

Web Tooling in WebSphere Development Studio Client

First of a 2 part series...

By Phil Coulthard & George Farr

The hot new release of WebSphere Development Studio Client is the follow-on to WebSphere Development Tools. This new product is available to every iSeries programmer at V4R5 or above. In this article, we will go into detail on the new Eclipse-based Web tooling that is part of WebSphere Development Studio Client. The Web Tooling in WDS is massive and awesome, and well beyond what can be covered in one article. However, we can give you a taste of that tooling...

Web Project and Perspective

Web tooling, like most tooling in Eclipse for a particular task, has its own project type and its own perspective with a collection of views and editors optimized for the role of developing Web applications. The perspective's main view is the standard Eclipse Navigator view, which lists all existing projects and allows exploration, or drill-down, of those projects.

A Web project is used to capture all your Web resource files that constitute a single Web application, such as the HTML, JavaServer Pages, image files, servlets, and so on. To create a Web project, use the File->New->Other menu item and select the Web category on the left, and the Web Project type on the right. This launches the Web project wizard. Other than the project name, you typically can take all the defaults on this wizard. The Web project will be created and pre-populated with the folders that are required by the industry standard for Web applications (the Sun Servlet 2.2 specification).

Once you have a Web project, you can start to use the many views provided in the perspective, such as:

- **Gallery.** Pre-supplied images, sounds, scripts and so on that can be dropped into your Web page.
- **Outline.** When editing an HTML file, shows the tag hierarchy of the page. You can directly work with the tags from this view, if desired.
- **Links.** For the selected file, shows all files that use it or link to it, and all files it uses or links to.
- **Styles.** Specialized editor for cascading style sheets.

There are also specialized editors for creating logos and animated images. The heart of the Web Tooling however, is the **Page Designer**. This is the editor for your HTML and JavaServer Pages, and it supports both source editing (like SEU) and What-You-See-Is-What-You-Get (WYSIWYG, like SDA) editing.

An e-business application typically includes:



1. **Web pages** (typically JavaServer Pages or HTML files, each with JavaScript).
2. **Web page flow controllers** (typically a Java servlet).
3. **Business logic** (either Enterprise Java Beans or RPG/COBOL logic).


A JavaServer Page is very much like a display file, in that it defines the constant part of the page with HTML tags, and leaves holes for the dynamic part of the page by use of special JSP tags that extract data out of Java Beans.



A servlet is a Java class (which extends the servlet class) and is what creates and populates those Java Beans, and then passes control to the JSPs. To populate the bean, a call to some business logic is usually involved to get the output data (and pass the input data). The Servlet does this as well. It is, in fact, the coordinator between the business logic and the JSPs.

To help you get going quickly with servlets and JSPs, there are wizards supplied to generate these, given various different inputs. These are available from the main toolbar while in the Web perspective.

There is a Servlet wizard  that just generates a servlet class stub to get you going. There is also a Database Web Pages wizard  that helps you build an SQL statement, and then generates an input JSP prompting for the input variables in that statement, and an output JSP that shows the results of running that SQL statement. It also will generate the servlet and Java Bean.

There is also a Java Bean Web Pages wizard  that, given a Java bean, will generate an input JSP to prompt for the input properties and an output page to display the output properties. It too will generate a servlet that binds the two pages and that calls one or more of the methods in the Java Bean when Submit is pressed on the first page.

iSeries Extensions to Web Tooling

The base Web tooling is designed to create web pages and Java code to run on all platforms that support Java and an application server such as WebSphere Application Server. To that excellent base have been added additional tools explicitly for an iSeries developer who wants to use new or existing RPG or COBOL business logic to driver their Web applications. That includes a Web Interaction wizard that, given the parameter information for an RPG or COBOL program or procedure, generates an input JSP for the input parameters, an output JSP for the output parameters, and all the glues in-between.

This wizard also allows you to start with your own input or output page, in which case it helps you bind the input and output fields on those pages to the input and output parameters in your business logic.

In addition, Web GUI controls (known as Design Time Controls or DTCs) have been added to the Page Designer that offer DSPF-like properties such as edit code support and validity checking by data type.

One of the more interesting DTCs adds is a subfile control, which is close to the heart of many iSeries programmers. Lets walk through using the Web tooling to create a Web page that prompts

the user for input, then on submit calls an RPG program with that data, and responds with a page showing output data returned by the RPG program. This input and output page duo is known as a single “Web interaction”. Our example interaction will have an input screen that will ask for a customer number, and an output screen that will display customer information.

Hopefully, it’s a small enough example to get you started, yet big enough to get you interested. We will use the page designer to design the input page by hand, then use the Web Interaction wizard to generate the output page, and all the Web application glue to enable calling an existing RPG program.

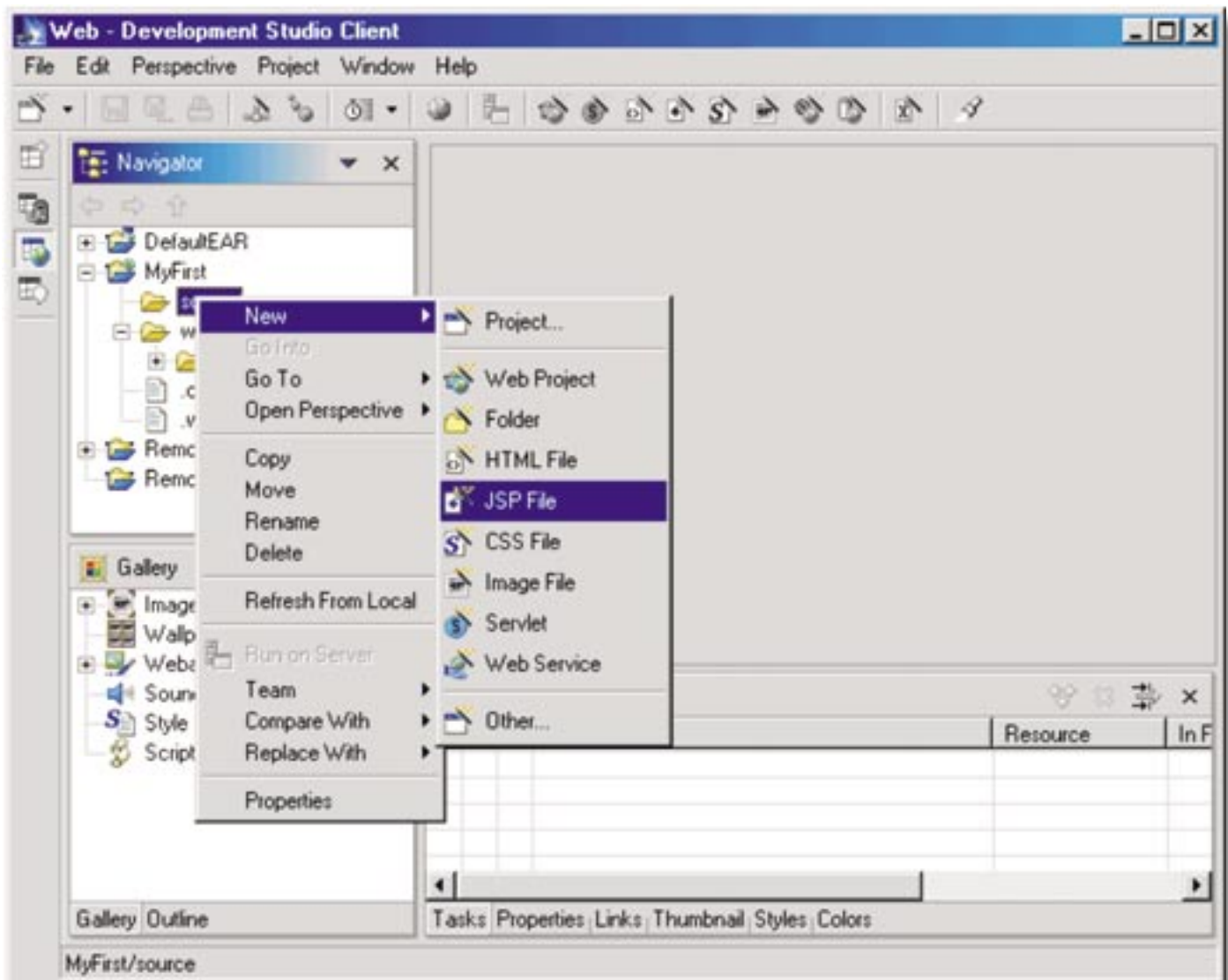


Figure 1. Your first JSP

My first JSP screen

Create a Web project if you have not already, named say MyFirst. Right click on the project and select New->JSP file from the popup menu, as shown in **Figure 1**.

This will bring the page designer into the empty pane and allow you to start designing your first JSP. Remember that this is similar to SDA! You are now designing your input screen as you did with SDA designing your 5250 screens. We will not go into details here on how to create this screen and make it look good, however, here are some hints:

1. To get to page property and change the look and background of the page,

just select Page from the menu and then properties

2. Add an HTML Form by selecting Insert from the menu. Once you have a form, you can add iSeries Design Time Controls in it.
3. To add iSeries Design Time Controls such as an entry field or a push button, select Insert->Design Time Control from the main menu, and pick the iSeries control you want on your page. Once on the page, right click and select Properties to set attributes such as edit code and data type.

Figure 2 shows a simple screen with a couple of iSeries controls (Entry field and push button).

In addition, we changed the look of the screen by adding a sky blue background to it.

Once you are done designing the input screen, you are now ready to create the output screen to display customer information. In this case, we will not create the page manually so that we can show you another feature in Web Tooling, which allows you to have the tool itself generate the page for you. In other words, we are done designing screens in our simple example and now we are ready to create an iSeries interaction. An interaction is nothing but an input JSP page and an output one. Both connected with an iSeries program or service program. To that end, we created

a wizard for you in the Web Tool to ask those questions and create all the necessary code for you to do the actual call. To start the Web interaction wizard go to the icon for that or just go to the menu and select New->Other->Web->Web Interaction. You will then be presented with a screen as shown in **Figure 3**.

All you need to do on this screen is specify a name for the interaction. No big deal, just pick a name as you do for variables in your program. Once that is done press the “next>” button. This will take you to the next screen that will ask you for the input page and the output page. In our case, this example has an input page, which we just finished designing; however, we do not have an output page. Therefore, for the input page you can press the add button and pick the page you just created.

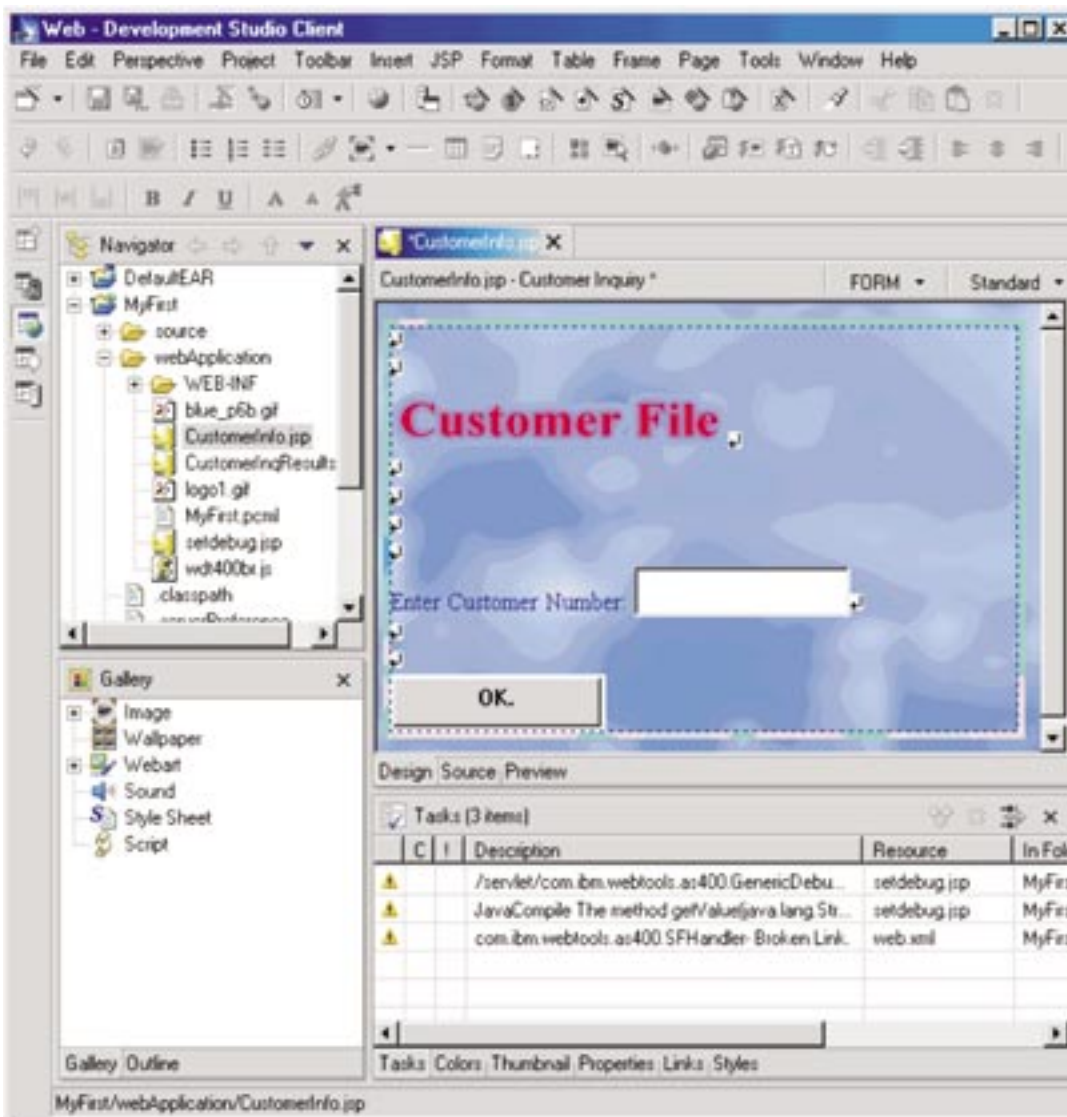
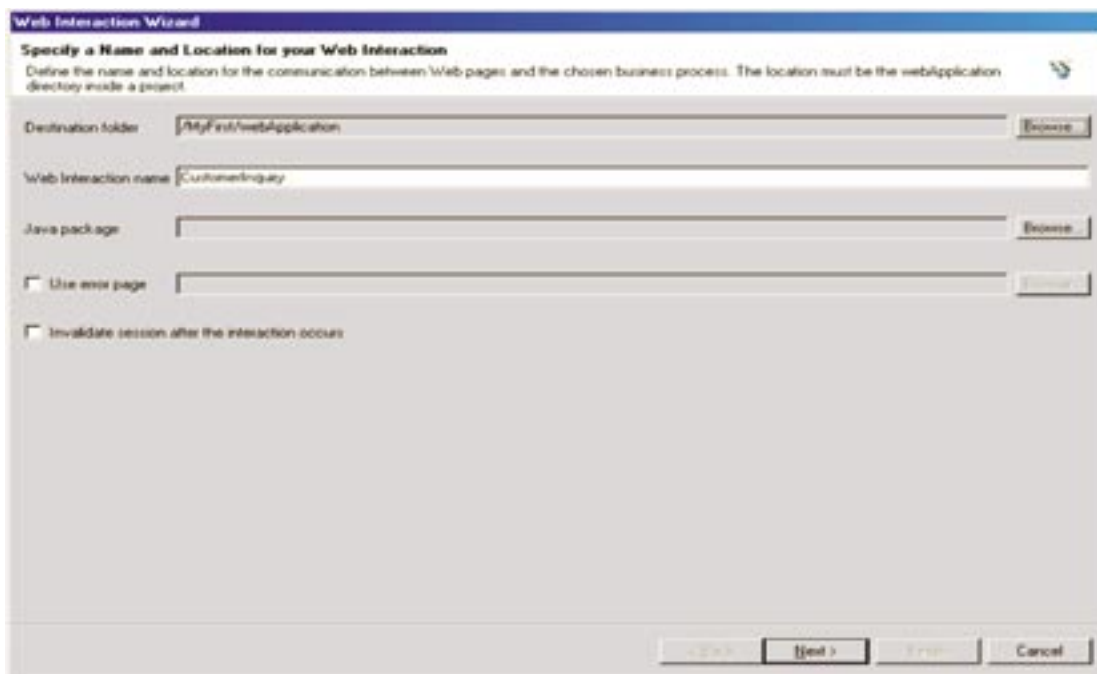


Figure 2. Creating your first JSP with iSeries controls.



George Farr



Phil Coulthard

Figure 3. Web Interaction Wizard

As for the output page, we want to have the designer generate it for us and therefore you need to press the button that says 'generate output JSP'. This will tell the Web tool to generate the output page for you.

In summary, at this point you told the interaction wizard to use an existing JSP page, which you just created and generate the output page with all the fields you will be specifying soon. In the next issue of this magazine, we will explain how to link the iSeries program (*PGM) or service program (*SRVPGM) to the input and output JSP pages with all its input, output, or both parameters...

IBM, WebSphere, iSeries, and VisualAge are trademarks of International Business Machines Corporation in the United States, other countries, or both. Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States, other countries, or both. Windows, Windows NT, and Windows ME are trademarks of Microsoft Corporation in the United States, other countries, or both. Other company, product, and service names may be trademarks or service marks of others.

George Farr and Phil Coulthard are coauthors of the Midrange Computing books *Java for RPG Programmers* and *Java for S/390 and AS/400 COBOL Programmers*. George can be reached at farr@ca.ibm.com, and Phil can be reached at coulthar@ca.ibm.com.

DOMINATE

ProData's Tools are the Dominant Leader in iSeries Utilities.

DBU, the "original" iSeries AS/400 database utility, allows users to view and update any file instantly without time consuming queries, DFU or programming. Now available in multiple power packed interfaces!

- Graphical User Interface
- Green-Screen
- Operations Navigator

Ask about DBU pricing AT PAR (for a limited time only)

CvtRPGIV Move to ILE! Program templates and RPGIV book included. All for \$995!

ProTools Zip/unzip files, email direct from your iSeries, and more...with the source code included. Priced at only \$495

SQL/Pro to query and format reports.

RDR to retrieve deleted records.

ProData

FREE TRIALS 800.228.6318 WWW.DODBU.COM