

Modernizing Your Applications with WDS/400

OCEAN Technical Conference Catch the Wave



Susan M. Gantner
susan.gantner @ partner400.com
www.partner400.com

*Paris Gantner*400
Your partner in AS/400 and iSeries Education

In this session, we will look at several options for modernizing your applications and your development environment. Many of the tools and techniques we will explore will help you web-enable your applications or develop new browser based or graphical user interfaces. We will look in detail at the tools that are part of the new WebSphere Development Tools package, which is included on all V5R1 iSeries development systems. This toolset includes:

- WebFacing to translate your DDS green screens into browser-based interfaces
- VisualAge for Java to develop Java applications or pieces to fit in with your RPG or COBOL applications
- WebSphere Studio to build interactive web pages powered by RPG or COBOL programs running on your iSeries
- VisualAge for RPG to develop modern graphical interfaces to RPG logic, revitalizing not only existing RPG code, but also RPG skills
- And last, but certainly not least, CODE/400 for supercharging your programming productivity both in developing your new applications and in maintaining the older ones.

The author, Susan Gantner, is co-founder of Partner400, a firm specializing in customized education and mentoring services for AS/400 and iSeries developers. After a 15 year career with IBM, including several years at the Rochester and Toronto laboratories, Susan is now devoted to educating developers on techniques and technologies to extend and modernize their applications and development environments. This is done via on-site custom classes as well as conferences and user group events.

Together with her partner, Jon Paris, Susan authors regular technical articles for the IBM publication, eServer Magazine, iSeries edition, and the companion electronic newsletter, iSeries Extra. You may view articles in current and past issues and/or subscribe to the free newsletter or the magazine at: <http://eservercomputing.com/iseries/>.

Feel free to contact the author at: susan.gantner @ partner400.com and visit the Partner400 web site at www.partner400.com.

Notes

WebSphere Development Studio

Partner400

Available for V4R5 and later systems

- A single package that encompasses all of the "Traditional" green screen tools
 - ADTS (PDM, SEU, SDA, RLU, DFU and AFP)
- All of the host based compilers
 - RPG, COBOL, C and C++
- Unlimited licenses of the WebSphere Development Tools (V4R5-V5R1)
 - CODE (aka CODE/400)
 - VisualAge for RPG
 - VisualAge for Java Professional (with the ET/400 Enterprise extensions)
 - WebSphere Studio Pro (with iSeries extensions)
 - WebFacing
 - A brand new tool to give your tired old green screens a new look!
- Websphere Development Studio Client replaces WDT (V5R1-V5R2)
 - Details on later chart

The best news of all?

- You don't have to persuade the boss to spend any money to get it!

WDS is 5769-WDS for V4R5 systems and 5722-WDS for V5R1. For V4R5 users on Software Subscription 5769-WDS can be ordered as a no-charge refresh if they currently license ADTS and at least one compiler. Note that for V4R5 users it is the V4R5 version of the compilers that is shipped.

Individual compilers can no longer be ordered.

Note that the C and C++ compilers are brand new and replace the previous offerings. They run in the PASE (i.e. AIX on iSeries) environment but produce conventional OS/400 programs. Although the compilers are included as part of the WDS package at present (V5R1) it is necessary to purchase the PASE environment from IBM. This a small (approx. \$100) one-time charge levied by the UNIX foundation. In future releases of OS/400 this charge may disappear.

The workstation part of the tool set (WDT/400) can be ordered as a separate product for individuals such as consultants who do not own an AS/400 or iSeries system but who wish to take advantage of the tools. Since the back end components will be shipped on all development systems, they can be assured that they will be able to connect when working on a customer site.

As of V5R2 and later shipped versions (mostly after June 2002)of V5R1, WDS is shipped in place of WDT. WDS differences are summarized on the next chart.

Notes

WDS*c* - WebSphere Dev Studio Client *Partner400*

Replaces WDT/400 - the workstation components of WDS

- On all V5R2 systems and V5R1 systems shipped after June 28

Based on IBM Eclipse "plug-in" development workbench

- Eclipse integrated tools include:
 - New Java development tools
 - New Remote Systems Explorer
 - New Database and XML tools
 - Websphere Studio with iSeries extensions
 - Webfacing Tools
- CODE and VARPG are still packaged with WDSC, but are not fully integrated into the workbench

The WebSphere Development Studio Client for iSeries (WDS*c*) replaces WDT/400 in V5R2 and systems shipped after June, 2002 with V5R1. VAJava is completely replaced with new Java tooling. Other new functions are added such as Remote Systems Explorer and database and XML tools.

CODE and VARPG continue to be shipped and installed as part of the package, but have only peripheral integration with the workbench. The CODE editor, for example, can be evoked from the Remote Systems Explorer, which provides a list oriented interface of iSeries objects.

VisualAge for Java is not part of WDS*c*, but much of its function and many new features are provided by the "native" Java development support built into WDS*c*.

The Webfacing and WebSphere Studio features are very similar to the ones in the original WDT package.

Notes

The CODE Components

CODE Editor

- Language sensitive editing for:
 - RPG, COBOL, CL, C, C++, DDS, HTML, Java and more
- Program Verifiers for RPG, COBOL, DDS
 - Performs a super fast "mini compile" on the PC
- Language Reference manuals online, F1 enabled
 - It has never been easier to find out how to do things
- Invoke Host Compiles
 - Compiles your program and returns error feedback

CODE Designer

- WYSIWYG design of DSPFs, PRTFs, PFs

CODE Project Organizer

- PDM on Steroids!

CODE Distributed Debugger

- Remotely debug RPG, COBOL, CL, C, C++, Java

You can manage your development-cycle tasks from CODE Project Organizer. This tool lets you create and manage development projects on your iSeries system from your Windows(R)-based workstation. With this tool you can view iSeries libraries, files, and members; launch the host compilers, or the workstation editor or debugger.

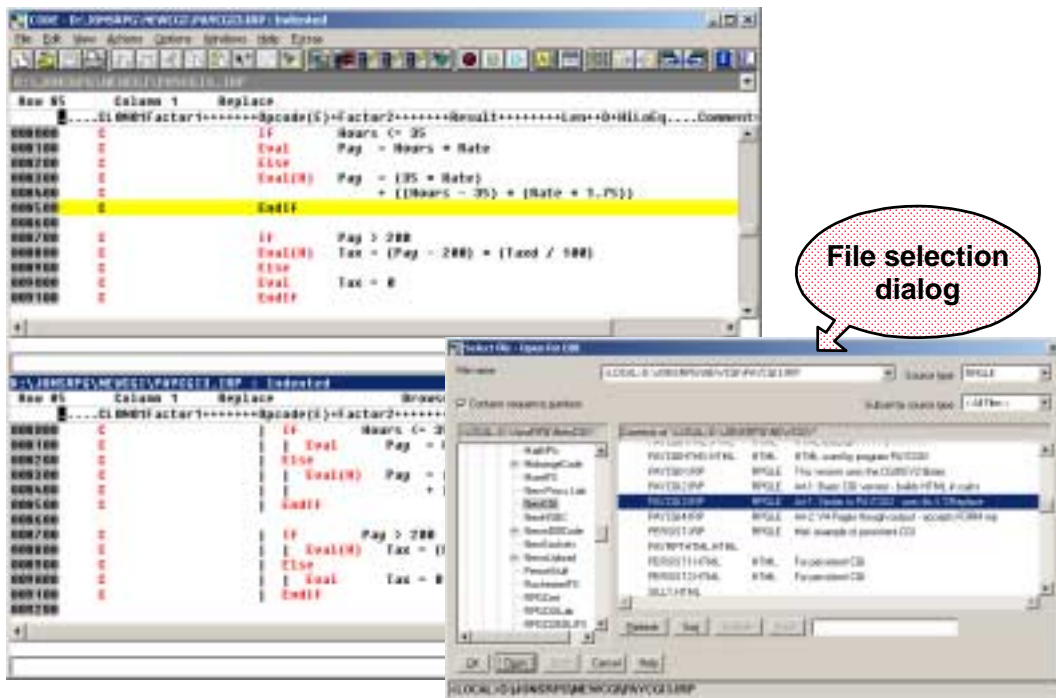
Your program editing tasks are simplified by the syntax checking and token highlighting features of CODE Editor. The editor can access source files on either your workstation or your iSeries system directly. When a compilation results in errors, you can jump from the compiler messages view to an editor view of the offending source statements to correct them. You can use CODE Editor to do offline coding of your applications, then run them through the Program Verifier to check for errors.

You can create and update DDS definitions for display files, printer files, and physical files using the CODE Designer utility. This tool provides a graphical user interface with which you can quickly define your DDS files, and save them locally or upload them directly to your iSeries system. By creating your DDS screens visually you can improve their usability (because more of your time is spent on visual design, and less on getting the syntax right) and your programming productivity. You can also develop DDS definitions using CODE Designer while offline

With the Distributed Debugger you can debug your program running on the iSeries system from a graphical user interface on your workstation. Object Level Trace takes debugging a step further, so that you can step through a distributed application, and debug both client and server code from a single workstation.

Notes

Code Editor - Split Screen Indented View *Partner400*



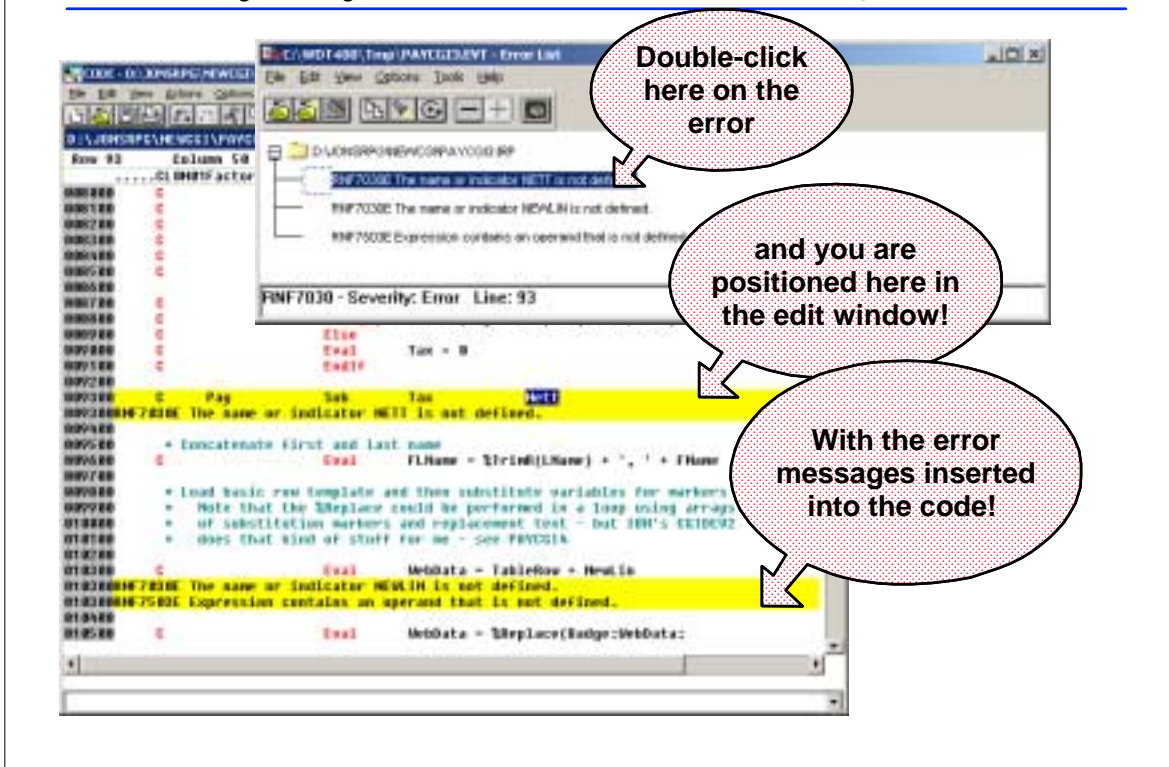
CODE can work in a split screen mode as shown here. There are significant differences between this and the SEU split screen capability, however. Because we're on a Windows workstation, we can typically see much more code in this split screen mode. In fact, the split can be made vertically as well as horizontally so that you can see many more lines of code.

The split screen can contain the same or another source member. And it can contain an indented view of the code, as shown here. The indented view we show here is not updatable, because indented views are read-only. However, if I were editing either the same or a different source member in "normal" (not indented) mode, we could update both views in split mode. We can even split the screen into more than 2 parts if we want to.

Notes

The Easy Way to Locate Errors!

Partner400



One of the most useful tools in the CODE toolset is the CODE verifier, which is illustrated here. The verifier will locate any "compile-type" errors in your code. Because the verifier runs locally on your workstation, there is no need to submit a batch compile job and wait for the results. If errors exist, they will pop up in a window as illustrated on this chart. Double clicking on an error in this window will position directly to the line of code in error in the editor window.

The verifier can also be used when you are not connected to your iSeries host. You need to have previously requested that the external file definitions and any copy members, etc. be "cached" locally on your workstation disk. Then you can do verifications locally to find any errors in your code without even a connection to your host. This feature is perfect for telecommuters.

Notes

VisualAge for RPG (VARPG)

Partner400

Create PC GUI applications using RPG as the base language

- Compile and run as a Windows executable
- Or as Java (run anywhere!)
- Can create non-visual projects too
 - So your reusable subprocedures can be used in a PC environment!

GUI Builder

- Visually layout Graphical User Interface
- Program the event logic ... in RPG
- Uses the CODE/400 editor

Compiler and Runtime

- Uses the latest RPG language level

Debugger

- GUI debugger to simplify the task of finding those nasty bugs

If you are already an experienced RPG programmer, you can create GUI interfaces to RPG programs very quickly in VisualAge RPG. You build the GUI Interface by selecting visual parts from a parts palette and dropping them on a design window. You can alter the size and text of push buttons, check boxes, and other GUI components.

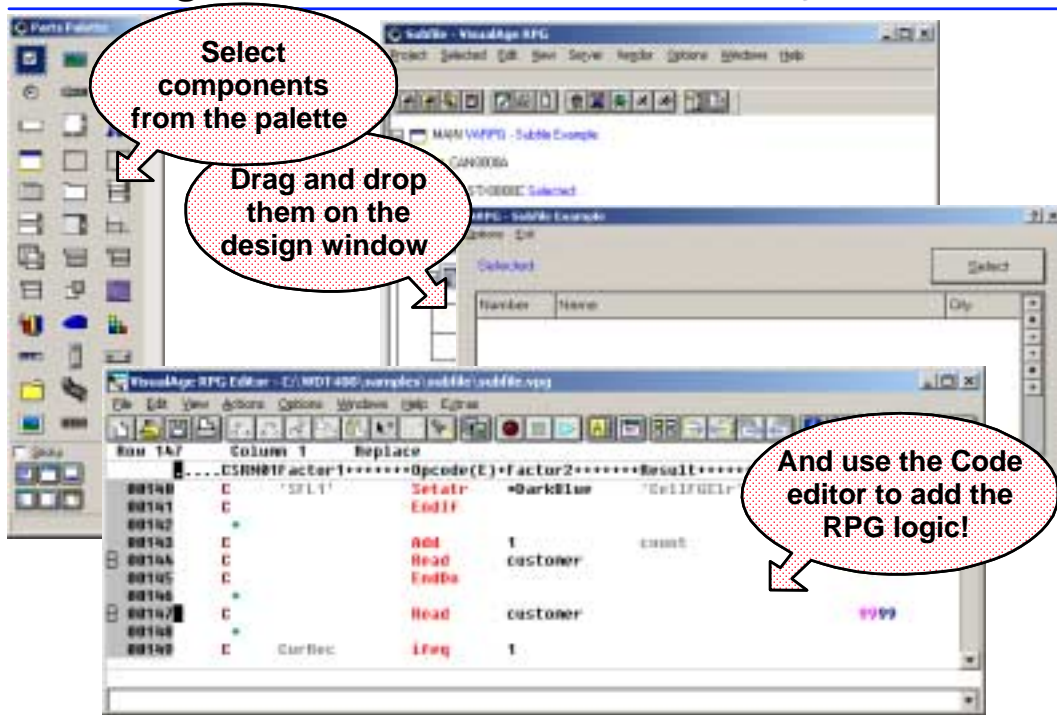
You then associate user events, such as the pressing of a button, with VARPG code you develop. The VARPG code can include statements to access iSeries databases. Accessing the database couldn't be simpler, you just write READ, WRITE and CHAIN statements as normal and the compiler takes care of communicating with OS400.

The compiled program runs as an executable on a Windows system, or as an applet in a Java Virtual Machine in a Web browser, and accesses remote data on the iSeries system using TCP/IP.

The only skills you need to develop these applications are the RPG skills you already have, and the ability to learn the straightforward new user interface of the VisualAge RPG GUI Designer.

Notes

VisualAge RPG: RPG With Attitude! *Partner400*



VisualAge RPG's drag and drop interface makes developing the graphic user interface for your applications easy. Simply choose a part from the Parts Palette (such as an entry field, a push button, a subfile, an image) and drop it onto a window in your work surface. Logic is added to the application by coding Action Subroutines (BEGACT / ENDACT). Action subroutines will be called by the VARPG run time support when the user takes some action on the parts on the window (such as pressing a button).

The RPG logic is "ordinary" RPG IV logic, with some extensions to deal with interacting with the attributes of parts (such as the color of text or backgrounds) and with the event-driven nature of the applications that are built using VARPG. However, if data is needed from the iSeries host for the VARPG application, a simple READ or CHAIN operation will supply it, just as it does on the host.

Notes

VisualAge for Java Professional

Partner400

A full Integrated Development Environment (IDE) for Java

Repository / Workbench

- Supports version control

Incremental, automatic compilation

Built-in functionality:

- Editor, debugger, JDK, WebSphere Application Server, and more ...

Visual Composition Editor

- Visually layout and "wire" GUI and non-GUI

Support for creating Java

- Classes, Beans, Applications, and Servlets

Support for 3rd party source control tools

- SourceSafe, ClearCase, PVCS, Aldon Affinity, ...

Note: VAJava is no longer part of WDS as of the new WDSC Client. A later chart will illustrate Java features available in WDSC,

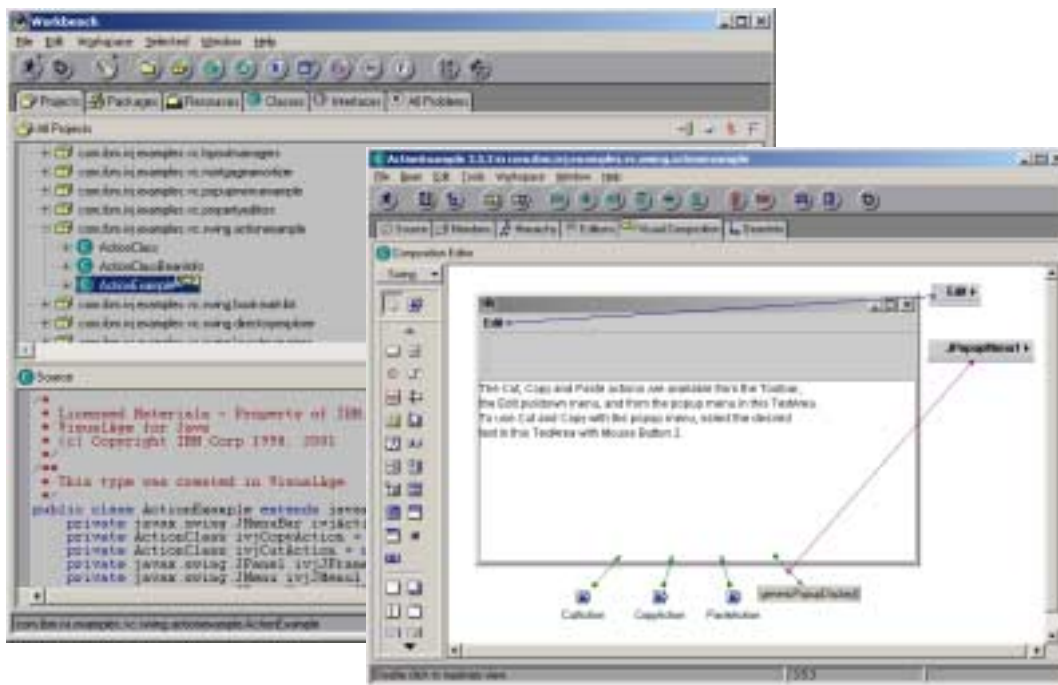
You can use the Professional version of VisualAge for Java provided to generate any type of Java applications including ones that run on your iSeries system and access the database. This award-winning product has been customized for iSeries programming to make it easy for you to call compiled iSeries applications from your Java code.

The VisualAge for Java Integrated Development Environment (IDE), is a set of windows that provide access to development tools. You can open a window for a package, project, class, or member. Standard windows include the Workbench, Repository Explorer, Log, Console, and Integrated Debugger. Other components include:

- The Visual Composition Editor, which can be used to create graphical user interfaces from prefabricated beans, and to define relationships (called connections) between beans.
- The Integrated Debugger, with which you can debug applets and applications running in the IDE. In the debugger, you can view running threads, suspend them, and inspect their visible variable values.
- The WebSphere Test Environment (installed with the JSP/Servlet Development Environment), which provides server runtime support for locally testing and debugging JSP files and servlets. JSP files and servlets that run successfully in the WebSphere Test Environment should also run successfully in a WebSphere Application Server production environment.
- The JSP/Servlet Development Environment, which enables you to develop, run, and test JSP files and servlets.
- The Create Servlet wizard, which enables you to create servlets. When you are developing servlets with the SmartGuide you can use JSP files that inherit from the PageListServlet class.

For larger scale or enterprise-wide applications, you can use VisualAge for Java Enterprise Edition (purchased separately) to develop applications using Enterprise Java Beans (EJBs).

VAJava IDE & Visual Composition Editor *Partner400*



VisualAge for Java and the visual Composition Editor are ONLY available in the older version of the toolset - WDT/400. There is currently no visual composition editor for Java in WDSC. The new Java tooling in WDSC is covered in a later chart.

Notes

Enterprise Toolkit For iSeries - ET400 *Partner400*

Includes AS/400 Toolbox for Java

Remote Export/Compile/Run/Debug

Import Tool for Display files

- Converts display formats to Swing

Program-Call SmartGuide

- Generates Java Bean for calling *PGM object from GUI

OS/400 Specific Beans

- Entry field, label, comboBox, etc. with OS/400 attributes
 - For example edit-codes and error checking
- DFU-like Beans for accessing DB2/400
- PDM-like Beans for retrieving lists of AS/400 object information
 - Including support for PDM's subsetting capabilities, IFS info, etc.

The toolkit provides you with two wizards. The first can be used to convert an existing display file object into Java GUI code. The other creates a program call class that calls any iSeries program.

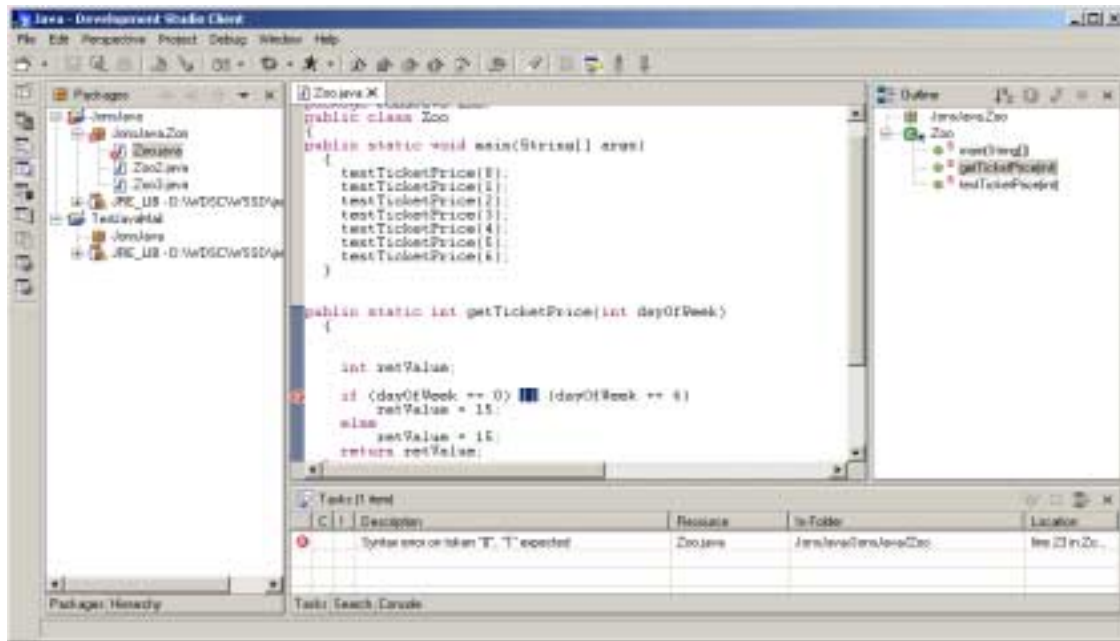
The toolkit also provides a set of beans that can be opened in the Visual Composition Editor and used for accessing and formatting your iSeries data:

- JFormatted beans, to extend Swing (or JFC) classes and provide the ability to convert iSeries fields and attributes, and provide edit code, edit word, formatting, and verification capabilities.
- Data File Utility (DFU) beans, to extend the support of code to access one or more iSeries database files. These beans also allow manipulation of records within the database files.
- Object List beans, which provide access to iSeries object names (for example, libraries, files within a library, or user IDs on the system) and allow you to set listing properties for selecting the desired type of object list.

The AS/400 Toolbox for Java also provides a set of Java classes that allow you to easily access iSeries data from within Java programs.

Java programmers can write custom user interfaces that access remote OS400 applications, using the AS/400 Toolbox for Java component. You can use either the Visual Composition Editor in VisualAge for Java or hand code your own custom calls to Java GUI classes (e.g. Swing classes) to develop your GUI interface, then use the program call support within the Toolbox to call applications running on the iSeries system.

Notes



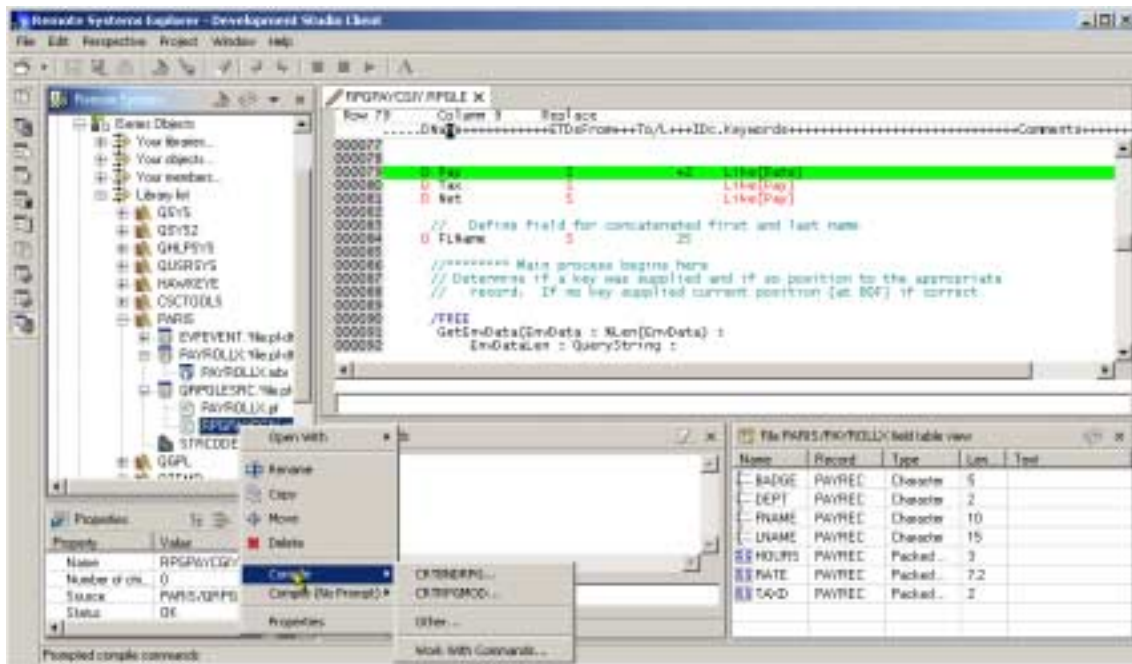
The WDSc Java Perspective includes a built-in editor and debugger, an incremental, automatic compiler and support for Java technology including classes, Java Beans components and servlets. The iSeries extensions include access to Java Toolbox classes in addition to the ability to easily compile, deploy, run and debug remotely on the iSeries.

Shown here is an example of the concept of perspectives in the Eclipse workbench. This is the Java perspective, which has windows corresponding to the Java packages, the Class hierarchy, source editor and error feedback.

Notes

WDS Remote System Explorer

Partner400



Here we see the WDS Remote Systems Explorer, which provides a list-oriented interface to OS/400 objects similar to the Windows file Explorer interface. Notice that we can bring up our RPG source code into the editor in the workbench. However, CODE still provides far superior edit capabilities on "traditional language" source members. Over time, the editor capability within the native workbench editor should improve and may eventually replace CODE. However, this is not anticipated to happen in the short term.

Notes

WebSphere Studio

Create and manage Web Projects

- Group all files related to the Website
- Check-in and Check-out files
- Publish project to web server
- Register any tool for any file type

Page Designer

- WYSIWYG editing of HTML, JSPs

Wizards for creating:

- SQL statements
- JSPs and servlets
 - from SQL statements and Java Beans

Integration with VAJava

- Read and Write classes, beans, servlets from/to VAJava
- Publish website to VAJava for testing purposes

WebSphere Studio provides an easy deployment mechanism for your Web-enabled applications. These include applications you have hand-coded on your workstation that use HTML, JSPs, Java servlets, and applications whose Web front-end you create using one of the code-generating wizards within WebSphere Studio.

You can easily customize HTML and JSP pages using the Page Designer editor in WebSphere Studio. Page Designer lets you create and update input forms, change the appearance or placement of text, and add backgrounds and images to your pages. You can also add controls to handle input errors or to provide fancy touches like scrolling banners; Page Designer generates the appropriate JavaScript code for you.

You can use the Publishing Setup wizard in WebSphere Studio to set the parameters to use for publishing your project to an iSeries server. When you choose to publish your work, WebSphere Studio takes care of creating the necessary files and folders on the iSeries system, and also of creating or checking for the WebSphere Application Server services that your application needs.

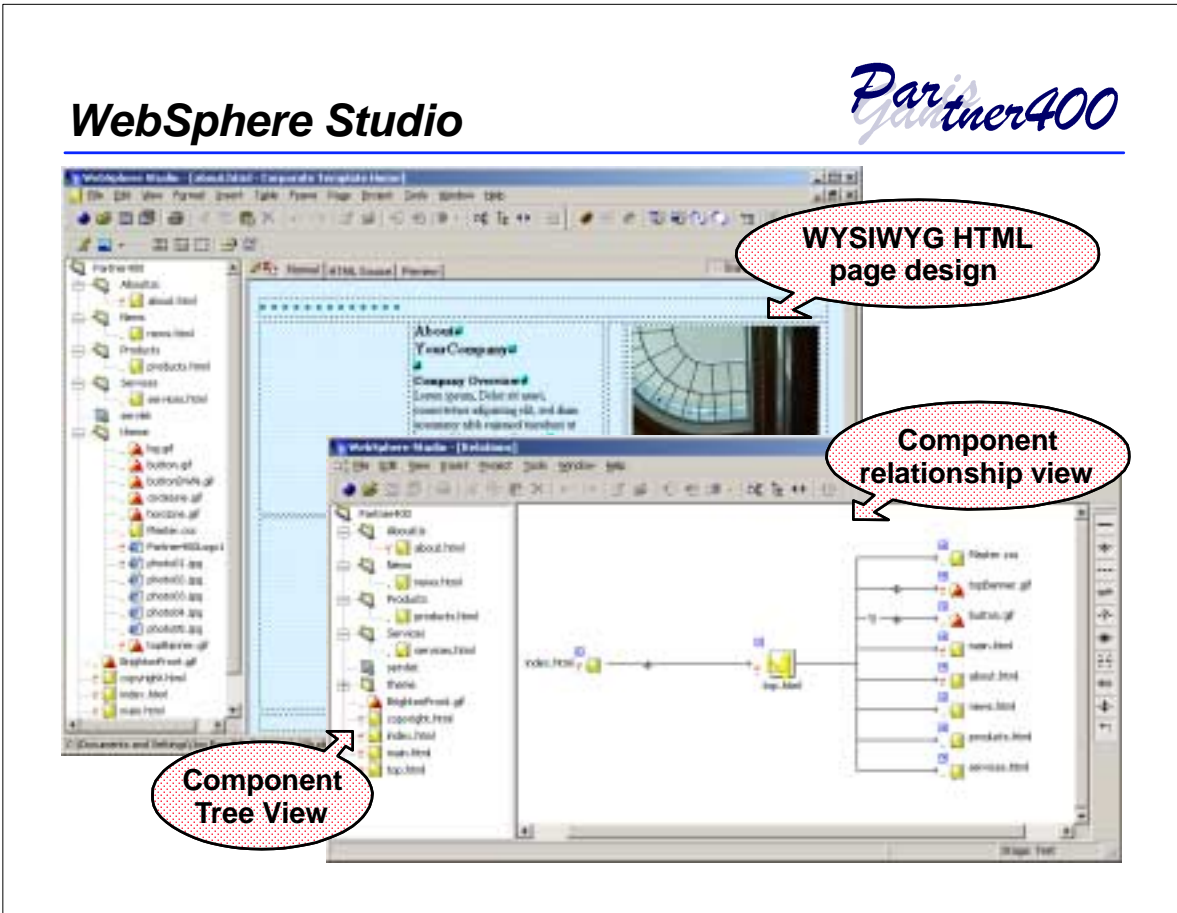
Once you have tried out your deployed application you can easily make changes, then republish the application; WebSphere Studio republishes only changed files by default, which speeds up your development and deployment process. WebSphere Studio also takes care of stopping and restarting the required WebSphere Application Server services after publishing.

WebSphere Studio functions are available in both WDT/400 and WDS.

Notes

WebSphere Studio

Partner400



WebSphere Studio provides the ability to develop and maintain web pages as well as generate web applications that utilize RPG and/or COBOL programs on the iSeries.

Notes

WebSphere Studio - iSeries Support *Partner400*

Display file style Palette Parts in Page Designer

- Smart entry fields, labels, buttons, subfile, etc

Program-call Wizard

- For using new and existing iSeries *PGM/*SRVPGMs
 - Must be non-interactive programs
 - Use WebFacing for interactive programs
- Describes *PGM/*SRVPGM input and output parameters

Generate required output components

- HTML Form prompting for input
- JSP to display output
- Java Bean to call *PGM or *SRVPGM
- Servlet to glue it all together

WebSphere Application Server (WAS) Configuration Wizard

- Configures your application on iSeries WAS

AS/400 Design-Time Controls are a feature specific to the iSeries version of WebSphere Studio; they allow you to create Web versions of your input and output pages with the same kinds of input validation and output formatting and subfile controls that native DDS screens provide. You do not need a detailed knowledge of HTML or JavaScript to accomplish any of these tasks.

Notes

WebFacing - The New Kid in Town

Partner400

Allows existing applications to utilize the browser

- Program does not even need to be recompiled
 - May need small changes to accomodate unsupported DDS features
 - Or to support additional features only available to browser users
 - For example to supply the details of an image to be displayed
- Program can now work with 5250 or Browser
- WebFacing is not strictly a screen scraper
 - Display file buffer is intercepted in Workstation Manager
 - No 5250 control characters to strip off
 - Much faster in operation than conventional methods

You can use the WebFacing tool to create Web front ends to green-screen applications. You choose a Web style from one of several predefined styles, or you can design your own, then work with the tool to generate a set of JSP pages and JavaBeans that interact with the logic of your program. The WebFacing tool is ideal for applications that you want to deploy broadly over a corporate intranet or the Internet, where rapid deployment takes precedence over customizing each page's look and feel. Because WebFacing does not require you to make changes in the application code, you can continue to deliver your application via 5250s, while using the same programs to deliver the application through the Web. Note that there are some restrictions on which DDS keywords can be converted, as described in the WebFacing documentation.

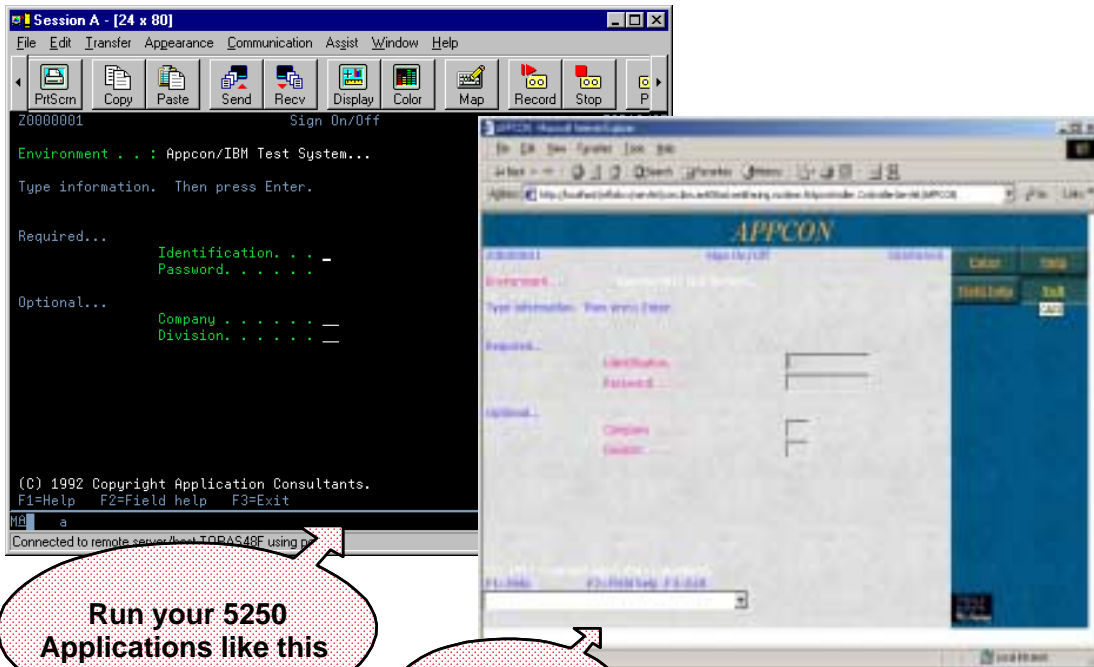
The WebFacing tool is a good choice for programs where the DDS user interface is completely interwoven with the business logic. If your program is well architected, with a set of distinct entry points for each element of business logic, and a user-interface component that simply generates appropriate calls to business logic entry points based on user inputs, you should consider using WebSphere Studio to develop your own servlets and JSPs to access your code.

The WebFacing tool is available in both the WDT/400 and WDS*c* packages.

Notes

Your Choice: Green Screen or GUI

Partner400



As we mentioned earlier, one of the main benefits of WebFacing is that it leaves the original program unchanged. Because of this, same program with no changes can drive a browser or a conventional 5250 simultaneously.

When a user invokes a WebFaced application from the browser, the WebFacing server on the iSeries system starts the host program. The server interacts with all calls to READ, WRITE, and EXFMT operations and use the JSPs and Java Beans to interact with the browser.

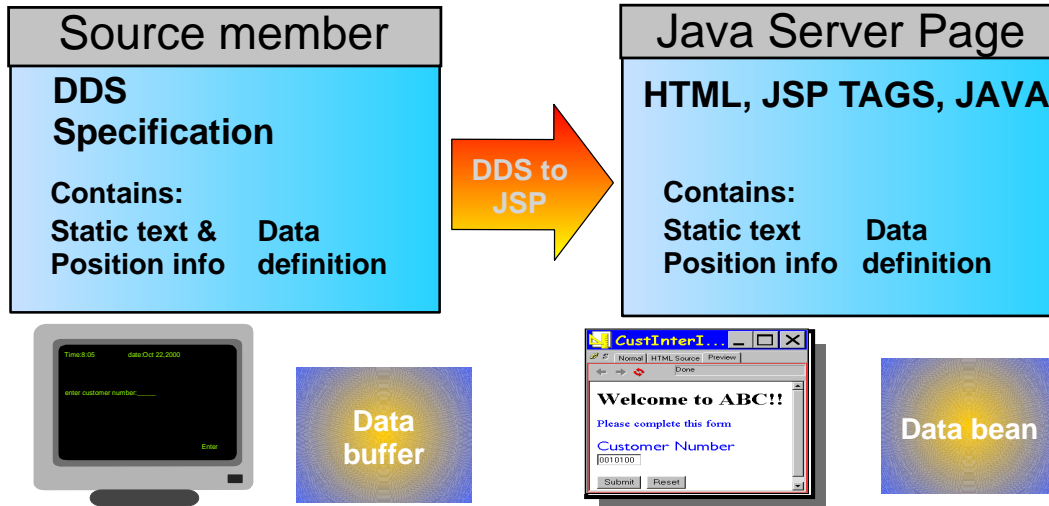
You may need to make coding changes if your application uses DDS keywords that are not supported by WebFacing, or if you want to modify the DDS screens so that the conversion to Web format produces a more attractive or consistent result.

WebFacing also allows you to insert additional information in the in the display file DDS that is ignored by the DDS compiler. This can be used to supply information to the WebFacing tool as to how the file should be converted in the event that you need to reconvert the file in the future.

IBM supplies an API that can be queried by your program if you wish to vary the output of the program based on whether it was invoked from a 5250 or a browser. The API is QpfEnvironment in Service Program QPFENV. This API returns 1 (one) when invoked by a WebFaced UI and 0 (zero) when invoked by regular 5250.

Notes

WebFacing - DDS to JSP Conversion *Partner400*



Select one or more DDS members to convert

Creates JavaBeans and JavaServer Pages for each record format

Bean holds data for record format

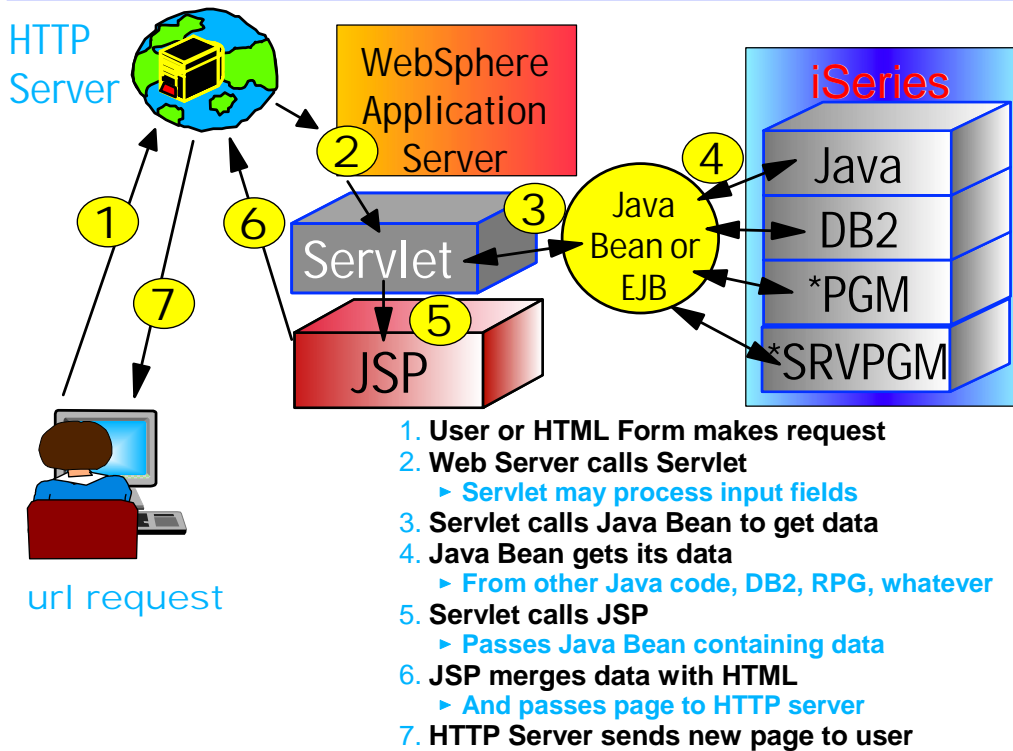
JSP displays and/or prompts for the data

Webfacing runs with Websphere Application Server. Someone in the shop needs to understand how to set up and manage a WAS server, which is far more complicated than the webfacing itself.

Notes

WebFacing - The runtime picture

Partner400



The browser understands HTML and JAVA script, in this environment as in most application server environments the HTML is produced by JSPs. The is JAVA script embedded in the HTML to recreate the functionality of the 5250 in a browser environment. Both HTML and Java script are stored on the web server and are only transferred to the browser on demand.

The application server environment uses Servlets to route requests from the browser. Servlets are JAVA programs running in the Application Server (AS). The Servlets that make up the WebFacing environment are published to the AS and are installed on the development system when the WebFacing component is installed.

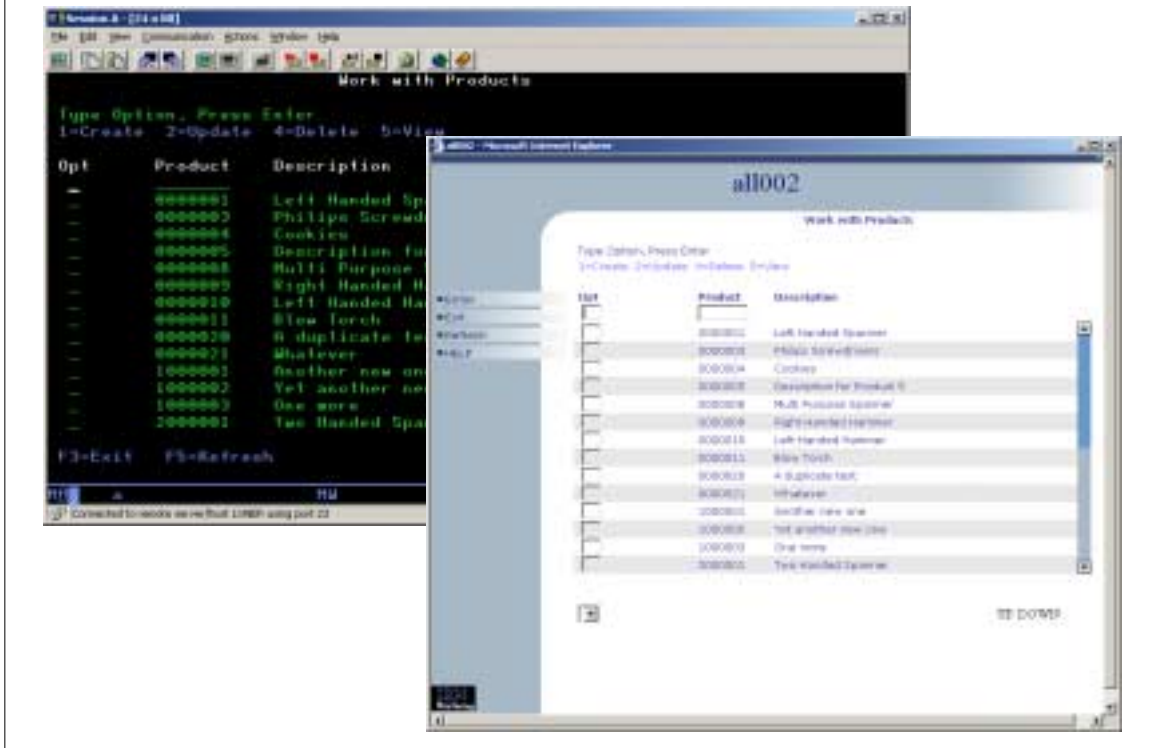
The data beans take the data that is entered in the browser and they are used to pass this data to the web facing server. When responding to a request the data beans get filled with the data from the interactive program and the data then gets inserted into the JSPs from the data beans. The layout of the data beans is determined at conversion time when the beans get generated. The Java Server Pages contain the looks of the UI. They are also created during DDS conversion

The WebFacing server is located on the server that contains the interactive program that drives the UI. At startup time of the application, it will receive the command used to start up the interactive application, and create a job if this is a new session. It will run the invocation command and the program will then send out a record format to be displayed. The request to display the record format and any data sent by the program will be routed thru workstation manager to the WebFacing server

Notes

A "WebFaced" Application

Partner400



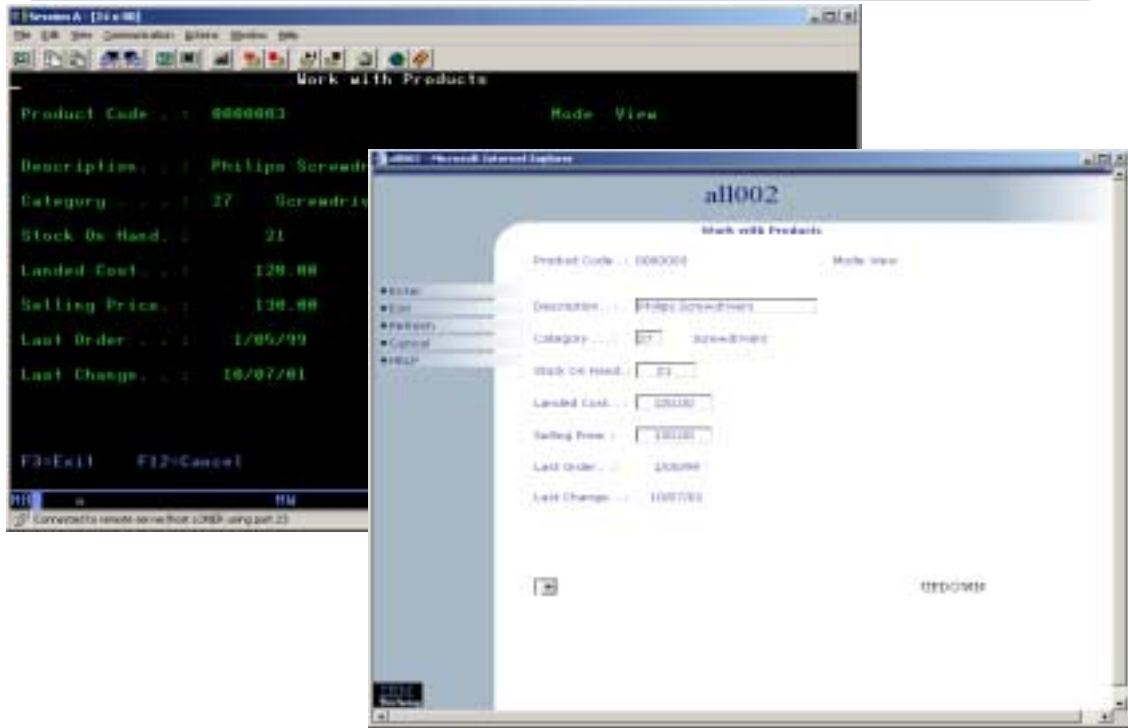
Here's what our ALL002 program looks like with default webfacing!

We can adjust this webfaced application to respond to a simple click on a Product code in the list as a selection in addition to allowing entries in the Option field.

Notes

A "WebFaced" Application

Partner400

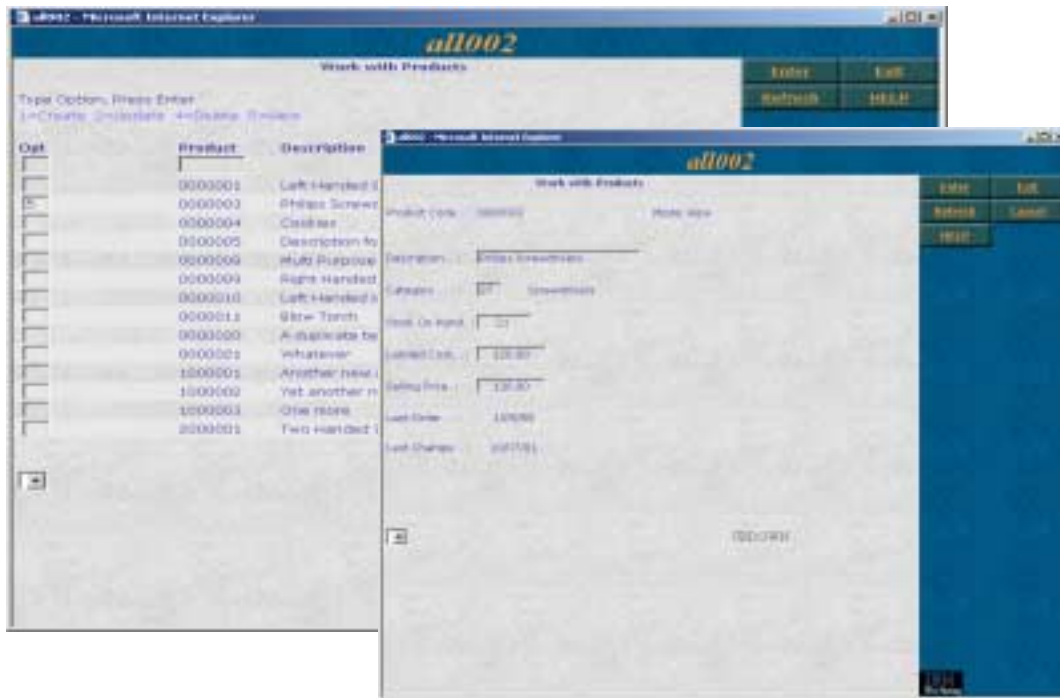


Here's what our ALL002 program looks like with default webfacing!

Notes

Try a Different "Face"

Partner400



You can quickly and easily add a new face to the application by applying a different style. Shown here is one of the alternative styles that comes with the product. You can customize the styles to look the way you want them to look.

Notes

Browser and Server Requirements

Partner400

Browser

- IE 5.0 or later will provide best support
 - Neither Netscape 4.7 or 6.0 have the support
- JavaScript
 - IE level of support is needed

HTTP server

- IBM HTTP server
- APACHE
- others
- Can be located on any system

Application Server (WAS 3.5.2 and WAS 4)

- Plugs into HTTP server
- Currently only WebSphere Application Server (WAS) supported

Web References

Partner400

www.ibm.com/software/ad/wds400

- WebSphere Development Studio for iSeries

www.ibm.com/as400/java

- The iSeries and AS/400 Java homepage

www.ibm.com/software/ad/vajava

- VisualAge for Java homepage

www.ibm.com/software/ad/vadd

- VisualAge Developer Domain

www.ibm.com/alphaworks

- Lots of free Java stuff! Eg, Java Beans, XML tools