iSeries WebFacing
IBM eServer iSeries Application Development

This presentation reviews the components of a Web application, the WebFacing Tool and how it is packaged, how the WebFacing Tool development time and runtime works and the steps to Web-enable a 5250 application using the WebFacing Tool.
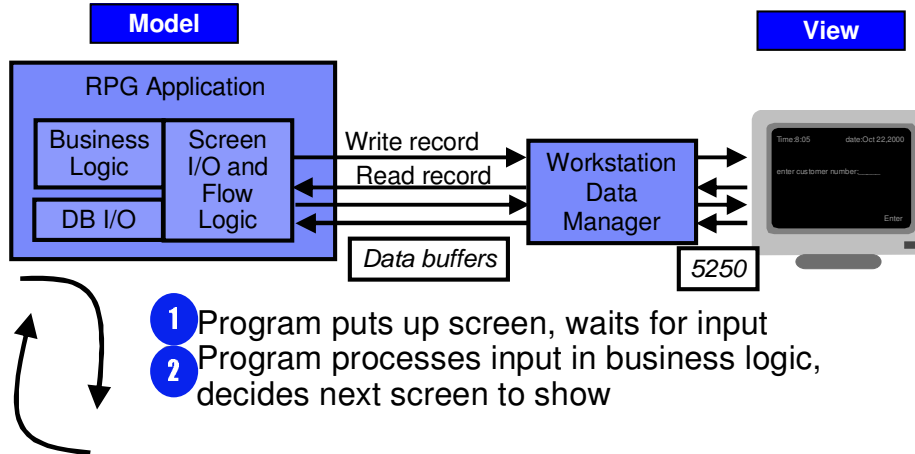
Table of contents

In the first section of this presentation we review what e-business is all about. Then we look at WebFacing and what it is. Next we introduce Development Studio Client. The WebFacing Tools are described in more detail. WebFacing customization is described followed by a review of the new WebFacing Tool Version 5.0 features.

IBM

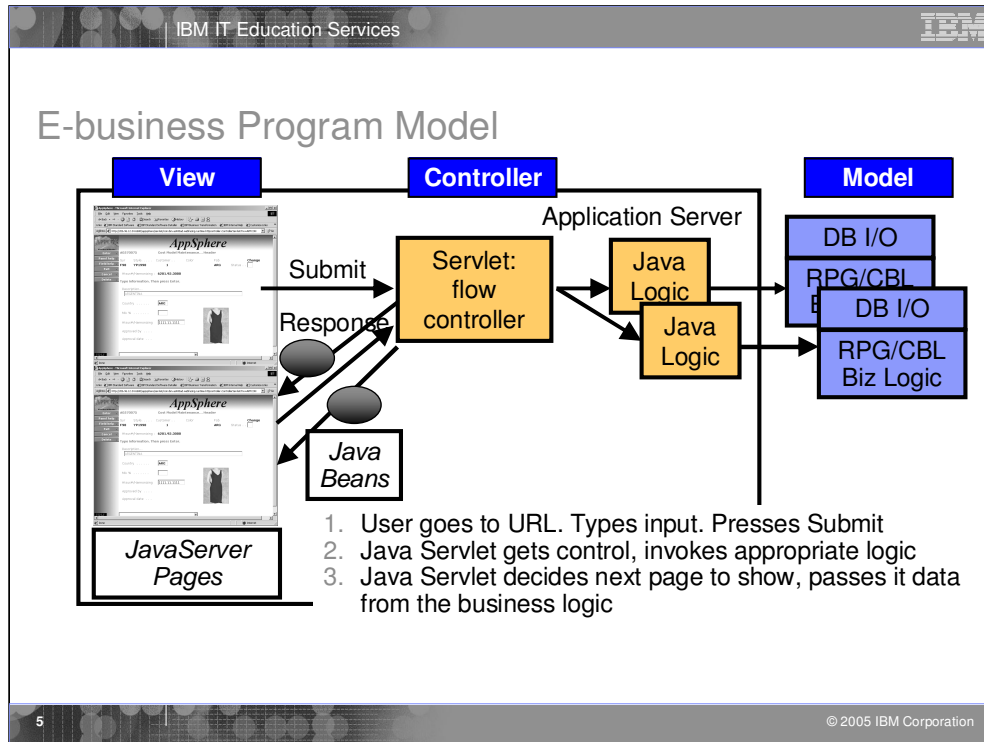# e-business Primer

- Program Models

- Servlets

- JSPs

In this section we review the programming models and their associative technology, servlets and JSPs.

iSeries Classic Program Model

Model

View

RPG Application

Business Logic | Screen I/O and Flow Logic
DB I/O |

Write record
Read record

Workstation Data Manager

Data buffers

Time:8:05   date:Oct 22,2000

enter customer number:_____

Enter

5250

1 Program puts up screen, waits for input
2 Program processes input in business logic, decides next screen to show

In the green-screen environment, the application performs READs and WRITEs to the workstation. The application data is sent to Workstation Data Manager. The Workstation Data Manager merges application data with the display file. The Workstation Data Manager generates a 5250 datastream that is sent to the display.

In this model, you typically have one model object and multiple views on that object. A view is a window onto the model. Presentation logic typically goes in the view. Business logic goes in the model.

4

E-business Program Model

View	Controller	Model

Application Server

View: JavaServer Pages, AppSphere screens

Submit → Servlet: flow controller
Response ←

Java Logic
Java Logic

DB I/O
RPG/CBL Biz Logic
DB I/O
RPG/CBL Biz Logic

Java Beans

JavaServer Pages

1. User goes to URL. Types input. Presses Submit
2. Java Servlet gets control, invokes appropriate logic
3. Java Servlet decides next page to show, passes it data from the business logic

The e-business program model follows the Model-View-Controller paradigm, organizing the application into three separate components:

•Model: the application model with corresponding data representation and business logic

•View: data presentation, providing views for user input

•Controller: to dispatch requests and control data flow

Here you can see the controller is added. The controller handles the interactions between the view and the model. When the model changes it updates the view, when the user does something with the view the controller informs the model.

First the application is converted. This creates JSPs for each record format as well as Java beans.

When the program performs a READ on a record format, control as well as the application data is sent to the controller.

Because the job was started by the WebFacing server, the controller knows this is a WebFacing request and passes the data and control to the WebFacing server which runs on the iSeries. Control returns to the Webfacing runtime servlet that runs in WebSphere Application Server. The WebFacing servlet locates the appropriate JSPs and Java beans. The WebFacing servlet tells WebSphere Application Server to return the JSP back to the browser. The JSP is compiled and the resultant HTML is returned to the browser. Note that no 5250 datastream is generated in this flow.

What Are Servlets?

- **Servlets are . . .**
- Java classes (programs written in Java)
- **Servlets run . . .**
  - On an application server such as WebSphere
- **Servlets are called . . .**
  - By a user pressing Submit, or clicking a link, that is mapped to the servlet
- **The input to Servlets are . . .**
  - User-entered data from a Web page
- **The output of a Servlet is . . .**
  - Control passed to a JavaServer Page

Servlets are server-side Java programs that use the *Sun Microsystems Java Servlet API* and its associated classes and methods, as defined in the *Sun Microsystems Java Servlet 2.3 Specification*. These Java programs extend the functionality of a Web server by generating dynamic content and responding to Web client requests. When a browser sends a request to the server, the server can send the request information to a servlet, so that the servlet construct the response that is sent back to the browser.

Just as applets run on a Web browser and extend the browser's capabilities, servlets run on a Java-enabled Web server, such as the WebSphere Application Server, to extend the server's capabilities. Servlets are commonly used to allow businesses to connect databases to the Web, due to their flexibility, scalability, and the their processing economy when developed in the WebSphere Studio Web development environment.
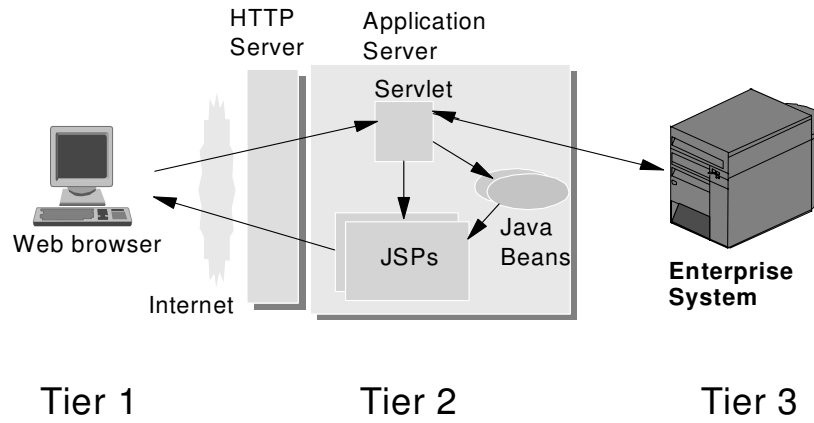
IBM

What Are JSPs?

Just like a *DSPF

- **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 fully resolved Web page, displayed to user
  - Can be just HTML, or DHTML …
    - HTML 4.0+, Cascading Style Sheets, JavaScript

JavaServer Pages enable you to generate dynamic web content, such as HTML, DHTML, XHTML, and XML files, to include in a Web application. JSP files are one way that the WebSphere Studio implements server-side dynamic page content. JSP files allow an Web server, such as WebSphere Application Server or Apache Tomcat, to dynamically add content to your HTML pages before they are sent to a requesting browser.

When you deploy a JSP file to a Web server that provides a servlet engine, it is preprocessed into a servlet that executes on the Web server. This is in contrast with client-side JavaScript (within <SCRIPT> tags), which is executed in a browser. A JSP page is ideal for tasks that are better suited to execution on the server, such as accessing databases or calling Enterprise Java beans.

Web App Architecture

HTTP
Server

Application
Server

Servlet

Web browser

JSPs

Java
Beans

Internet

Enterprise
System

Tier 1

Tier 2

Tier 3

Table of contents

Previously we introduced WebFacing and reviewed the goals and history of WebFacing. Now lets introduce Development Studio Client.

WebSphere Development Studio

RPG	COBOL

Current 5722-WDS customers with software subscription for V5R3 and V5R2, to upgrade use feature #: 2656
Available after GA

C/C++	PDM SEU SDA, RLU

Unlimited Licenses

iSeries Java™	iSeries Debug	iSeries Struts / Web	iSeries Web Service	Web Facing	iSeries Projects / RSE	+CODE +VisualAge RPG

JSF	EGL Java generation	Trace	Profiling	DB	XML	App Server	HATS Toolkit

www.ibm.com/software/awdtools/iseries

WebSphere Development Studio Client V6 based on RWD V6

10 © 2005 IBM Corporation

There is now only one application development product sold by IBM, for iSeries, as of V4R5. This is WebSphere Development Studio (Development Studio), which includes all four host compilers, all traditional tools (ADTS = PDM+SEU+SDA+RLU+DFU+AFP+CGU), and unlimited licenses of the workstation-based toolset named WebSphere Development Studio Client (formerly WebSphere Development Tools).

If you are an existing customer who has a subscription, you can upgrade to Development Studio free of charge. Without a Software Subscription, there is an upgrade fee. New licenses of Development Studio are priced very competitive compared to the combined prices of all constituent products. As of V5R1, there is no way to purchase the compilers or tools individually. So if you have RPG at V5R1 or higher, you must have Development Studio and hence are entitled to Development Studio Client.

For consultants who do not have an iSeries of their own, but still wish to have the client tools, Development Studio Client is also made available as a passport advantage product so it can be purchased "off the shelf" from IBM Direct.

Development Studio has been a huge success, with over 80,000 licenses sold. Just as every development machine used to have PDM and SEU, every development machine will now have all the modern Application Development tools from IBM. This ubiquity is especially important for business partners who build and sell software. These Business Partners are now free to build software using any of the technologies or tools in Development Studio, and can assume their customers will have the tools required to tailor everything from RPG to Java and Web user interfaces. This effectively raises the lowest common denominator to a level unparalleled by any other operating system.
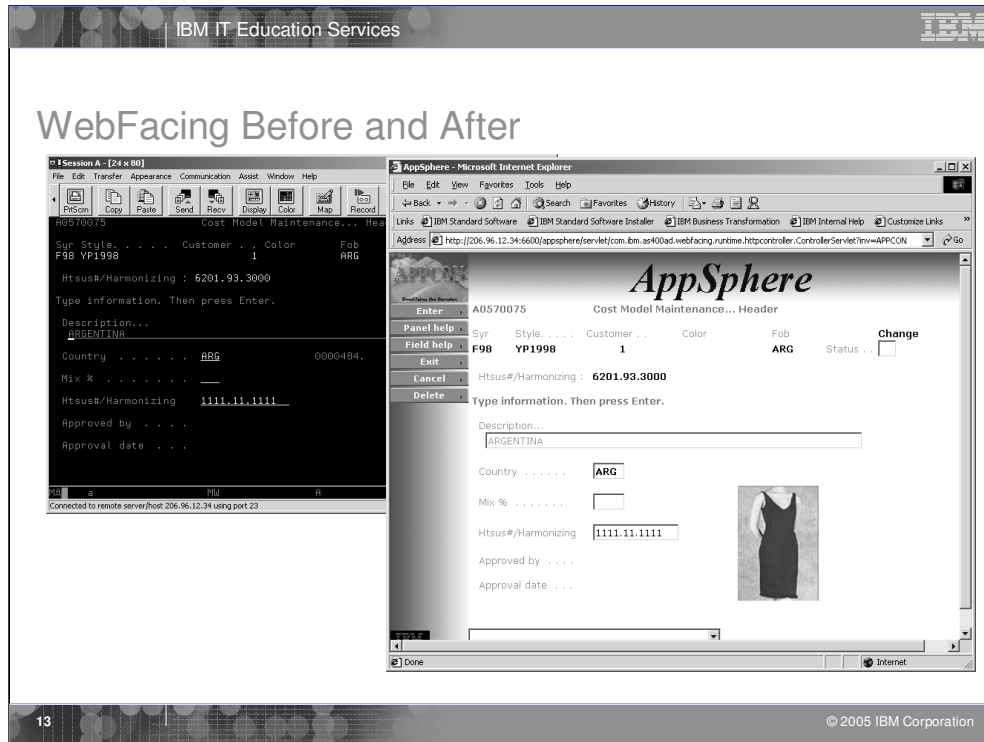
WebSphere Development Studio Client Advanced Edition 6.0

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

iSeries	iSeries	iSeries *	iSeries	Web Facing *	iSeries Projects	+CODE +VisualAge RPG
Java	Debug	Struts / Web	Web Service		RSE	
JSF	EGL Java generation	Trace	Profiling	DB	XML	App Server / HATS Toolkit
	EGL * COBOL generation	EJB * / J2EE *	Test * Cases	Portal * Toolkit		

www.ibm.com/software/awdtools/iseries

WebSphere Development Studio Client V6 based on RAD V6

There is now only one application development product sold by IBM, for iSeries, as of V4R5. This is WebSphere Development Studio (Development Studio), which includes all four host compilers, all traditional tools (ADTS = PDM+SEU+SDA+RLU+DFU+AFP+CGU), and unlimited licenses of the workstation-based toolset named WebSphere Development Studio Client (formerly WebSphere Development Tools).

If you are an existing customer who has a subscription, you can upgrade to Development Studio free of charge. Without a Software Subscription, there is an upgrade fee. New licenses of Development Studio are priced very competitive compared to the combined prices of all constituent products. As of V5R1, there is no way to purchase the compilers or tools individually. So if you have RPG at V5R1 or higher, you must have Development Studio and hence are entitled to Development Studio Client.

For consultants who do not have an iSeries of their own, but still wish to have the client tools, Development Studio Client is also made available as a passport advantage product so it can be purchased "off the shelf" from IBM Direct.

Development Studio has been a huge success, with over 80,000 licenses sold. Just as every development machine used to have PDM and SEU, every development machine will now have all the modern Application Development tools from IBM. This ubiquity is especially important for business partners who build and sell software. These Business Partners are now free to build software using any of the technologies or tools in Development Studio, and can assume their customers will have the tools required to tailor everything from RPG to Java and Web user interfaces. This effectively raises the lowest common denominator to a level unparalleled by any other operating system.

IBM IT Education Services

Table of contents

Now you know what an e-business application is and what technology it uses.
Now we introduce what WebFacing is.

WebFacing Before and After

© 2005 IBM Corporation

This slide shows a sample "before and after" of a green screen that has been WebFaced. These screens are courtesy of APPCON, an IBM iSeries business partner. See more at www.appcon4.com.
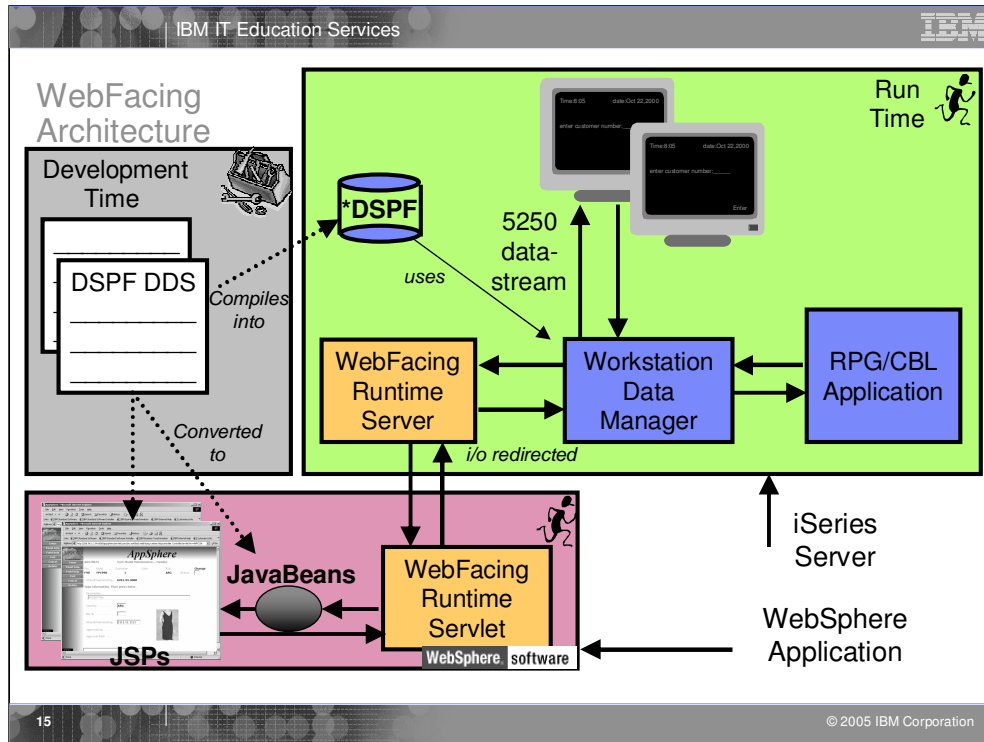
As you can see IBM's new WebFacing Tool converts existing 5250 interfaces to browser-based graphical user interfaces. With little or no modification to your original iSeries applications, you can extend the use of your programs to the Internet or an Intranet. Whether your applications are new or were written before the Internet became a viable platform for conducting business, with the WebFacing Tool, your applications can be available anywhere that users have access to a browser. You can use the WebFacing Tool with applications where DDS source code was used to create 5250 display screens. The tool has user-friendly wizards that facilitate selecting your original application's DDS source, converting the source, and deploying the new browser-based interface to your program as a WebSphere application. The conversion creates JavaServer Pages and JavaBeans that substitute for your DDS code and make Web access possible. After your DDS code has been converted, you can access the application through a browser or continue to use 5250 displays. Having the interface to your applications based on JavaServer Pages allows for more flexibility in customizing their appearance. Before your DDS code is converted, you can use the Style properties pages to change the look and feel of the pages that will be generated for you. Styles allow you to define attributes in your Web pages such as graphics, fonts, colors, and layouts. You can use one of the supplied styles or create your own. If you would like to update the appearance of a previously converted project, simply run the WebFacing Tool again and select a new style.

What is WebFacing?

- A Re-Facing tool and runtime for iSeries 5250 applications

- Development-time conversion
  - Of 5250 display file DDS source into Web page source (JavaServer Pages or JSPs)

- Run-time intercept
  - Of workstation data manager I/O to pass application data to generated Web application, prior to creation of 5250 data stream
  - Enables existing applications to run without change, indeed without even knowing they have been WebFaced

With the WebFacing Tool, you can quickly convert your DDS display file source members so that the user interface of your iSeries programs can run in a browser. When you convert your DDS display files, JSPs and Java beans are generated for you that substitute for the DDS code and make Web access possible. In the WebFacing Project wizard, you can select one or more DDS source members to convert, and select a Web look and feel from one of several predefined styles, or you can design your own Web style for use with your applications. The tool creates three Java beans and one JSP per record format; the Java beans hold the data for the record format, or control its appearance or other characteristics, and the JSP handles displaying the Web version of the screen, prompting for data, and handling input errors. The wizard generates an application home page to launch the Web-enabled version of your program.

When a user invokes a converted application from the browser, the WebFacing server on the iSeries system starts the host program. The server intercepts all calls to READ, WRITE, and EXFMT operations to DSPFs, so that in many cases your program (*PGM) can run without modifications, and without even detecting that it is being accessed using WebFacing. You might 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 Architecture

Development Time

DSPF DDS

Compiles into

Converted to

*DSPF

uses

5250 data-stream

Run Time

WebFacing Runtime Server

Workstation Data Manager

RPG/CBL Application

i/o redirected

JavaBeans

JSPs

WebFacing Runtime Servlet

WebSphere software

iSeries Server

WebSphere Application

AppSphere

15

© 2005 IBM Corporation

First the application is converted. This creates JSPs for each record format as well as Java beans. When the program performs a READ on a record format, control as well as the application data is sent to the Workstation Data Manager.. Because the job was started by the WebFacing Runtime Server, Workstation Data Manager knows this is a WebFacing request and passes the data and control to the WebFacing Runtime Server which runs on the iSeries. Control returns to the Webfacing Runtime Servlet that runs in WebSphere Application Server. The WebFacing Runtime Servlet locates the appropriate JSPs and Java beans. The WebFacing Runtime Servlet tells WebSphere Application Server to return the JSP back to the browser. The JSP is compiled and the resultant HTML is returned to the browser. Note that no 5250 datastream is generated in this flow.

Goals of WebFacing

- Quick conversion
  - Using existing skills

- Unlimited refinement
  - Using existing SDA skills, or using Web skills

- Cost effective
  - Tool part of ubiquitous tool set
  - Runtime part of operating system
  - Only pre-req is WebSphere Application Server Express or higher

You can get your RPG/COBOL programs running on the Web in a short time. Conversion is straight forward. After creating a project and specifying the DDS to convert, just choose Convert from the pop-up menu. Typically there is no change required by the host program. After the initial conversion, the new style can be applied to give the application a new look. With knowledge of SDA or JSPs and Cascading style sheets (CSS), you can modify shipped styles to create your own. You don't require other skills such as Java to WebFace an application. There are no additional costs from a previous version.

iSeries Developer Roadmap - Architecture

Traditional	Improve your Productivity	Enhance the End User Experience	Create a Modular Architecture	Integrate Applications	Integrate Business Processes

5250	5250	5250 GUI	5250 GUI	5250 GUI	5250 GUI
RPG/COBOL	RPG/COBOL	RPG/COBOL	ILE/Java	ILE	
				ILE and Java	
				Java/EJB	
	Remote System Explorer	HTML/JSP	HTML/JSP	HTML/JSP	HTML/JSP
		Servlets	Servlets	Servlets	Servlets
		Portlets	Portlets	Portlets	Portlets
		XML	XML	XML	XML
		IBM WebFacing Tool IBM WebSphere Host Access Transformation Server		Connectors	Process Choreography
				Web Services	Web Services
			DB2 and SQL	DB2 and SQL	DB2 and SQL

WebSphere Development Studio Client for iSeries

WebSphere Development Studio Client Advanced for iSeries

17

History of WebFacing

- V5R1: "First Edition"

- Version 4

- Version 5

- Version 5.1
  - Better runtime performance
  - Scalability enhancements
  - Struts support (advanced)
  - Work SpoolFile support (advanced)

- Version 5.1.2
  - Invite keyword
  - System screen support (advanced)
  - Portal support (advanced)
  - Single Signon (advanced)

- Version 6.0
  - WebSettings in workbench
  - Conversion independent of Code designer

May '01 — GA WDT V5R1

June '02 — GA WDSc V4

New iSeries models no interactive requirement for WebFaced applications

April'03 — GA WDSc V5

October' 03 — GA WDSc V5.1

July 04 — GA WDSc V5.1.2

July 05 — GA WDSc V 6.0

18

© 2005 IBM Corporation

In May 2001 WebSphere Development Studio (WDS) for iSeries—a single package containing RPG, COBOL, C, C++, ADTS and unlimited licenses of WDT was introduced. Further, it introduced WebFacing, the technology for converting green-screens to a Web interface. This phase marked the beginning of a new world where all developers had all tools for modern application development.

The long-term goal has been to collapse the many tools in WDT into a single, integrated tool that can be used for all development. The first phase of that tool was introduced when IBM announced and delivered Version 4.0 of its client tool suite for iSeries, formerly known as WDT in June 2002. With Version 4.0, the suite received a new name—WebSphere Development Studio Client (Development Studio Client). The IBM WebFacing Tool was now integrated into the Development Studio Client workbench.

In version 5.0 of Development Studio Client, additional enhancements were made to the IBM WebFacing Tool.

These include:

•Support for viewing and printing spooled files

•Struts-compliant code generated by the WebFacing Tool conversion process

•Automatic configuration for UTF-8 support when you deploy to WebSphere Application Server version 5.0

•Better performance through data compression

•Support for function keys within window records

•Enhanced hyperlink support

18

IBM

Table of contents
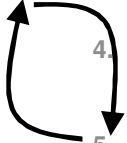
Previously we introduced Development Studio Client. Now lets look at how you create a WebFaced application, test it and deploy it.

WebFacing Scenario

1. **Create new WebFacing project**
   Specify members to convert, look-and-feel style, how to call/invoke the application
2. **Convert the project**
   Creates Web stuff from the DDS and UIM source
3. **Run it!**
   Select "Run on Server" to run in built-in WebSphere
4. **Refine it**
   Using Web Settings in CODE Designer (launched from WebFacing)
   Using project's Properties dialog
5. **Repeat**
6. **Export as EAR file, import into WebSphere Application Server on iSeries**
   Or use built-in server-tools support to publish to WebSphere Application Server on iSeries

Here is a typical WebFacing scenario. After you start the workbench and open the WebFacing perspective, you create a WebFacing project. When you create this project you select the DDS source on the server that you want to convert, the CL command to start the application and the text for the Web page link and the style you want to use for your new Web pages. Next you convert the DDS to JSPs, beans. After conversion you are ready to run your JSPs using the Run on Server option which starts the WebSphere Test Environment, a local copy of the built-in WebSphere Application Server. Now you can iteratively improve the look of your new Web pages by using the Web Settings in CODE Designer and the Web Project's Properties dialog. You can then re-test your changed WebFaced application. You can continue to refine it and retest your new Web pages until you have a design that you like. Finally you can move the new WebFaced application to a production server.
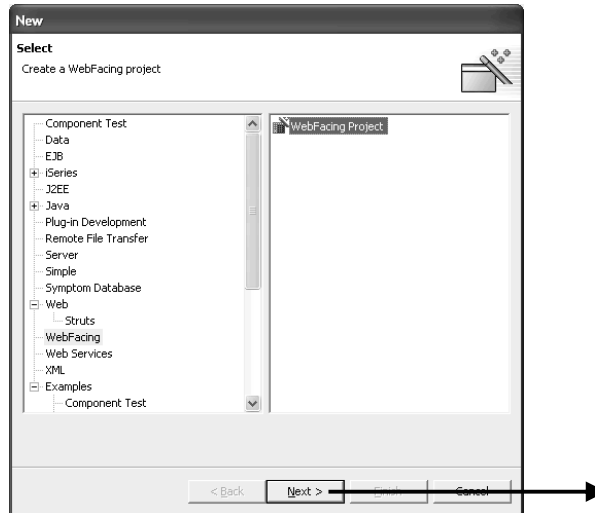
To begin, you create a new project. If WebFacing doesn't appear in the menu you select Other and then select WebFacing from the next dialog.
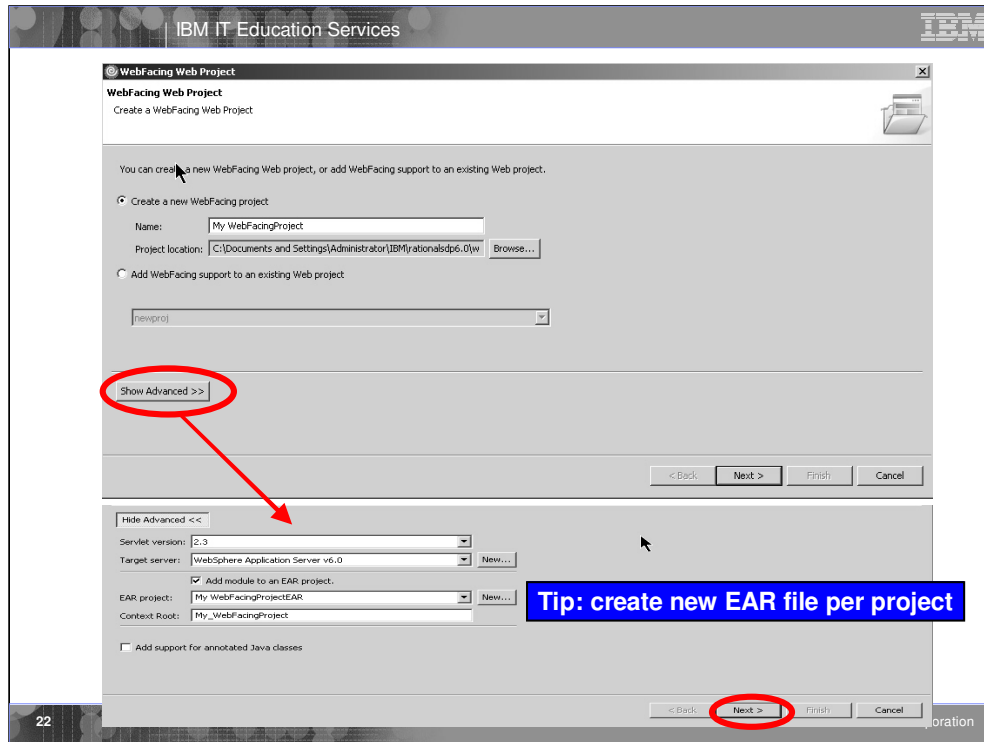
On this page of the wizard you give the project a name and specify where the files are to be stored. The default is to store them in the workspace directory. Web applications (which WebFacing is based on) are part of an EAR (Enterprise Archive) file, so we need to specify the EAR file name. The Context root is what you would specify as part of the URL in the browser to invoke the application. In Development Studio Client Advanced Edition you also can Enable Struts support. In doing so you comply with the Model-View-Controller paradigm. If you want to write a custom tag library for use with WebFacing, you would use the Generate JSP custom tags selection. This is an advanced edition feature only. What this means is that at conversion time the process of converting the DDS into JSPs can be influenced by user-written classes to change the JSP source that is emitted for fields.
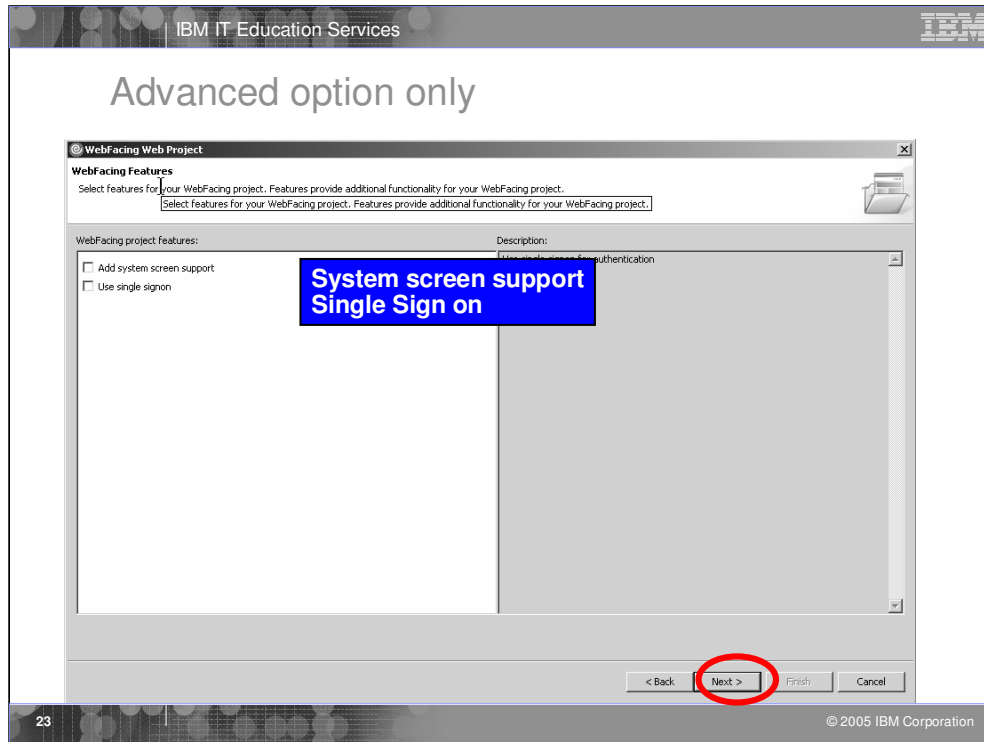
Advanced option only

System screen support
Single Sign on

© 2005 IBM Corporation

On this page of the wizard you select which connection (iSeries) contains the DDS. The member types allows you to select DSPF and/or MNUDDS (for menus). WebFacing supports menus. If you click Refresh list you will retrieve the library list, after you have signed on. From the library list you drill down to find the DDS source members you want to convert. When you click the >> push button, the selected members are moved to the list of members to be converted. If you click the folder all the members in the folder are moved to the list of members to be converted. In this case you don't need to select each member.

23

On this page of the wizard you select which connection (iSeries) contains the DDS. The member types allows you to select DSPF and/or MNUDDS (for menus). WebFacing supports menus. If you click Refresh list you will retrieve the library list, after you have signed on. From the library list you drill down to find the DDS source members you want to convert. When you click the >> push button, the selected members are moved to the list of members to be converted. If you click the folder all the members in the folder are moved to the list of members to be converted. In this case you don't need to select each member.
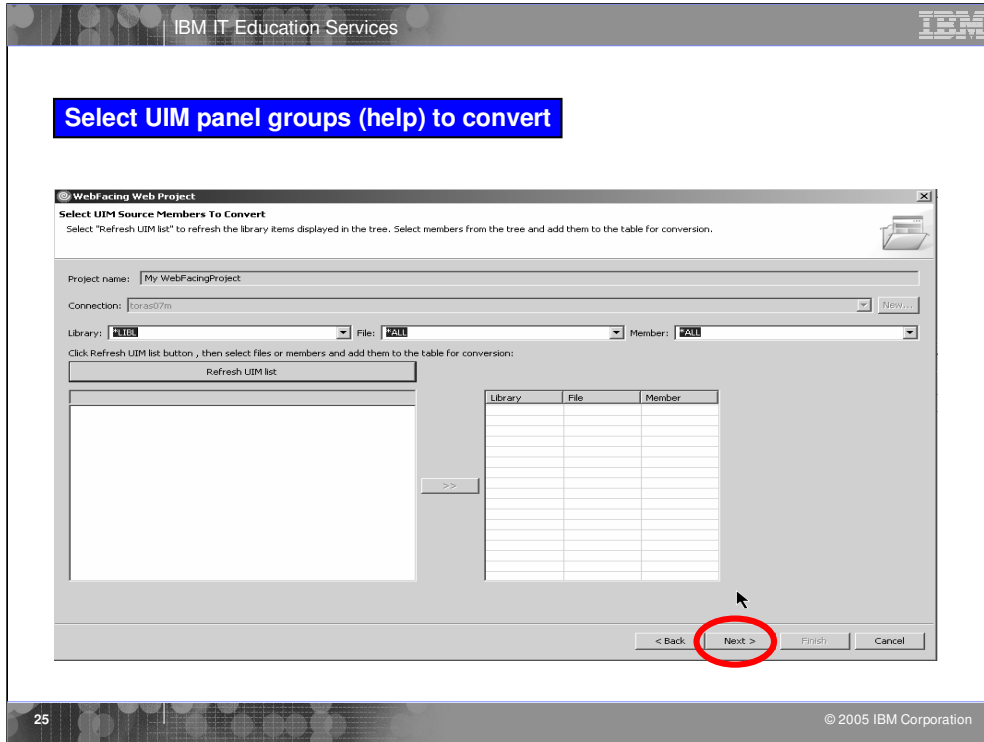
On this page of the wizard you select the UIM help members you want to convert. Again you drill down to locate the members.

WebFacing Web Project

Specify CL Commands

Enter the CL commands that are used in your application, the command labels you want to use, and the signon preference for the generated hypertext links.

Project name: My WebFacingProject

WebFacing generates hypertext links that you can use to launch your application from the web. In order to do this, it needs to know the text that will be shown for each link and the CL command that each link will invoke. On this page you can define multiple links, in case several CL commands are used to launch your application.

If your program requires parameters, you can enter them as variables in the CL command. For example: to invoke program MYPGM with a part number parameter, you would enter CALL MYPGM PARM(&part) as the CL command. The variable "&part" will be replaced when you click on the invocation link. For details, refer to the generated index.html file. The invocation name is used to uniquely identify each hyperlink in the index.html file.

CL command: Call Ordentr

Command label: Order Entry Application

Invocation name: INV1

Specify how to invoke application

○ Prompt for signon

● Specify OS/400 signon values

Specify how to signon

User ID: WEISS

Password: ******   Confirm password: ******

CL Command	Command Label	Invocation name	User ID
CALL ORDENTR	Order Entry Applica...	INV1	WEISS

Delete
Move Up
Move Down

**Can specify multiple invocation commands
- Each becomes a button in index.html**

< Back   Next >   Finish   Cancel

26

© 2005 IBM Corporation

On this page of the wizard you specify the command to call the application. The Command label text is the text that will appear on the generated Index.html page to launch the application. The CL command is an iSeries command to start the program. Typically you just do a CALL to the program. If the program has parameters you specify them here. Specifying parameters results in input fields being generated on the index.html page where the user would enter their value. If Prompt for sign-on is selected, an authentication dialog (login.jsp) is displayed before the application is launched. Alternatively you can specify a User ID and password to be passed to the WebFacing server before the program is started. Either way, the User ID and password is used to start an interactive job on the iSeries and invoke the specified command.
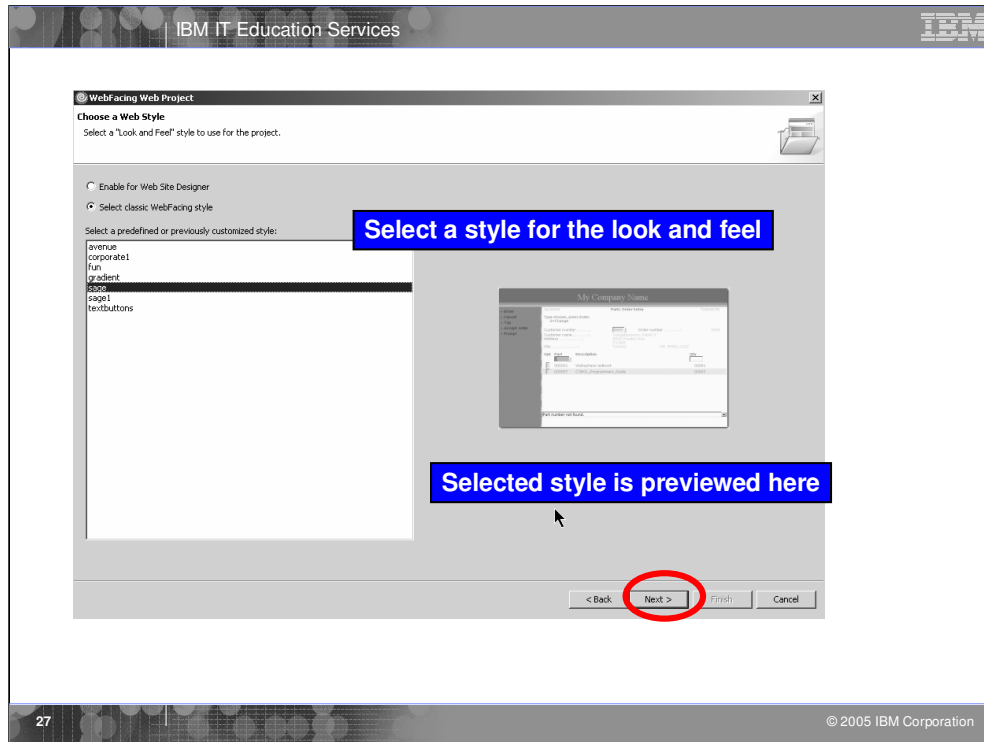
On this page of the wizard you choose the style for your application. After the project has been created another style can be applied without performing conversion. After conversion the WebFacing Tool WebSettings option or the Properties dialog allows you to customize one of the pre-defined styles and save it as a new style that can then be used by other projects.

WebFacing Web Project

Complete WebFacing Project
You have indicated the specifications for your new WebFacing project. To continue, click Finish.

You are about to create a WebFacing project.
Do you also want to convert the selected display files?

○ Yes. I want to create the project and proceed with conversion now.

● No. I only want to create the project now.

That's it! Press Finish to create project

< Back | Next | Finish | Cancel

© 2005 IBM Corporation

On this page of the wizard you choose just to create the project and perform conversion at a later time, or create the project and perform conversion when Finish is pressed. You click Finish to complete the creation of the WebFacing project

Right-click and select Convert to do conversion

Can also selectively convert individual display files

Progress Information

Converting 1 - 3.

Parsing: <toras07m>WFLABXX/QDDSSRC(ORDENTD)

Cancel

Usually 1 – 5 minutes per member
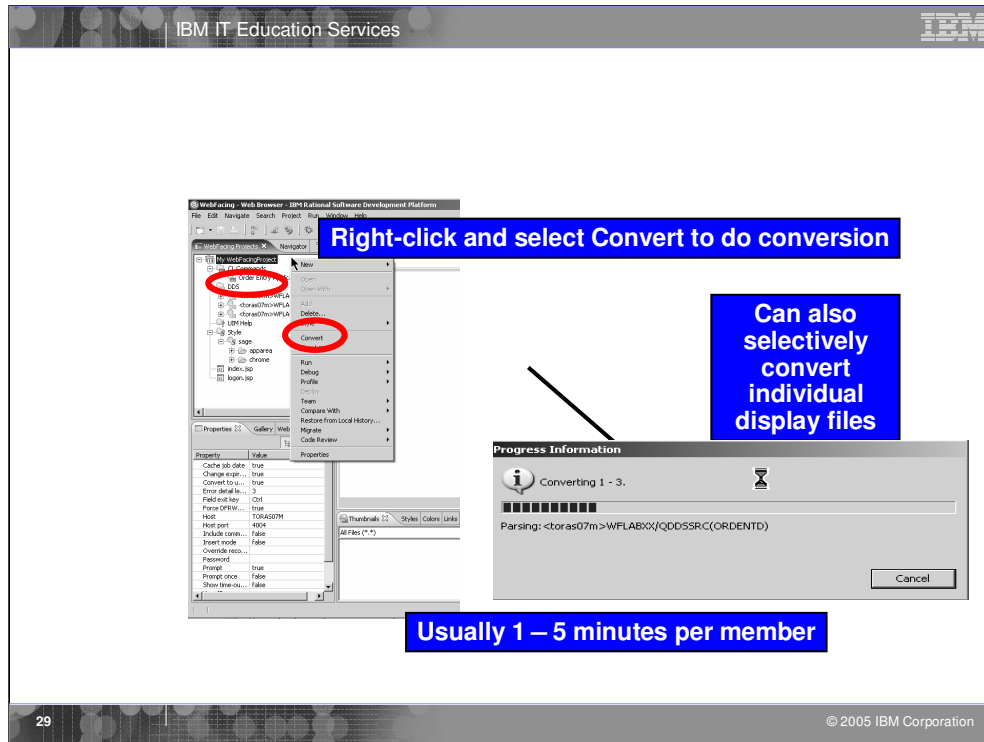
To perform conversion, right click on the DDS folder and choose convert. This will convert all members in this folder. If you made changes to a single member, you could just select that member to convert. Note that the source remains on the iSeries. The conversion process reads the DDS members and performs conversion in memory. Conversion creates 2 JSPs and 3 Java beans for each record format. It is important to convert all DDS members that are part of the application. If a member is not converted you will get a 'class not found' message at run-time.

Conversion

- Creates Java Server Page and XML descriptions per record format:
  - XML describes data,view,and feedback for record format
  - JSP displays output and prompts for input
- Creates an "invocation page" per conversion project: (index.html)
  - Web page with links
  - Each link results in starting a job, running a user-supplied CL command which starts the application

30
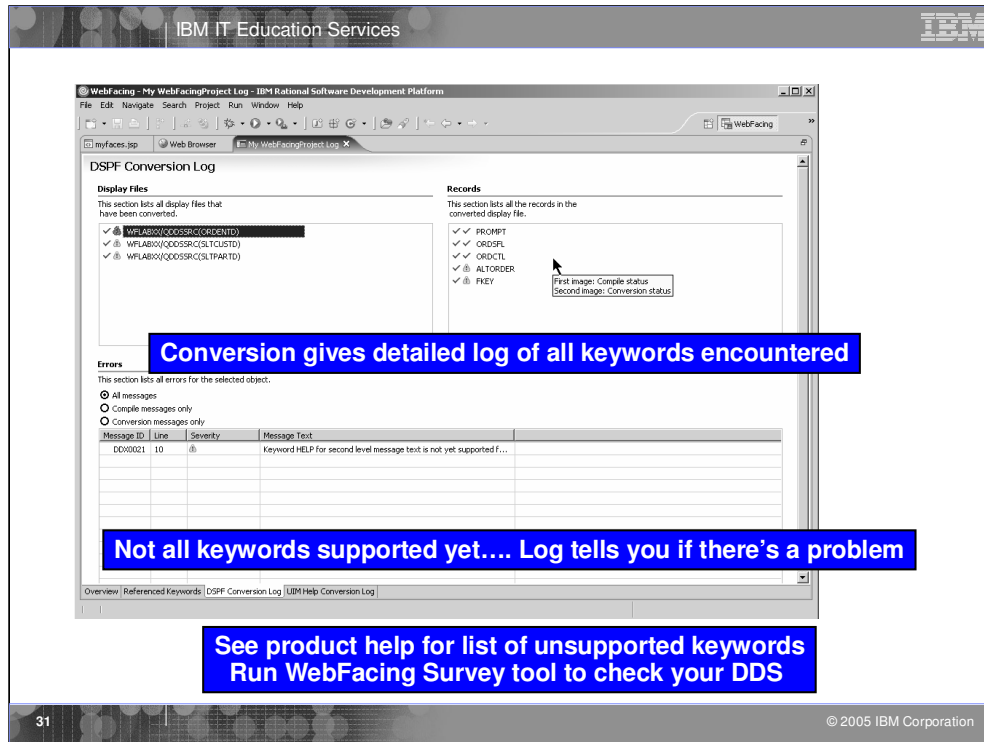
Once conversion is complete, the conversion log is displayed in its own view. Note the tabs at the bottom of the log view to see different parts of the conversion log. Review the conversion log to be sure all members were converted. A red X symbol indicates conversion failed for that member. No JSPs or beans were created. See the Message Text to determine the cause of the failure and fix as necessary.

After conversion, you can see generated files per record format

© 2005 IBM Corporation
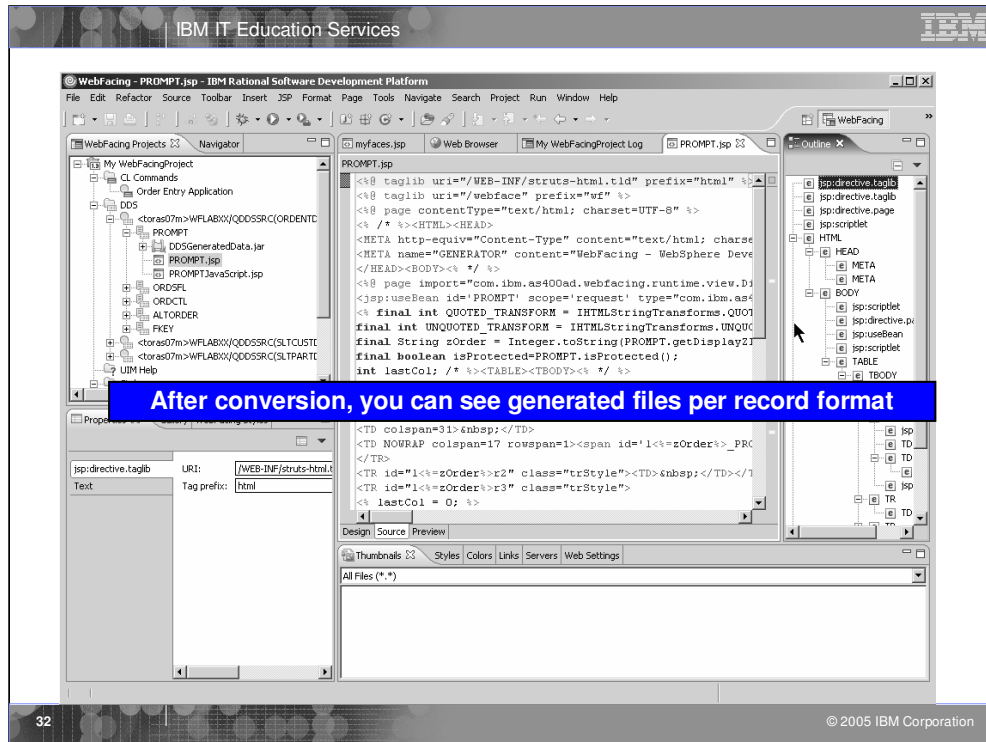
By switching to the WebFacing Projects view you can see the directory structure of the project after conversion. Note that there are 2 JSP files created for each record format. To see the generated JSP, double-click it to open it in Page Designer.

Invocation Sequence

- Generated link calls WebFacing Servlet runtime

  Extracts CALL information from web.xml

  Extracts user ID information from web.xml, or prompts for it

  Starts server job, in WebFaced mode

  Calls/invokes application

  Application runs, writing dspf formats and then reading

  Workstation data manager passing data to WF Server

  Which passes it on to WebFacing Servlet

  Which puts it in a bean and passes it to the JSPs

  All written JSPs up to the read are collected into one

  User sees screen, presses Enter

  Control goes back to WF Servlet, which passes to WDM

  WDM passes data back to application as usual

Here you see the invocation sequence. The generated link calls the WebFacing Runtime Servlet which extracts the call information from web.xml. Then the user ID information is extracted from web.xml or user ID information is prompted. The server job is started in WebFaced mode. This calls the application. The application runs and writes dspf formats and then reads. The Workstation Data Manager passes data to the WebFacing Runtime Server which passes it to the WebFacing Runtime Servlet which puts it in a bean and passes it to the JSPs. The user sees the screen and presses Enter. Control returns to the WebFacing Runtime Servlet which passes to the WorkStation Data Manager. Workstation Data Manager passes data back to the application.

How does it all work?

Tier 1

Tier 2

Tier 3

Servlet

JSPs

Beans

1. Open socket
2. Flip WM switch
3. Start job
4. Call host app

App writes,
reads data

5250
Host
app

User issues the URL:
http://server:port/web-path/index.html

JSP page output.
User enters data, and
presses Submit...

WebFacing
Runtime

WebFacing
Server

34

Here you see the invocation sequence. The generated link calls the
WebFacing Runtime Servlet which extracts the call information from web.xml.
Then the user ID information is extracted from web.xml or user ID information
is prompted. The server job is started in WebFaced mode. This calls the
application. The application runs and writes dspf formats and then reads. The
Workstation Data Manager passes data to the WebFacing Runtime Server
which passes it to the WebFacing Runtime Servlet which puts it in a bean and
passes it to the JSPs. The user sees the screen and presses Enter. Control
returns to the WebFacing Runtime Servlet which passes to the WorkStation
Data Manager. Workstation Data Manager passes data back to the
application.

Once the applcation has been converted it can be tested in the built-in WebSphere Test Environment. Locate the index.html page and choose Run on server. If not yet started, the WebSphere Test Environment will start. The index page will appear in the internal browser view.
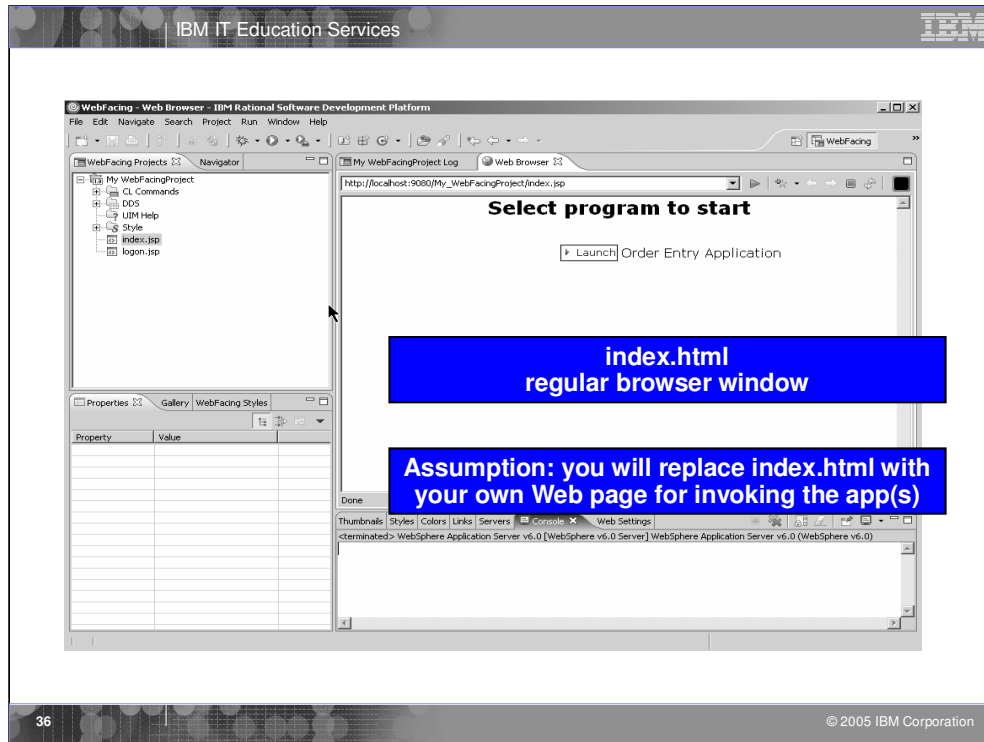
This is the index.html page that was generated based on your choices during conversion or by settings in the Properties dialog. If you click on a link, the WebFacing run-time servlet will be invoked (in WFRun.jar). Passed as a parameter is the invocation file that contains the system name and the name of the program to call. To see how WebFacing is invoked choose View/Source from the browser window.

Tip: dbl click tab to maximize

Default logon page – you can define your own
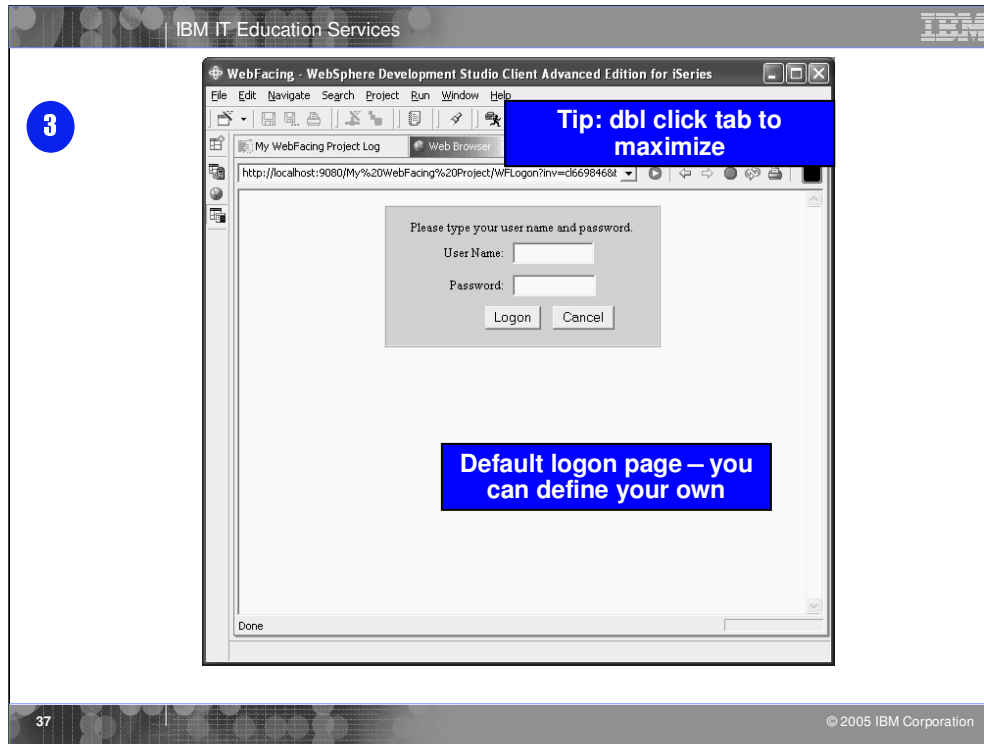
The Sign-on dialog opens. This is default sign-on page. You can design your own sign-on page.
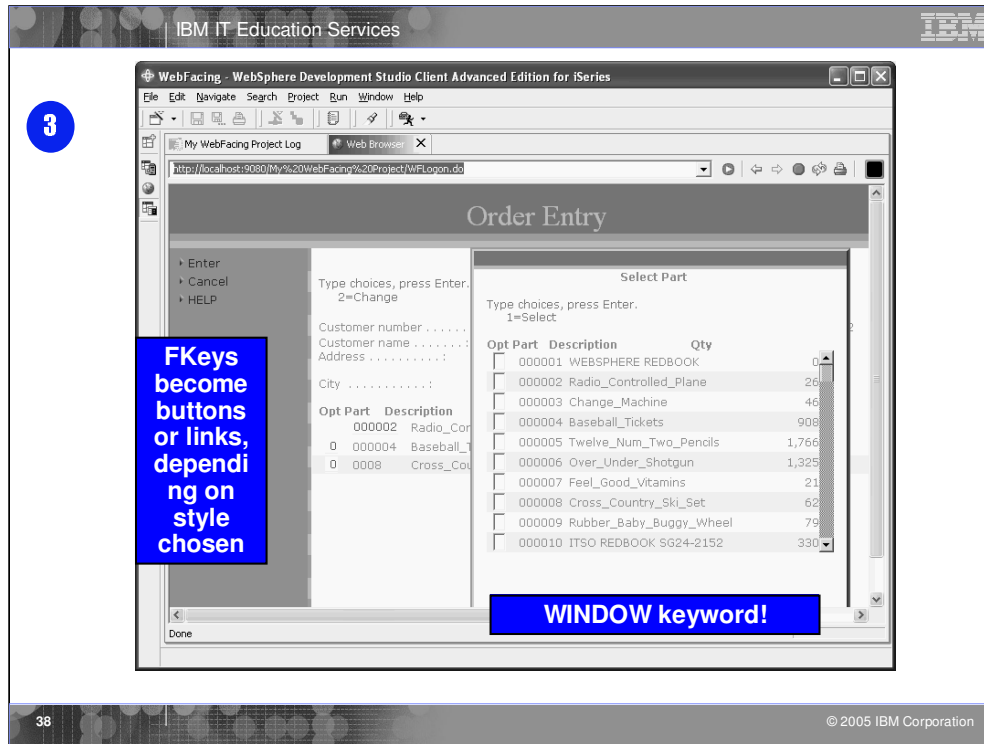
Here's the result of a WebFaced application. running in a browser. In this case, the DDS used the WINDOW keyword. In that case windows are created that can be moved around the browser. Notice that subfiles are fully supported. If you click the up/down arrows in the subfile scroll bar, this action will navigate the subfile.

6 Deploying to Remote WebSphere

- **Use File->Export to export EAR file**
    - Export to a mapped IFS drive
    - Use WebSphere Administrator Console to import EAR file
    - Bring up Browser, enter URL to start application

- **Or, configure a server in Server Tools for iSeries WebSphere Application Server** and use Publish to push to it, and then Run On Server to run it
    - This requires a PTF still coming at time of press
    - This also replaces the server's configuration file
        - So use a dedicated WAS instance for testing

- **Tip:** look at Change Management tools like SoftLanding's to automate delta deployments

Once the application has been tested and is running properly it is deployed to the production WebSphere Application Server.

Choose File Export and specify the EAR file name you specified when you created the project. Once the EAR file has been deployed, use the WebSphere Application Server Administration console to install the application in WebSphere Application Server.

Table of contents

Previously we reviewed a typical WebFacing application scenario including application testing and deployment. Now lets look at customizing a WebFaced application.

40

WebFacing Examples

- Customers/Business partners
  - Basic conversion, minimal customization
    - Gulistan Carpet
      - Sales management application
    - Astech
      - Typical Work-With application
    - GUS
      - Charisma ERP application
  - Conversion with added customization
    - APPCON
      - AppSphere ERP application
  - Conversion customization tools
    - ebt-now
      - Built additional tools to analyze application
      - Conversion results modified to achieve custom results
    - Computer Guidance Corporation
      - Tooling to improve overall Web UI

Gulistan Green Screen

Session A - [24 x 80]

File   Edit   Transfer   Appearance   Communication   Assist   Window   Help

Initial screen

Gulistan

Sales Management System

JOSEPH H. FORTE

Announcements

Courtesy: Gulistan
Carpet Inc.
www.gulistan.com

Current
announcements
listed here

MA   a   ⇧   01/001
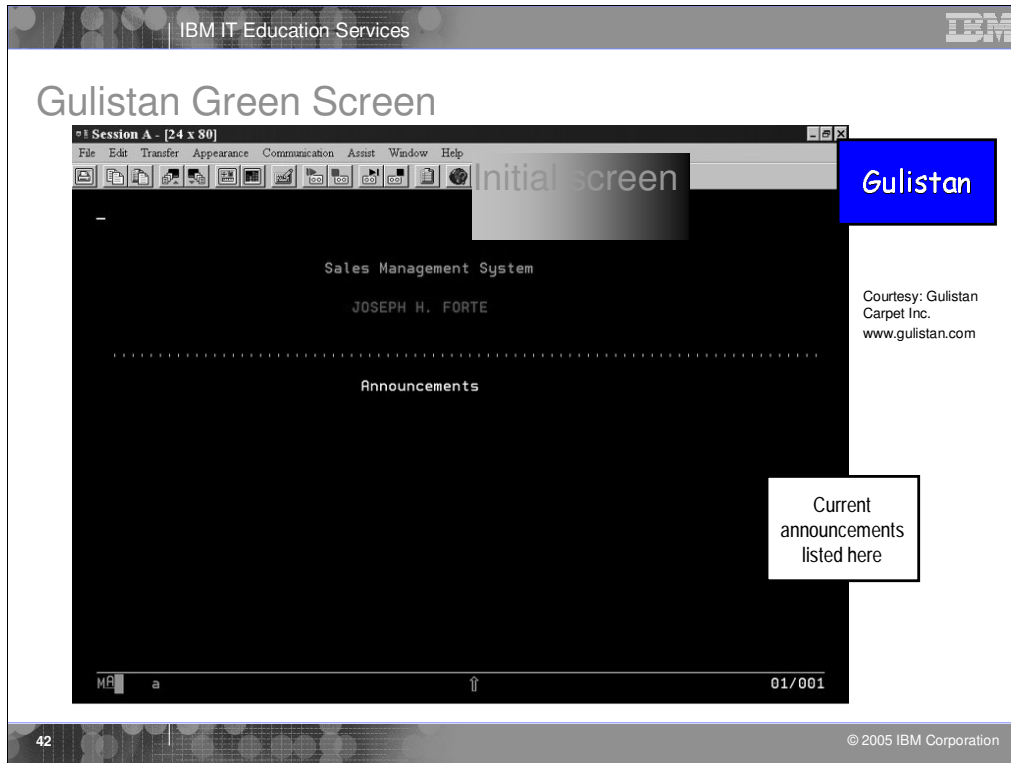
'Before' screen of Gulistan initial screen

Gulistan Web Page

Initial screen

GULISTAN CARPET

- Enter
- Customer List
- Carpet Order Detail
- Sample Order Detail
- Carpet Activity
- Sample Activity
- Carpet Sales
- Group Carpet Sales
- Log Off
- Back

Sales Management System

JOSEPH H. FORTE

Announcements

The user has added graphics to their converted application

Courtesy: Gulistan Carpet Inc.
www.gulistan.com

WebFaced version of Gulistan initial page using one of the supplied styles
Note they've customized the header by adding their own graphic

Gulistan Green Screen — Sales summary screen

Courtesy: Gulistan
Carpet Inc.
www.gulistan.com

```
Session A - [24 x 80]
File   Edit   Transfer   Appearance   Communication

 4/30/02                    Sales Summary by Salesman           11:06:09 SLF100
NORTH CENTRAL
JOSEPH H. FORTE

                            MAR 02  TO  MAR 02      NOV 01  TO  MAR 02
      Style Description     Mtd Yards Mtd Dollars   Ytd Yards  Ytd Dollars
 _   44600 NORTHFIELD SD-26       411      2,699        9,500       60,523
 _   47650 TOP AUTHORITY          402      7,701        1,181       20,512
 _   42270 FIDDLER                  0        686-       2,398       19,433
 _   49130 WORKPLACE 26           119        988        2,656       18,611
 _   45280 PERMAPOINT             598      6,568        1,747       17,532
 _   41210 CIMARON                867      9,510        1,195       14,440
 _   39000 SPECIAL 3900             0          0        1,180       13,491
 _   44620 NORTHFIELD SD-28       248      1,627        1,899       12,261
 _   45680 PRECISION                0          0          877        9,372
 _   42040 DATAWEAVE              104      1,165          640        6,434
 _   49030 WORKPLACE                0          0          748        4,600
 _   40920 CROSSCURRENT             0          0          271        4,249
 _   44630 NORTHFIELD SD-28 UN      0          0          618        4,113
 _   86510 AROUND THE BLOCK        22        376          211        3,331
 _   41590 DURAPOINT SUPREME        0          0          245        3,098  +

MA   a                                           ⇧                    07/002
```

Gulistan typical Work With Green screen

Gulistan Web Page

Sales summary screen

Here is a subfile converted

Courtesy: Gulistan
Carpet Inc.
www.gulistan.com

WebFaced version of Work With green screen
Note subfile has a slider; user clicks on up/down arrows to navigate the subfile

ASTECH Green Screen
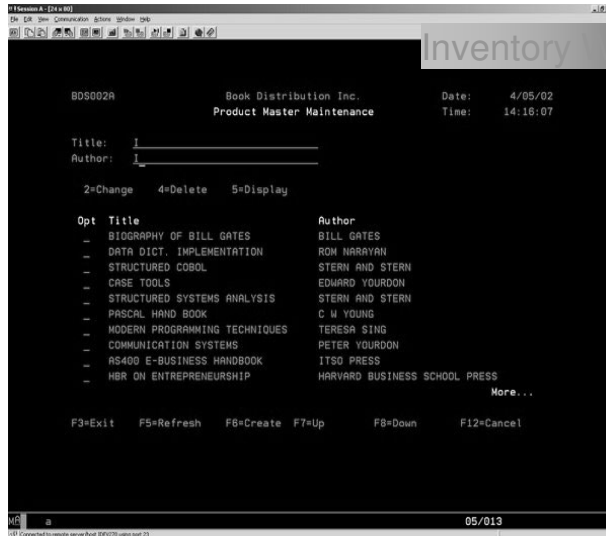
ASTECH

ASTECH offers WebFacing services!
www.astech.com

Courtesy: ASTECH Solutions Inc
www.astech.com

Astech Work With inventory green screen
Astech will provide WebFacing services to customers

ASTECH Web Page

Inventory WorkWith
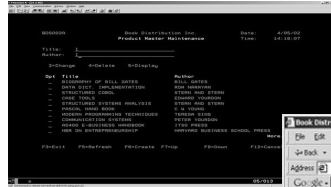
The user created their own style for conversion
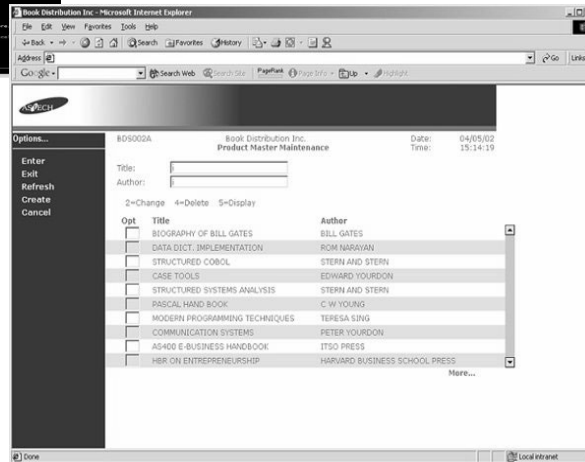
Courtesy: Astech Solutions Inc.
www.astech.com

WebFaced version of Astech's Work with green screen
Note the style is not one shipped with WDSC. This customer has created their own style

GUS Green Screen

Main menu

Courtesy: GUS Anwendungssysteme & Softwareentwicklung GmbH
www.gus-group.com

Menu screen from GUS
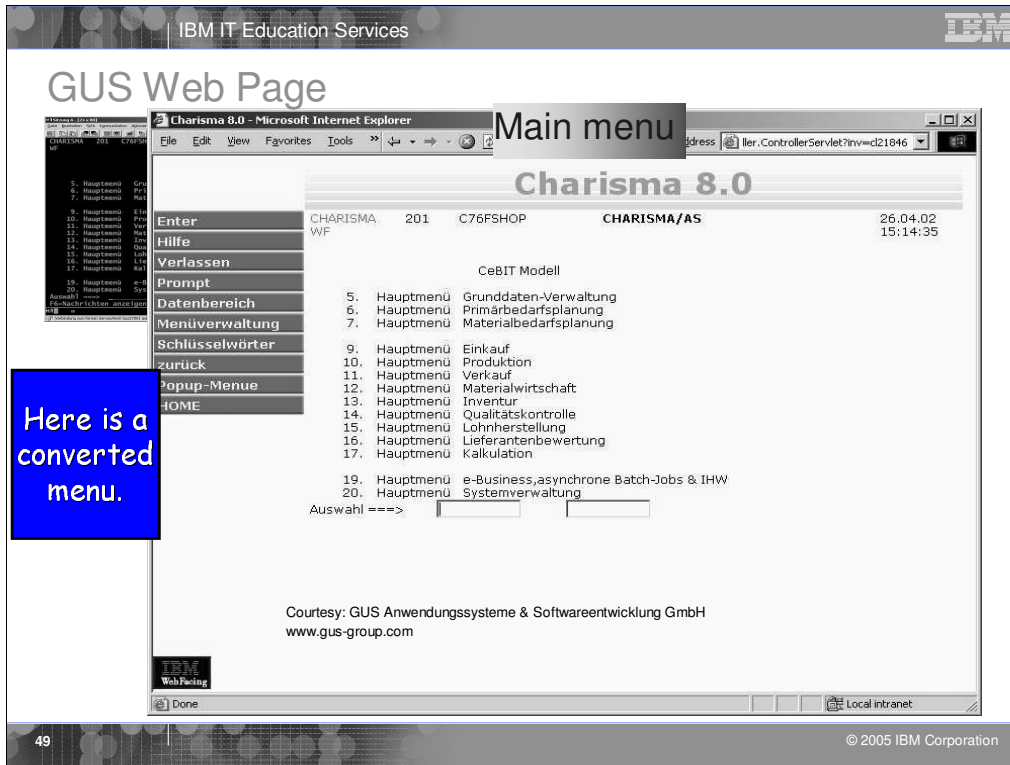
Webfaced version of GUS menu screen

GUS Green Screen

Work with or controller

Courtesy: GUS Anwendungssysteme & Softwareentwicklung GmbH
www.gus-group.com

Example of green screen using the
WINDOW keyword to display a pop-up
window

GUS Web Page

Work with for controller

Here is a converted screen with a pop-up window
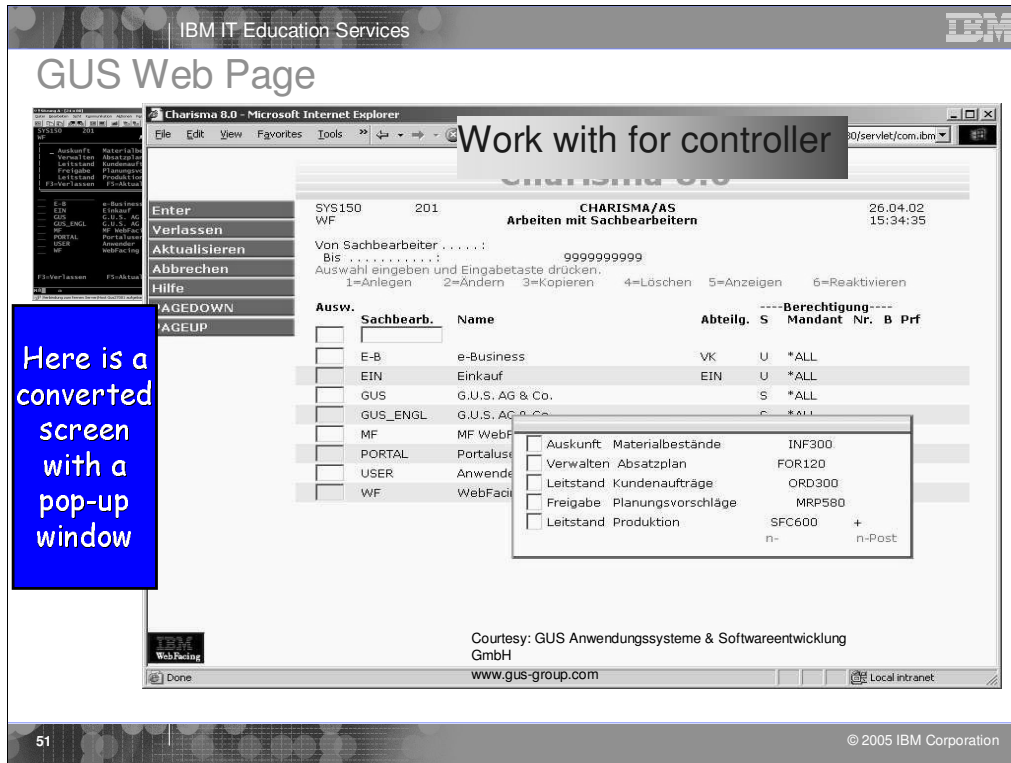
The converted screen includes a window that can be moved around the screen

Courtesy: GUS Anwendungssysteme & Softwareentwicklung GmbH
www.gus-group.com

Another example of a window in a green
screen showing a calendar in a window

GUS Web Page

Orders with calendar window

WebFaced version of screen has a window that can be moved about the screen.

APPCON Green Screen

Work with products

APPCON

APPCON offers
WebFacing
services!
www.appcon4.com

Courtesy: APPCON
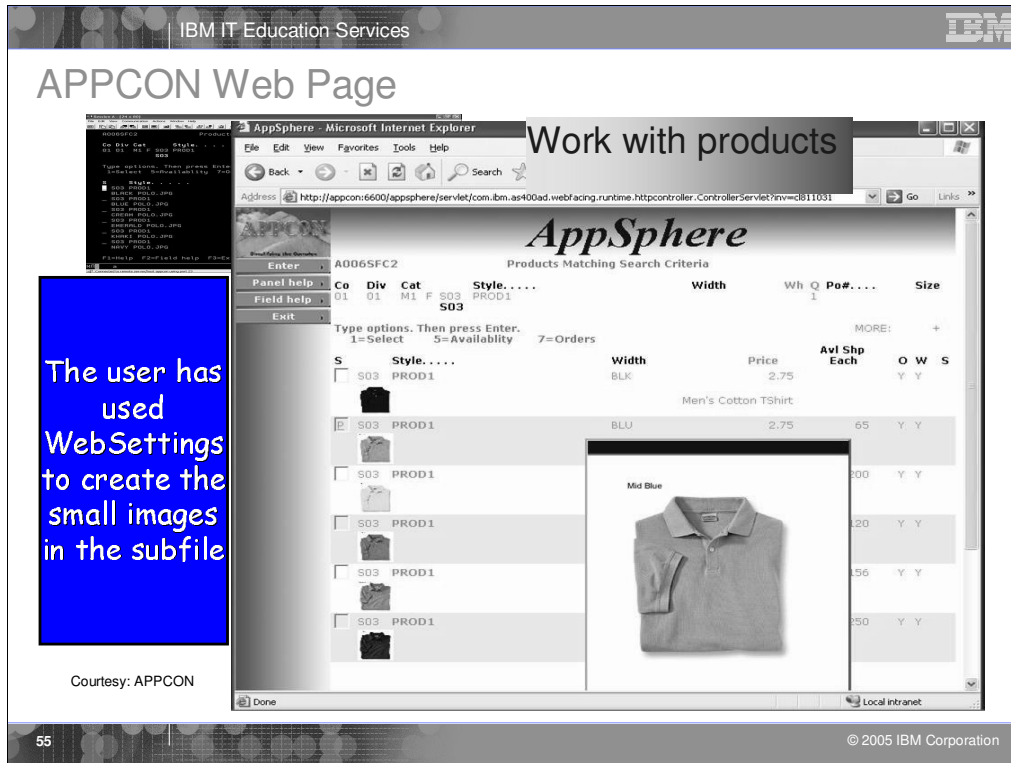
Example of green screen from AppCon
Appcon provides software for the apparel
industry

Is an early adopter of WebFacing

Also provides WebFacing services

APPCON Web Page

Work with products

The user has used WebSettings to create the small images in the subfile

Courtesy: APPCON

WebFaced version of Appcons green screen

Note that the subfile shows a small image of the item
The image was created by setting WebSettings in the DDS before conversion
The pop-up window with the larger image was generated using the Web Interaction Wizard - another component of WDSC

APPCON Green Screen

Product position summary

Courtesy: APPCON

Another green screen from Appcon showing details on an item

© 2005 IBM Corporation

WebFaced version of detail screen

Again, the image was generated by using the WebSettings feature

In WebSettings you can indicate that a field contains an image name

During conversion the correct tag is created

During runtime, the field value is added to the tag and the image is displayed

APPCON Green Screen

Bill of material detail

Courtesy: APPCON

Another screen from Appcon showing the bill of material for an item

Courtesy: APPCON

WebFaced version of Bill of Material screen

Again, WebSettings were used to create the image

Note the pop window with the radio buttons This pop up was generated because the option field had the VALUES keyword

When the user hovers the mouse over the option field the pop up is displayed

User then clicks on the option and the correct value is inserted in the option field

User can still enter directly into the option field

ebt-now Green Screen

Contract inquiry

ebt-now

```
Telnet/400 Ver 3.21  (evaluation) (C) 1998 Albert York  10.20.3.195  10/26 SA

File  Edit  Options  Auto Refresh  Macro  Help

CFC600                Inquiry forward contract position        9/12/01 18:16:12

Client: 222224 TEST CLIENT DAVID PICKERING


                        1. Forward contract position per client
                        2. Forward contract position by BBC
                        3. Forward contract position by contract

Option              :    _

Closing date        :    12/31/99
End date            :    0/00/00
Include Expired     :    N                   Y=Yes N=No

BBC Account         :    ___ _              For Choice 2
Long/Short          :    _                  For Choice 3
Executing broker    :    ___ _
Contract item       :
Due date            :    0/00/00
Amount currency     :    ___

F3=Previous   F7=Exit                                        F24=More keys
```

ebt-now
offers
WebFacing
services!
www.ebt-now.com

Courtesy: ebt-now
www.ebt-now.com

60

© 2005 IBM Corporation

Example of a green screen from ebt-now

ebt-now also offers WebFacing services

ebt-now Web Page

Contract inquiry

Main Menu

CFC600   Inquiry forward contract position   11/16/01   10:34:50

Client: 222224 TEST CLIENT DAVID PICKERING

Enter

CA03

CF05

CA07

CF08

CF24

HELP

1. Forward contract position per client
2. Forward contract position by BBC
3. Forward contract position by contract

Option	:		
Closing date	:	12/31/99	
End date	:	0 /00 /00	
Include Expired	:	N	Y=Yes N=No

BBC Account	:		For Choice 2
Long/Short	:		For Choice 3
Executing broker	:		
Contract item	:		
Due date	:	0 /00 /00	
Amount currency	:		

F3=Previous F7=Exit   F24=More keys

The user has used a supplied style for conversion

Courtesy: ebt-now
www.ebt-now.com

61

WebFaced version of ebt-now screen using one of the supplied styles

ebt-now Web Page

Ebt-now Web contract inquiry page. Here a supplied style for conversion was used.

ebt-now Web Page

Contract inquiry

The user has added graphics to their converted application

Courtesy: ebt-now
www.ebt-now.com

Customized version of ebt-now panel

Note the small calendar icon they have inserted next to the date fields

Clicking an icon causes the JavaScript calendar window to appear

Clicking a date push-button on the calendar inserts the selected date in the date field

JavaScript calendar was developed by ebt-now

CGC sample Green Screen

```
                        01 000 CGC General Contracting 340
    Date  1/21/03   XAP052      Vendor Maintenance              Time  8.19
                             Mode:  Update
    Vendor No.  2345     Reporting Vendor  2345            Status: ACTIVE
         ---- R e m i t  T o ----            ----- M a i l  T o -----
    Name    Janet                   Abrv _____   _____
    Addr 1 _____                     _____
    Addr 2 _____                     _____
    City   _____                     _____
    State     IN                                       __
    Zip         _____    County Code ___            _____
    Contact Name                      _____
    Phone                000 0000000      Fax         000 0000000
    Customer Ref No.      _____        Date Entered  02/23/2002
    Job/Sub              _____ ___       Usr Defn Ref  _____
    Fed ID (1) / SS# (2) _                I.D. Number   _____
    1099 Code            _                Alt Vendor    _____
    Vendor Type          _                Minority Code  0
    Terms Code           AA               Disc Pct (2)    _____
    Freight Amt(2)       _____         G/L Expense   0000.00000.000 _____
    Subcontract Vendor   Y                Avg Chk Clr Days  ___


     F3=Exit F4=Prompt  F10=Balances   F12=Previous   F22=Deactivate F23=Delete
    MA    a                                                    24/080
    Connected to remote server/host S1003A2D using port 23
```

Courtesy: Computer Guidance Corp

WebFacing without enhancements

Courtesy: Computer Guidance Corp

65

WebFacing — with CGC tool

Courtesy: Computer Guidance Corp

WebFacing — Survey tool

- •Downloadable from the web
- •Install from savefile
- •DDS source in libraries gets scanned
- •Report lists
  - •Keywords supported
  - •Keywords not supported

WebFacing — Survey tool

```
Viewer - QSYSPRT/1/QPADEV000X/865301/WEISS/TORAS07M
File  Edit  View  Search  Notes  Options  Help

WEBFACING REPORT FOR IBM Canada -- Libraries searched: WFLABXX

For latest DDS Keywords support information, please go to the following Web Site:
http://www-4.ibm.com/software/ad/wdt400/webfacing/ddsref/rwfkwd.htm

NOTE: When this report talks about the future keywords support, it is just a draft plan.
The plan may be altered according to customer requirement, design requirement...etc.
IBM does NOT commit to support all these keywords in the specified release as the following report.


WDS V5R1 GA  04/12/2001      KEYWORDS
Keyword                            Count
                                   -----
ALIAS                                  0
ALTHELP                                6
ALTNAME                                0
ALTPAGEDWN                             0
ALTPAGEUP                              0
CAnn/CFnn                             14
CHANGE                                 0
CHECK(RB)/AUTO(RAB)                    0
CHECK(RZ)/AUTO(RAZ)                    0
CHECK(other)                           0
CHGINPDFT(ME MF LC CS)                 0
CHGINPDFT(RI HI)                       0
CLEAR                                  0
CMP                                    0
CNTFLD                                 0
COLOR                                 20
COMP                                   0
CSRINPONLY                             0
DFT                                    0
DFT(implicit)                         50
DFTVAL                                 0
DLTCHK                                 0
DLTEDT                                 0
DSPATR(UL HI RI ND PR CS)             14
EDTCDE(other)                         14
EDTWRD                                 0
ERASE                                  0
ERRMSG                                22
ERRSFL                                 2
```
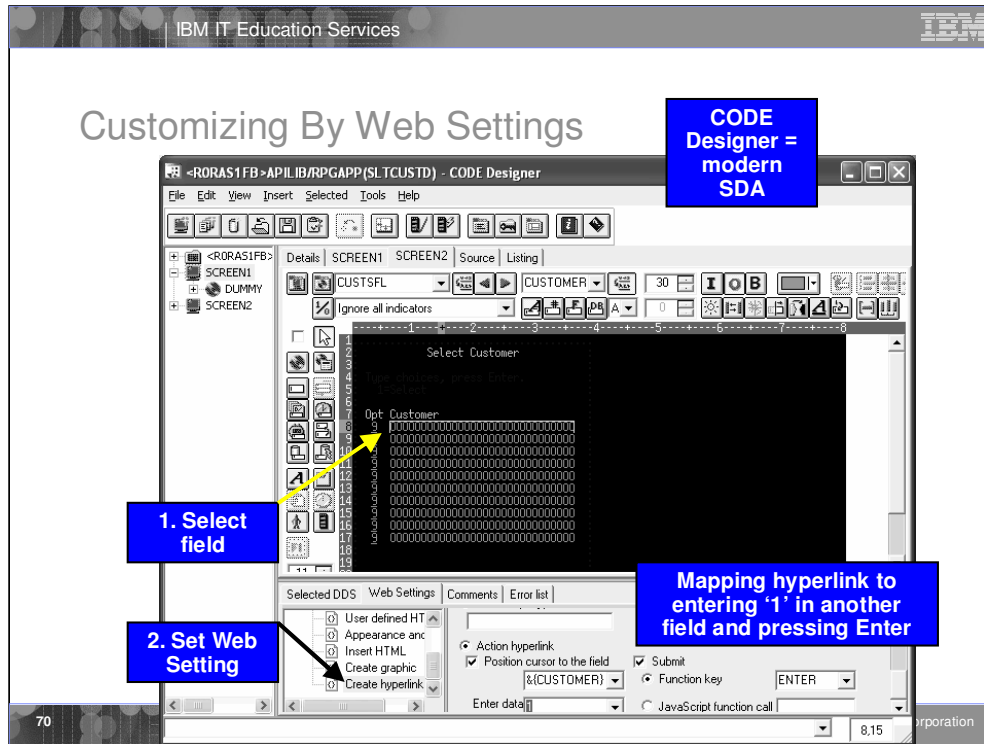
```
1 of 5     125%     SFLVIEW     No group is selected
```

3 Ways to Customize WebFaced Applications

1 By Properties

Much can be affected in the conversion and runtime, including overall style, by use of properties and style wizard

2 By Web Settings

These are DDS comments for affecting the conversion results. Easy to set with the CODE Designer tool

3 By Editing the Output

Because of the built-in Java and JSP editors, you can always edit what WebFacing generates

But be careful….

There are 3 ways to customize WebFaced applications.

1. Before conversion, use properties dialog to change conversion, runtime and style properties.

2. Before conversion use the WebSettings feature of CODE designer. WebSettings are stored as comments in the DDS. With WebSettings you may have certain fields that may not be relevant on the Web. You can specify that a field contains an image name, and an <IMG> tag should be created for that field. You can specify that a field is a hyper-link. Clicking on the link at run-time will launch the URL specifed in WebSettings.

3. After conversion, WebFacing copies the Cascading Style Sheet (CSS) to the project. With some knowledge of CSS you can modify it to customize your pages. You can use Page Designer to edit the created JSPs. Note, conversion will overwrite customized JSPs

Customizing By Web Settings

After a DDS source member has been loaded into CODE Designer, Web Settings can be accessed by clicking the icons in CODE Designer's DDS Tree. The DDS Tree is located on the left-hand side of the CODE Designer window. Web Settings are also accessible by selecting DDS objects from within CODE Designer's **Details** and **SCREEN** tabs. If Web Settings are available for the selected object, the Web Settings tab will be displayed in the bottom pane of CODE Designer.

After a DDS source member has been loaded into CODE Designer, Web Settings can be accessed by clicking the icons in CODE Designer's DDS Tree. The DDS Tree is located on the left-hand side of the CODE Designer window. Web Settings are also accessible by selecting DDS objects from within CODE Designer's **Details** and **SCREEN** tabs. If Web Settings are available for the selected object, the Web Settings tab will be displayed in the bottom pane of CODE Designer.

This book, the first in the IBM Press Guided Tour series, describes how to get started with IBM's new WebFacing tool and how to make progress with WebFacing efforts.

The systematic approach presented here takes readers through WebFacing a sample green screen application using IBM WebSphere Development Studio Client. It also demonstrates how to use the different features in the tool to enhance the application after the initial conversion. Even debugging strategies are discussed.

To maximize the speed and completeness of your learning experience, the author takes a practical, hands-on approach to explaining WebFacing features and how to use them. This approach, gained from involvement in defining the capabilities of the WebFacing tool allows you to benefit from both the step-by-step nature of the book as well as the most comprehensive treatment offered on WebFacing training.

The book's many exercises will teach you a variety of features in the WebFacing tool and the WebSphere Development Studio Client product. The relationship between these products will also be explained as well as the reason for each of the various exercises.

New Redbook

The IBM WebFacing Tool
Converting 5250 Applications to Browser-based GUIs

SG24-6801-00
www.ibm.com/redbooks

Follow this complete guide and easy
examples to use the IBM WebFacing Tool

Web enable existing host applications
on the IBM @server iSeries server

Use existing skills to
create a Web interface

Aleksandr Barlovich
Dean Henkel
Steve O'Shannessy
Barry Owen

Redbooks

ibm.com/redbooks

This redbook explains how the application conversion works and how you can customize your Web interface. It explains how to work with JavaServer Pages and cascading style sheets. It also offers helpful performance considerations and troubleshooting tips. Plus it looks at how to deploy WebFaced applications to Apache Tomcat.

This redbook is written for iSeries application developers who want an easy way to extend the life of existing host applications. It also applies to those who want to expand the reach of existing host applications to a wider set of clients.

New Redbook

IBM@server

Draft Document for Review July 2, 2003 12:40 pm

WebSphere Development Studio Client for iSeries V5.0

Learn a new way of building the applications for iSeries

Bring your 5250 applications to the Web with the IBM WebFacing tool

Catch up with the latest trends in the application development

Aleksandr Nartovich
Wilfried Blankertz
John Dell'Oso
Pentti Rasanen
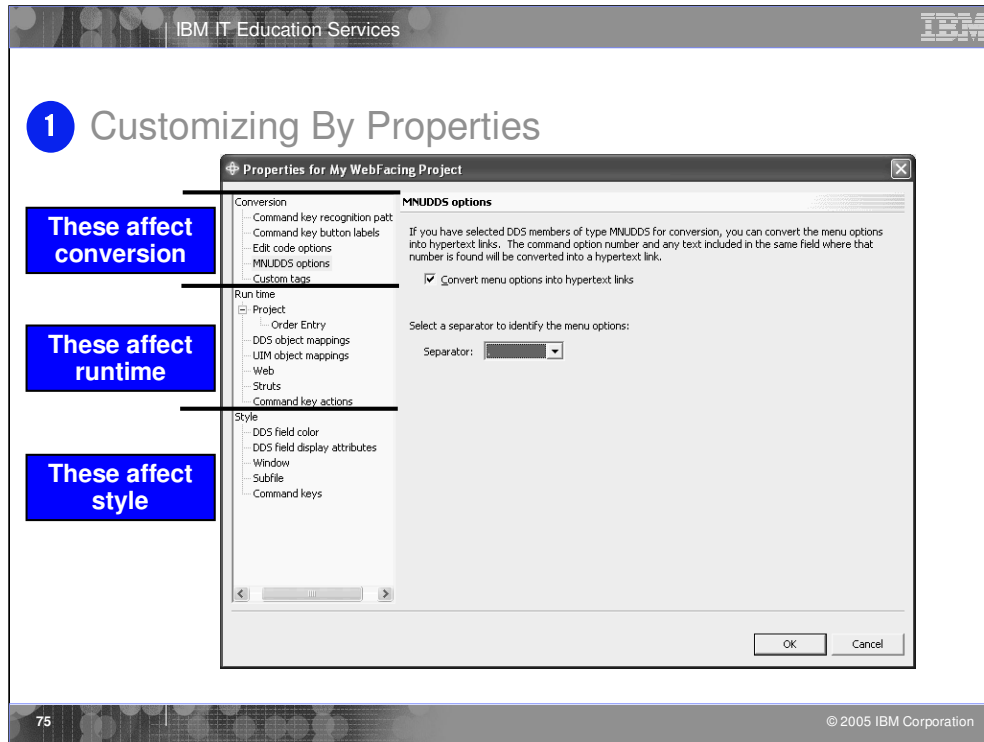Claus Weiss

ibm.com/redbooks

Redbooks

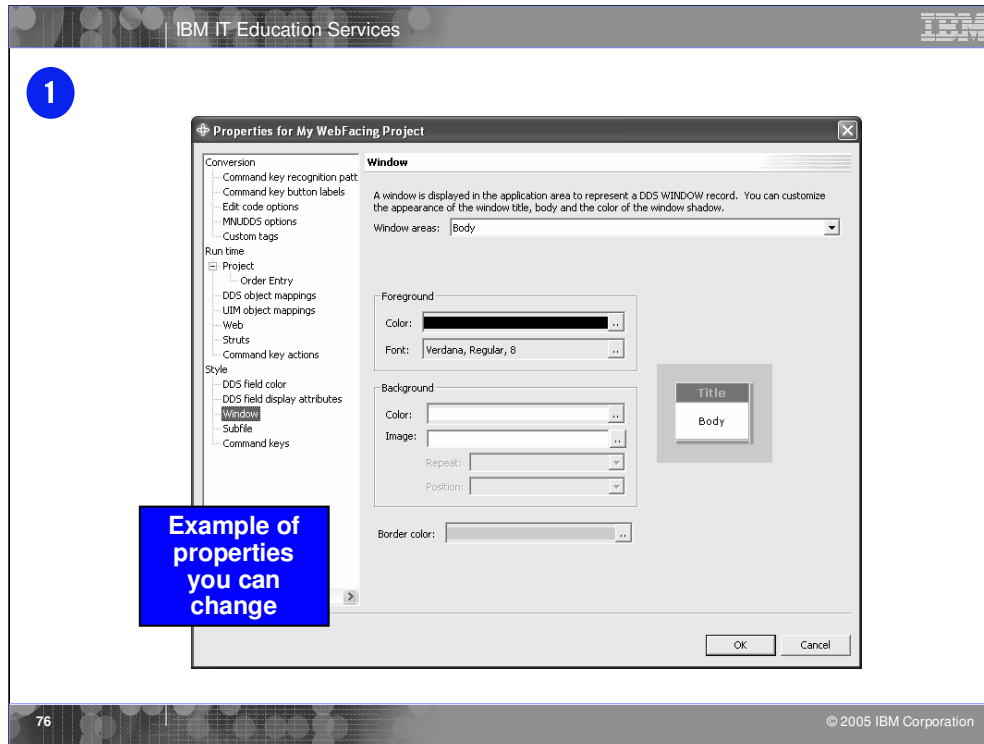SG24-6961-00

www.ibm.com/redbooks

Java and e-business are key to the future of the iSeries server. Web-enabling your 5250 applications allows you to quickly participate in the e-business arena by using existing applications and programming skills.

This Redbook discusses WebSphere Development Studio Client for iSeries V5.0 of which the WebFacing tool is a part. Development Studio Client includes several powerful tools that target the iSeries application developers:

- WebSphere Studio Site Developer Advanced (in WebSphere Development Studio Client Standard Edition for iSeries V5.0) or WebSphere Studio Application Developer (in WebSphere Development Studio Client Advanced Edition for iSeries V5.0) - the new IDE for developing applications. The unique characteristic of this IDE is the ability to add new features in the form of the plugins. Anyone can develop a new plugin and install it into the tool without creating a "plumbing" infrastructure. WSSD and WSAD include the development environment for creating plugins.

- Development Studio Client includes several iSeries specific features, like the IBM WebFacing Tool and the iSeries Web development tools, that are installed as plugins.

- Cooperative Development Environment (CODE) - a workstation-based tool that supports the development of the applications in many different host languages, including RPG and Java.

- VisualAge RPG - If you are already an experienced RPG IV programmer, you can create graphical user interfaces to RPG programs very quickly in VisualAge RPG.

- Integrated iSeries Debugger.

74

1 Customizing By Properties

Properties for My WebFacing Project

These affect conversion

These affect runtime

These affect style

Conversion
— Command key recognition patt
— Command key button labels
— Edit code options
— MNUDDS options
— Custom tags
Run time
⊟ Project
    ⋯ Order Entry
— DDS object mappings
— UIM object mappings
— Web
— Struts
— Command key actions
Style
— DDS field color
— DDS field display attributes
— Window
— Subfile
— Command keys

MNUDDS options

If you have selected DDS members of type MNUDDS for conversion, you can convert the menu options into hypertext links. The command option number and any text included in the same field where that number is found will be converted into a hypertext link.

☑ Convert menu options into hypertext links

Select a separator to identify the menu options:

Separator:

OK   Cancel

© 2005 IBM Corporation

To edit the properties for a WebFacing project, in the WebFacing Projects view, right-click the icon for your Project, CL commands, DDS, UIM Help, or Style folder and select **Properties**. Conversion properties control how selected DSPF and UIM files are being converted for WebFacing use. Values for conversion properties are stored in the file conversion.rules under the config directory of the WebFacing project. Run-time properties determine the behavior of the converted Web application when it is being used by an end user. You can customize the look of the application area and the command keys using the Style properties. If you want to change the look of the layout and frame surrounding these areas, you must use a CSS editor to update the style files stored in the chrome directory.
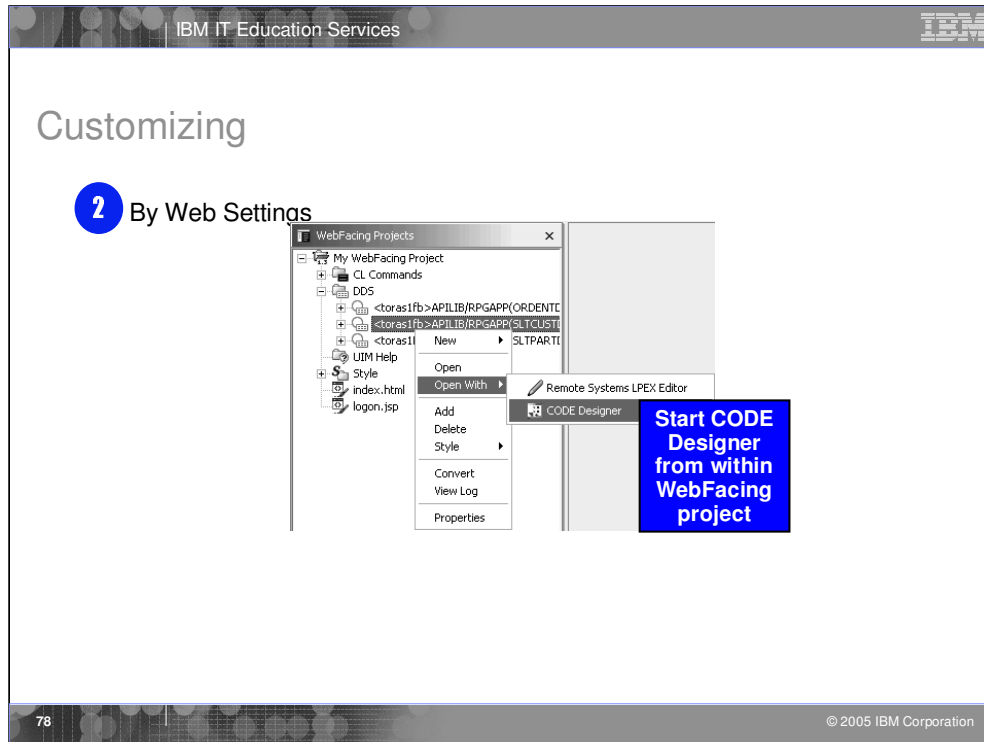
1

Properties for My WebFacing Project

Conversion
 Command key recognition patt
 Command key button labels
 Edit code options
 MNUDDS options
 Custom tags
Run time
 Project
  Order Entry
 DDS object mappings
 UIM object mappings
 Web
 Struts
 Command key actions
Style
 DDS field color
 DDS field display attributes
 Window
 Subfile
 Command keys

Window

A window is displayed in the application area to represent a DDS WINDOW record. You can customize the appearance of the window title, body and the color of the window shadow.

Window areas: Body

Foreground

Color:

Font: Verdana, Regular, 8

Background

Color:

Image:

Repeat:

Position:

Border color:

Title
Body

Example of properties you can change

OK   Cancel

Here you see an example of changing the style property Window. You use the Window screen to indicate how you would like DDS WINDOW records to look when they are converted for Web use. You can customize the look of the title, body, and shadow of the window.
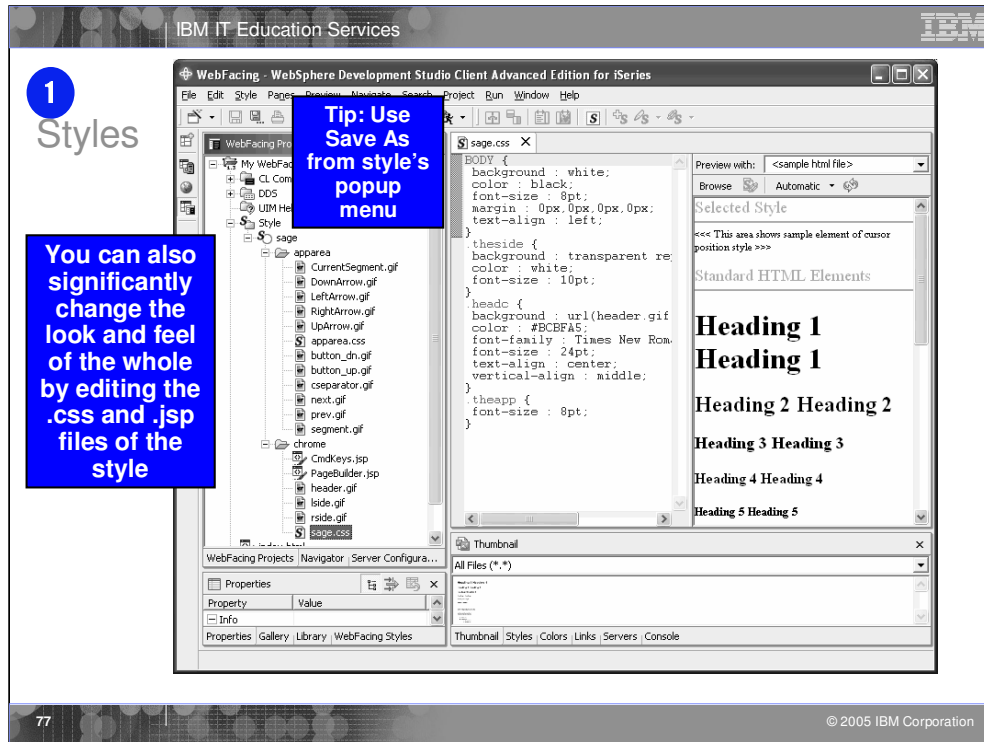
If you want to edit the style that is used for the application area and command key elements, you can use the Style properties pages or you can edit the apparea.css file directly using tools supplied in the IDE. The Style properties pages make it easier to visualize the modifications that are being made and shows you how these changes apply to the DDS elements such as window and subfile records. The changes made through the Style properties pages are then applied to the corresponding style class names in the apparea.css file.

If you want to edit the layout and the frame surrounding the application area and command keys, edit the user-defined CSS files in the \chrome directory. To use the IDE's CSS editor, right-click the file that you want to edit and choose **Open with** --> **CSS Designer**. If you want to edit the layout of the frame, edit the file PageBuilder.jsp. To edit the frame style, right-click **Style** --> **Edit Style**. PageBuilder.jsp can then be edited using the Page Designer tool supplied by the IDE. The CSS file in the \chrome directory can be edited using the CSS Designer in the IDE. The Web perspective can be useful when editing CSS files. To open the Web perspective, select **Window** --> **Open Perspective** --> **Other** --> **Web**.

Customizing

2 By Web Settings

Start CODE Designer from within WebFacing project

© 2005 IBM Corporation

You can use the Web Settings tab in CODE Designer to customize how your programs will look and function when accessed through a Web browser. Programs that use DDS source to describe 5250 display screens can be accessed with a Web browser after the DDS source has been converted with the WebFacing Tool. Use Web Settings when you want to manipulate the Web presentation of individual screens and individual fields within screens. If you want to change the Web presentation of multiple screens or of an entire WebFacing project, use Style properties.

Web Settings enable you to affect how your pages will appear before they are created using the WebFacing wizard. If you would like to modify your pages after conversion, you can use a text editor or a Web design tool such as WebSphere Studio. An advantage to customizing your pages with Web Settings is that the instructions for the customizations are embedded as comments in your DDS source; since Web Settings become part of your source, changes that you make are not lost if you later reconvert a WebFacing project.

78
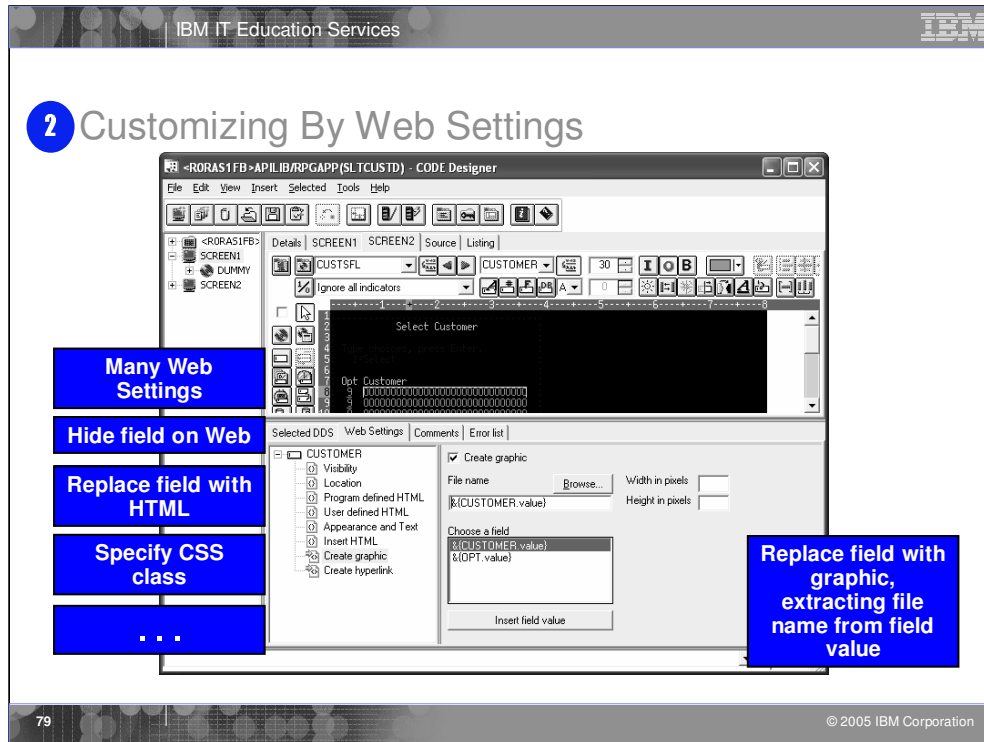
CODE Designer allows you to design DDS screens graphically. In CODE Designer, each DDS object is represented by an icon or push button. For example, there are push buttons so that you can easily create named fields or text constants for your DDS screen. The Web Settings available for each DDS object vary depending on the object that you are working with. The online help lists the Web Settings available for each DDS object. In the online help refer to the section of this document *Web Setting descriptions* for more details on each setting. When you use Web Settings for an object, special comments are added to your DDS source which later get processed by the WebFacing conversion. Web Setting comments begin with the characters *%%WB.

Here you can see the results of customizing your WebFaced application style using Web Settings. 1st column heading is hidden, new prompt text appears and the 2nd column heading is a hyperlink.

3 Customizing By Editing Output

If you want to edit the layout and the frame surrounding the application area and command keys, edit the user-defined CSS files in the \chrome directory. To use the IDE's CSS editor, right-click the file that you want to edit and choose **Open with** --> **CSS Designer**. If you want to edit the layout of the frame, edit the file PageBuilder.jsp. To edit the frame style, right-click **Style** --> **Edit Style**. PageBuilder.jsp can then be edited using the Page Designer tool supplied by the IDE. The CSS file in the \chrome directory can be edited using the CSS Designer in the IDE. The Web perspective can be useful when editing CSS files. To open the Web perspective, select **Window** --> **Open Perspective** --> **Other** --> **Web**.

Changes to the styles made through the property pages or by direct editing apply only to the current project. To save them for use in other projects, right-click **Style** --> **Save as** and give your style a name. Then this named style becomes available for selection the next time you choose a Web style during project creation or the next time you select a style to replace the current project style using right-click **Style** --> **Select Style**.

Summary

e-business Primer

WebSphere Development Studio Client

What is WebFacing?

WebFacing Tools

Samples and Customizing WebFacing

Summary

This presentation reviewed what e-business was all about. Then we looked at WebFacing and what it is. Next we introduced Development Studio Client. The steps to WebFace an application were described in detail. WebFacing customization was described followed by a review of the new WebFacing Tool Version 5.0 features.

83

Disclaimer

- **Acknowledgment:**

  This presentation is a collaborative effort of the IBM Toronto iSeries Application Development presentation team, including work done by:

  Claus Weiss, Phil Coulthard, George Farr, Don Yantzi, Satish Gungabeesoon

- **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 & 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 or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

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

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.

iSeries WebFacing