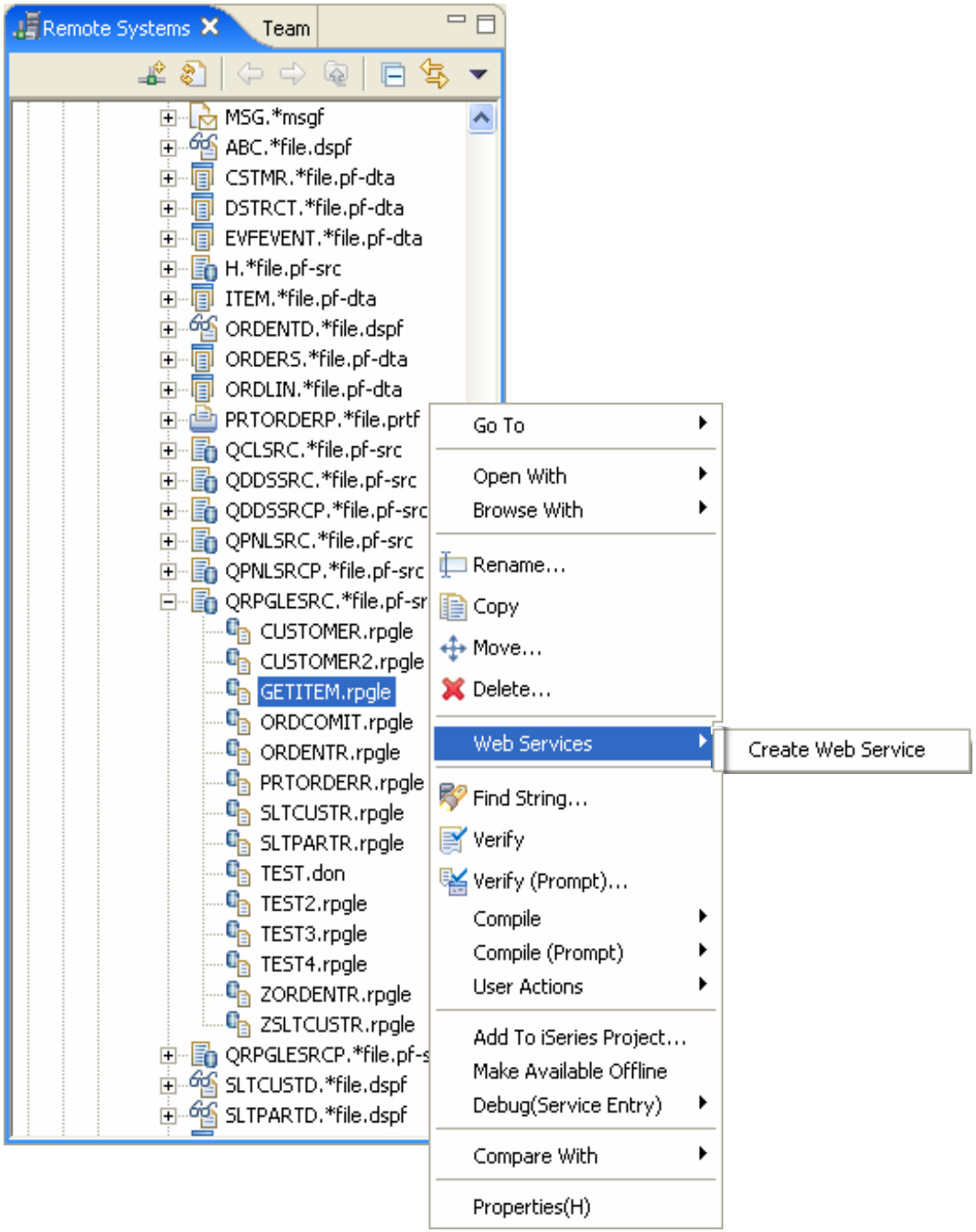
- Because:
  - They are the latest hype
  - They are hot
  - They are cool
  - They are useful
  - Somebody asks you to provide one for a certain task
  - Somebody provides one and asks you to use it

RPG/COBOL Program as Service Provider



System i5 Web Services Tools (new in V6.0.1)

- Single wizard to convert i5/OS program or service program procedure into a Web Service
  - Generates all required code for deploying Web Service to WebSphere Application Server
- Automatically determines program (or procedure) parameters when invoked from RSE
- Just point to RPGIV/ILE COBOL program source
- Go thru a couple of wizard pages and WSDL for the program or procedure will be generated



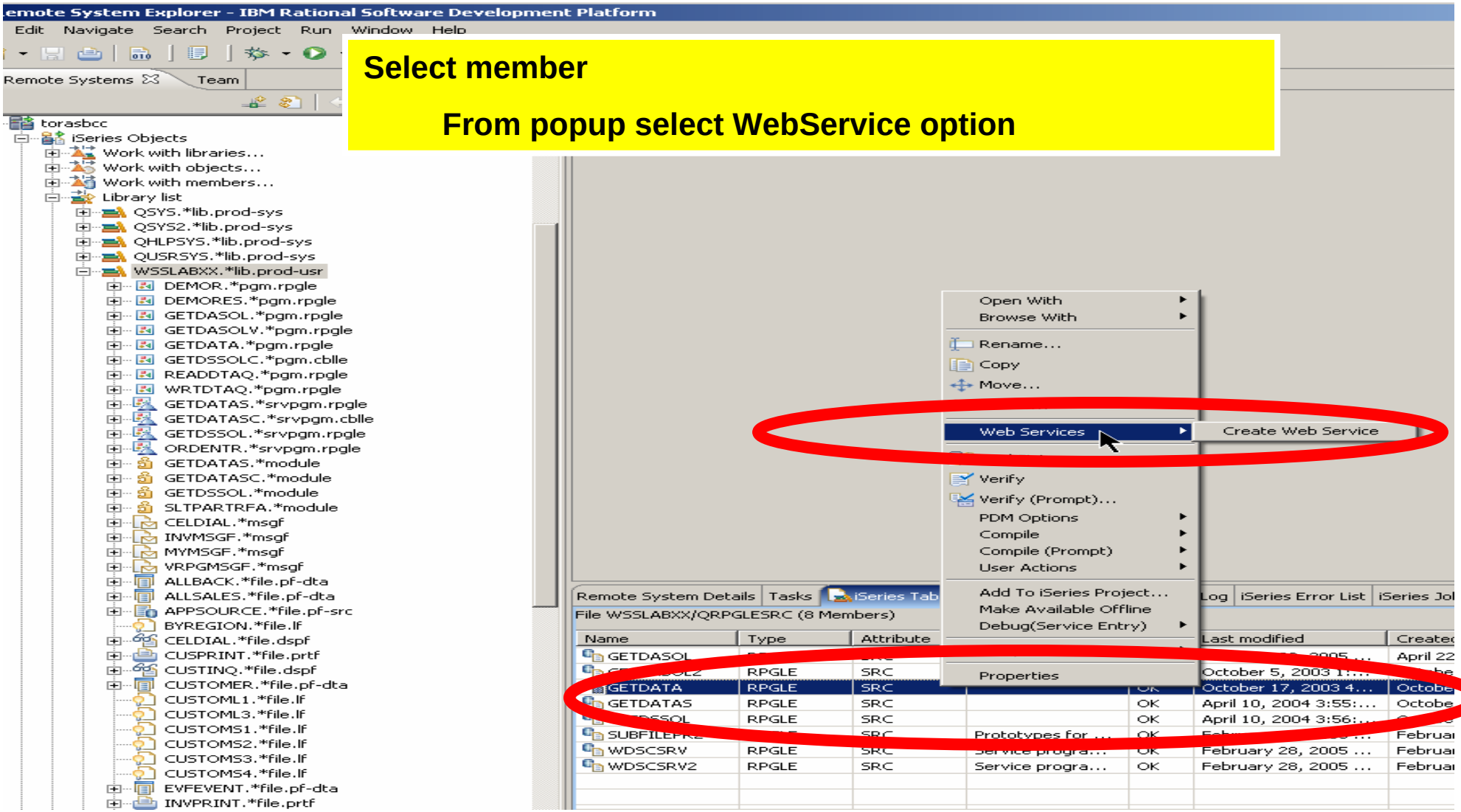
Tools

- **Web Services Tools in WDSc**
  - **Create**
    - **Web services from:**
      - **ILE RPG/COBOL source**
      - **Java Beans**
      - **Enterprise JavaBeans (advanced edition only)**
      - **SQL Queries**
      - **DB2 Stored Procedures**
      - **DB2 XML Extender calls**
    - **Web service client proxy given a WSDL document**
  - **Test**
    - **Test your Web services in the WebSphere test environment**
    - **Generate test code to test generated Web service client proxies**
  - **TCP/IP Monitor**
    - **Useful for debugging SOAP messages between provider and consumer**

WebService Wizard
Use Remote Systems Explorer to select member

Select member

From popup select WebService option



Web Service Wizard

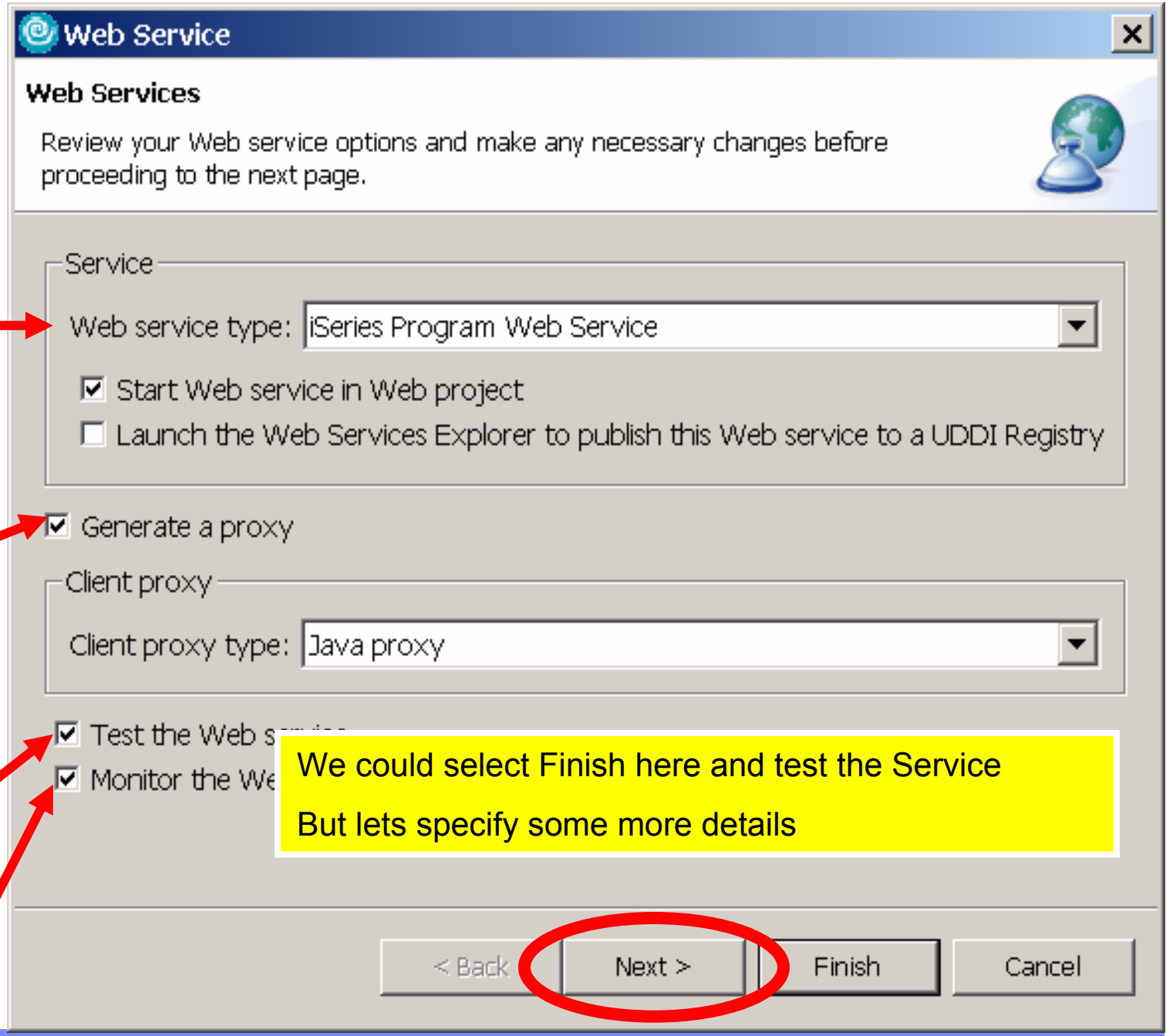
- Click Finish & Done!
- Drop-Dead Simple!

Default is iSeries Web Service type

The Java bean proxy will provide a remote procedure call interface to the Web service

We will test the Web service after it is created

We will monitor the SOAP traffic for this Web service



Configuring the Web Service- Edit Program/Parameters

-File name shows the Program source file the wizard was launched from (valid types incl. ILE RPG, COBOL, PCML)

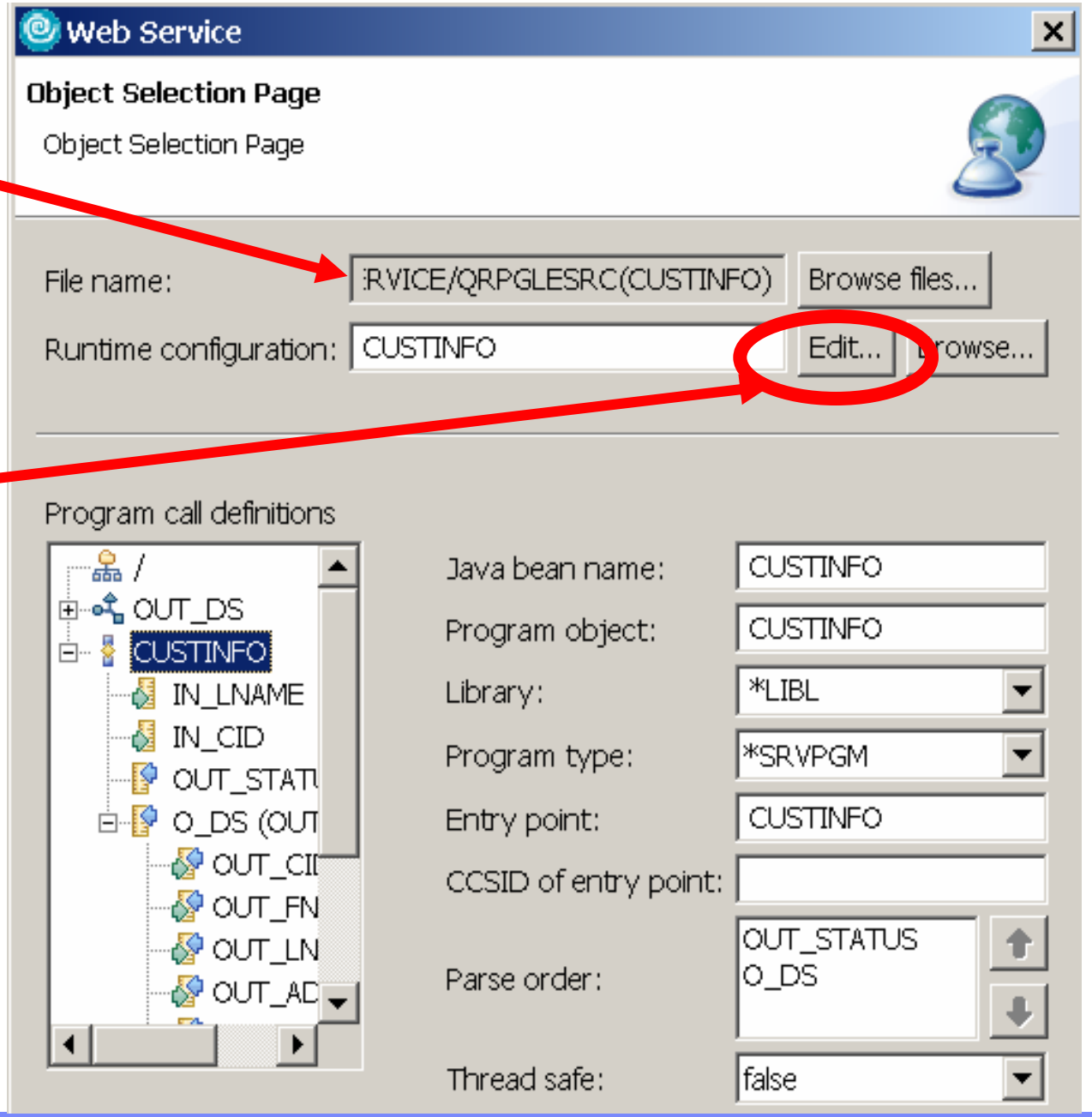
-**Browse files** to select a different file

-The default Runtime configuration is taken from the System i5 connection

-**Edit** to change the configuration

-**Browse** to choose an existing configuration (.config file)

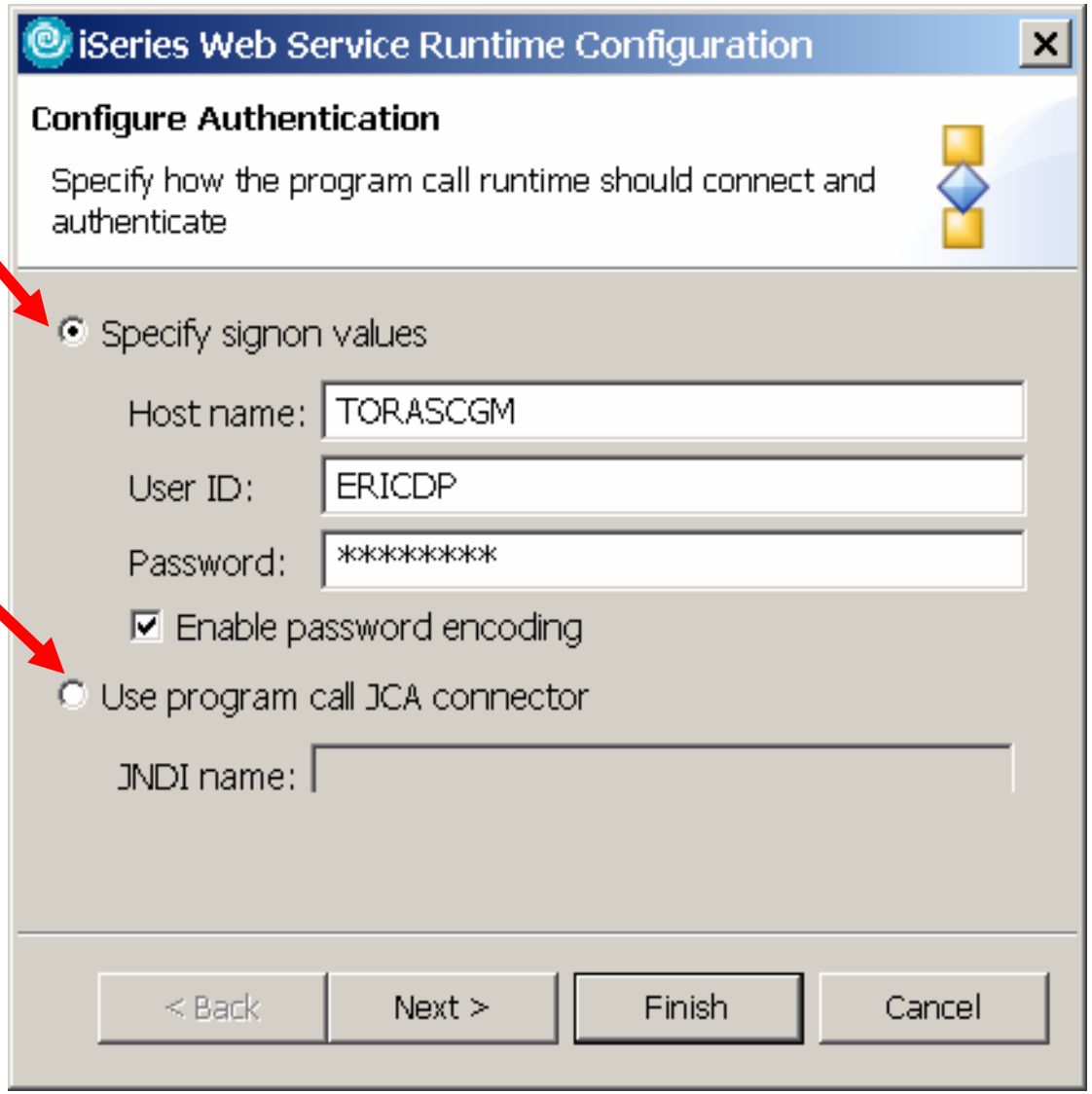
- Selecting the Program (**CUSTINFO**), you can change the Library, Program type, Program Object etc. (in most cases the defaults are sufficient)
- Expand the program to see program parameters
- Selecting a parameter (e.g. **IN_CID**), you can change Usage type, CCSID



Configuring the Web Service- Runtime Configuration

Signon information is defaulted from System i5 connection

Alternatively, can use the Program Call Java Connection Architecture connector for signon and connection management (advanced product only)

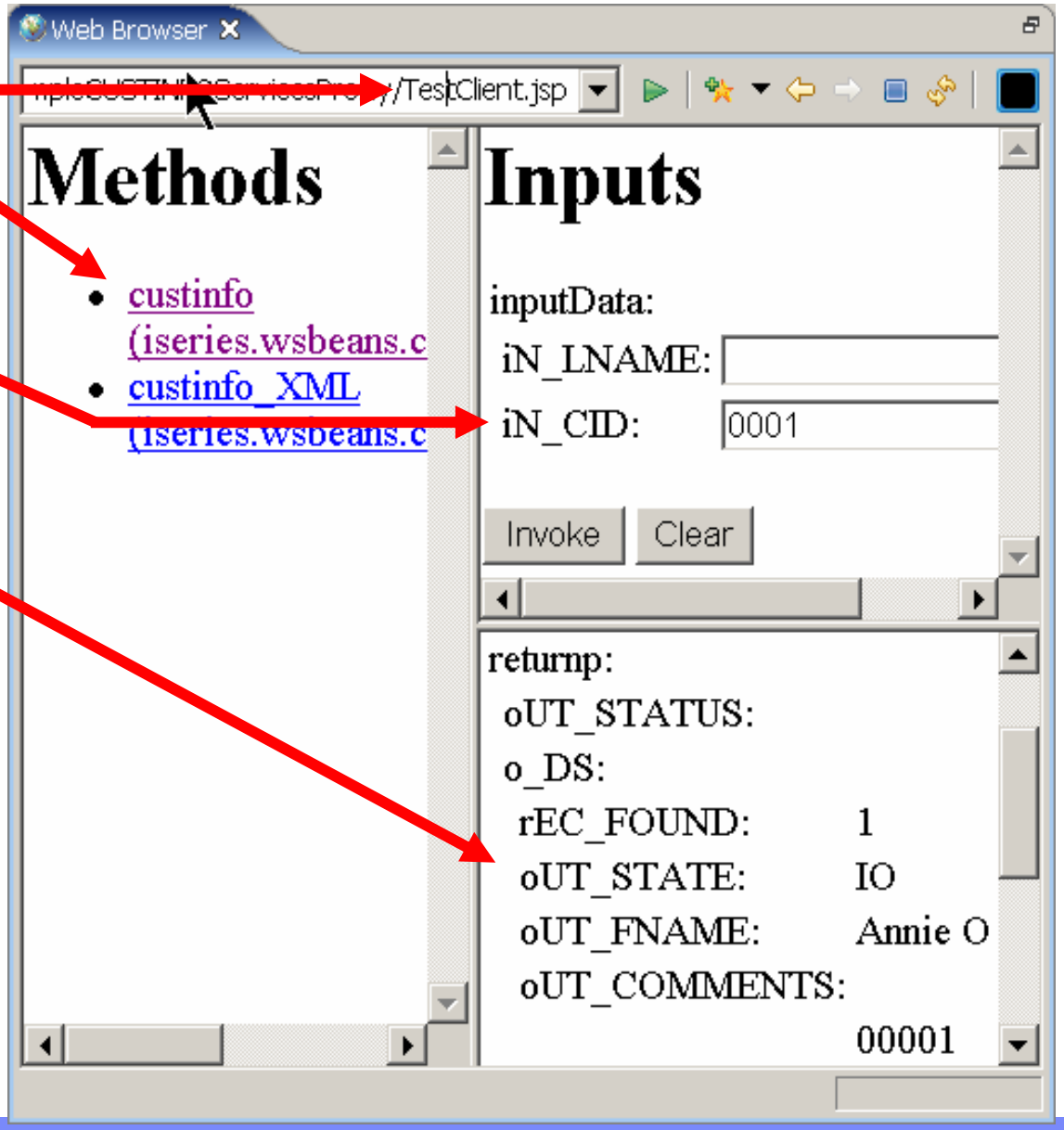


Testing the Web Service

After wizard completes, the generated Test Client JSPs are run on the server (selected option “Test the Web service” in the wizard)

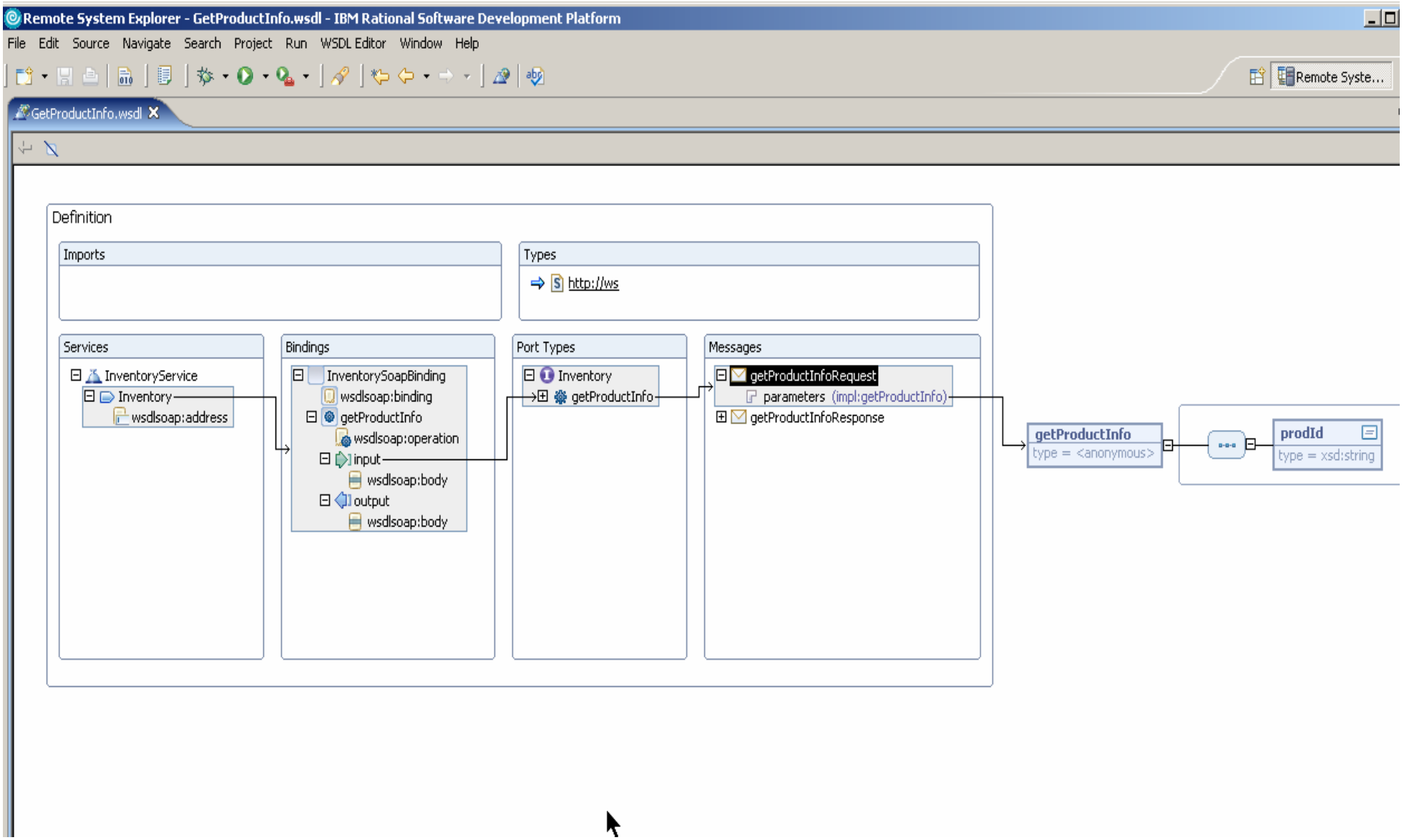
Select one of the methods to test, then enter the input data, Customer ID “0001”

Click “Invoke” and the results of the Web service are displayed, Customer ID “0001” is associated with Annie O!



- As you will see shortly, there are many different ways to test the Web service

WSDL editor in WDSc (graphical view)

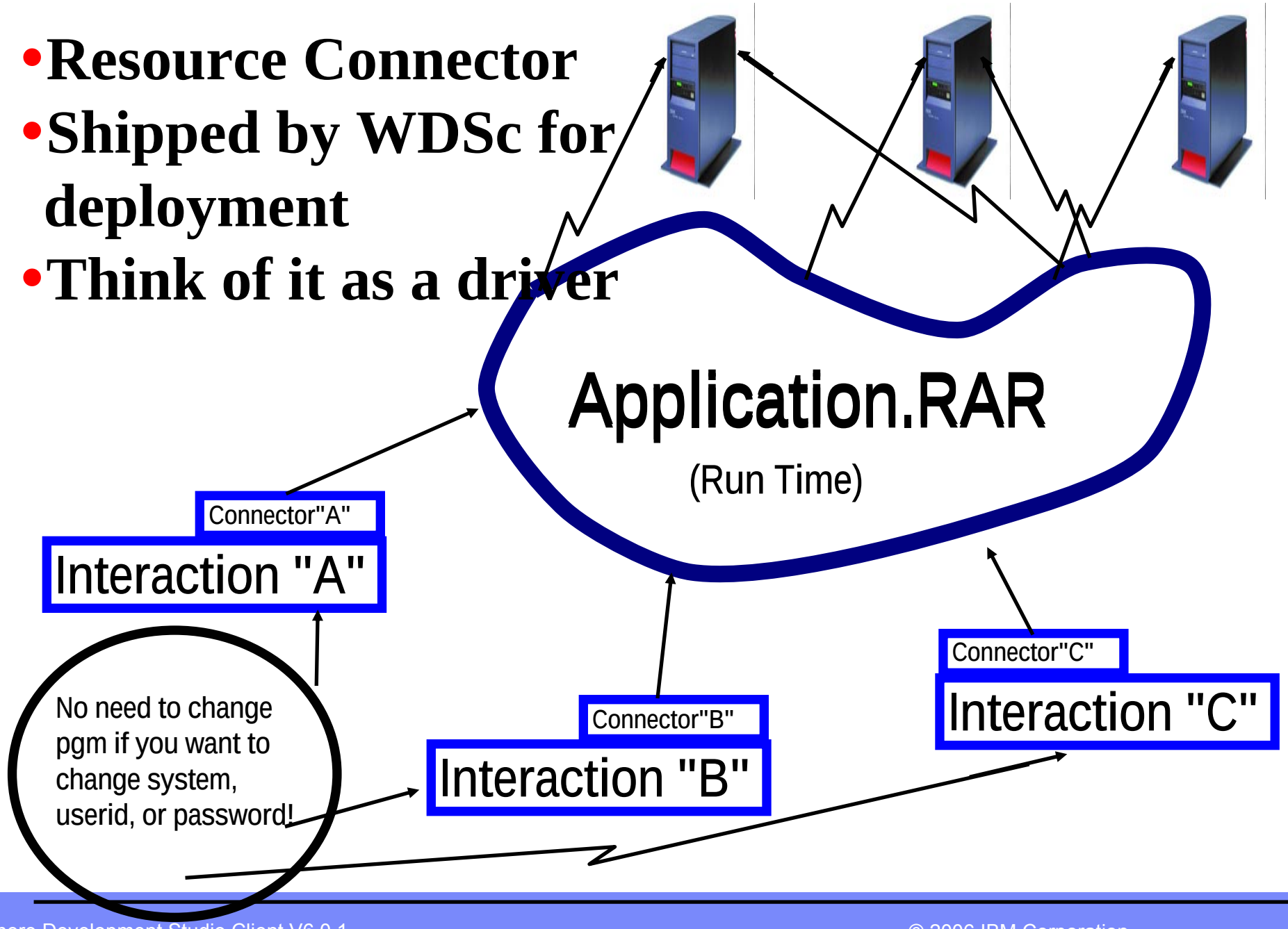


Web Tool for iSeries

- Wizards to help create iSeries RPG/COBOL web applications
  - Interaction wizard to build an RPG web application
  - WebService wizard to build an iSeries WebService
  - JCA wizard to build an iSeries Java Connector Architecture (JCA) connector

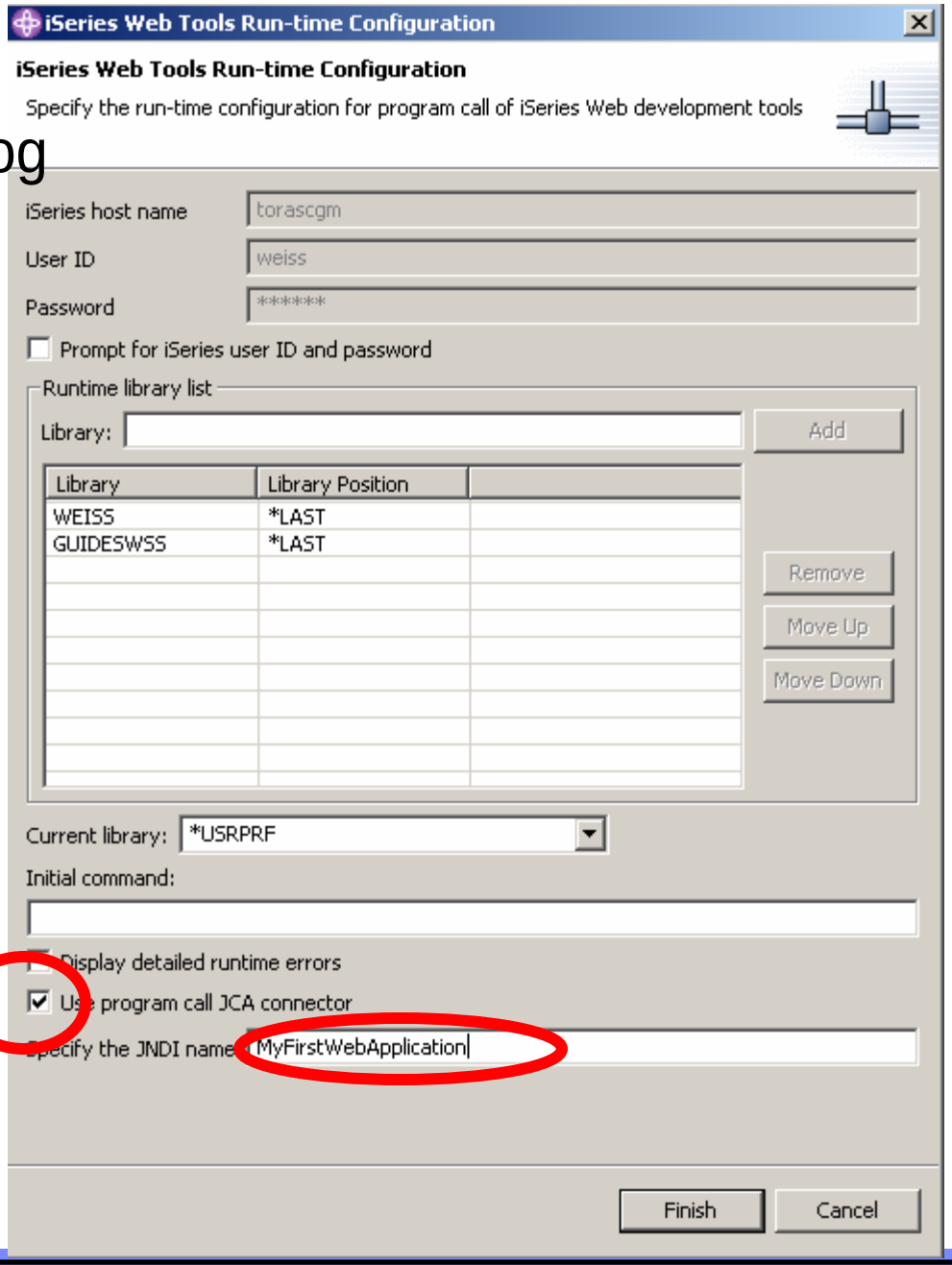
What is JCA Connectors?

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



Creating a connector

- Use runtime configuration dialog
- Check JCA checkbox
- Specify JNDI name



SUMMARY

- System i5 web tools, at a glance
  - Tools optimized for System i5 developers!
    - Web Interaction wizard (deprecated using STRUTS framework)
    - Jsf tools with iSeries support
      - Use Program call bean
      - Use direct access to data base
    - New and enhanced Web Service wizard
      - ✓Make your system i5 programs available as a Web Service
    - JCA wizard
      - ✓Create connectors for you system i5 programs

More Information?

- Information Sources
  - **www.ibm.com/software/awdtools/iSeries**
    - For iSeries Applicaton Development
  - **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 2002 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

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.

Trademarks and Disclaimers

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 of International Business Machines Corporation in the United States, other countries, or both:

AS/400	**e-business on demand**	**i5/OS**
AS/400e	**IBM**	**OS/400**
eServer	**IBM (logo)**	**System i5**
	iSeries	

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, other countries, or both.
Intel, Intel Logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
Linux is a trademark of Linus Torvalds 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.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Other company, product or 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 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 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.