Developing WebFacing Applications

Version 9 Release 5
Developing WebFacing Applications

Version 9 Release 5
Before using this information and the product it supports, read the information in "Notices" on page 213.
Contents

Chapter 1. Introducing the WebFacing Tool ........................................... 1
  What’s new in WebFacing ......................................................... 2

Chapter 2. Setting up the IBM WebFacing Tool .................................... 5
  Setting up WebSphere Application Server on your IBM i server .......... 5
  Installing the WebFacing server PTFs on your IBM i server. ............. 6
  Starting the WebFacing runtime server on the IBM i server .............. 6
  Transferring converted applications to the IBM i server ................. 6
  Starting and stopping the WebFacing server on your IBM i host ....... 7
  Configuring the WebFacing server for multiple interactive subsystems 7
  Changing the port used by the WebFacing server ............................ 9
  Enabling licenses ........................................................................ 9

Chapter 3. Creating a WebFacing Web project ...................................... 11
  Opening the WebFacing perspective ............................................ 12
  Choosing a Web style ................................................................ 12
  Selecting the source members to convert .................................... 13
  Specifying CL commands for invoking your application ................. 14
  Finishing the project and converting your DDS source ................. 15
  Analyzing the conversion logs .................................................. 15
  Working with WebFacing project resources ............................... 16
  Handling non-WebFaced IBM i applications using dynamic data stream transformation ......................................... 17
  WebFacing performance considerations within Eclipse workbench .... 18

Chapter 4. Creating a HATS/WebFacing enabled Web project .................. 21
  Linking a HATS/WebFacing enabled project with a HATS project ....... 21

Chapter 5. Customizing your WebFacing applications .......................... 23
  Editing properties of a WebFacing project .................................... 23
    Conversion properties ............................................ 24
    Command Key Recognition Patterns ............................... 24
    Key Button Labels ................................................... 24
    Edit Code Options .................................................. 25
    MNUDDS Options ................................................... 25
    XML Record Metadata ............................................. 26
    Run Time properties .............................................. 26
    Project Properties .................................................. 26
    <CL command> properties ......................................... 29
  DDS Object mappings .................................................. 29
  UIM Object Mappings ................................................. 29
  Java EE ................................................................. 29
  Command Key Actions .................................................. 30
  System Screens ........................................................ 30
  Style properties ........................................................ 30
  DDS Field Color ........................................................ 31
  DDS Field Display Attributes ......................................... 31
  Window ...................................................................... 32
  Subtile ...................................................................... 33
  Command Keys ........................................................ 33
  Changing the port used by your WebFacing applications ............... 34
  Extending your WebFacing application with command key actions .... 35
  Labels for command and function keys ..................................... 36
  Setting authentication options ............................................. 36
  Deployment descriptor .................................................... 45
  Working with WebFacing styles ............................................ 46
  Creating and modifying classic WebFacing styles ....................... 47
  Using the WebFacing Styles view ......................................... 48
  Editing a Web Site style ................................................. 49
  Web Site Designer style example .......................................... 51
  Web Settings ............................................................. 51
  Using Web Settings with your DDS source .............................. 52
  Editing Web Settings for a DDS member ............................... 54
  Web Settings descriptions ................................................ 54
  Web Settings considerations ............................................... 71
  Creating custom JavaScript library files .................................... 73
  WebFacing Accessibility .................................................. 73
  Bidirectional user interfaces in WebFacing ............................... 77
  Setting tab order for bidirectional applications ......................... 79
  Querying user access to your WebFacing application .................. 80

Chapter 6. WebFacing interoperability with HATS and other Web applications . 83
  WebFacing interoperability with HATS applications ..................... 84
    Considerations for linked HATS/WebFacing applications .......... 84
  Using the WebFacing application bridge to interoperate with other Web applications .................................. 86
  Programmatically invoking WebFacing applications from other Web applications .................................. 89
  URL parameters that can be determined dynamically .................. 90
  Filtering programmatic invocation commands .......................... 91
  Example URLs .................................................................. 91
  Servlet methods for calling a WebFacing application programatically ......................................................... 92

Chapter 7. Testing and deploying WebFacing applications .................... 93
  Testing WebFacing applications ........................................... 93
  Deploying WebFacing applications ...................................... 94
  Deploying to WebSphere Application Server .......................... 94
Chapter 1. Introducing the WebFacing Tool

The IBM® WebFacing Tool converts existing 5250 interfaces to browser-based graphical user interfaces. With little or no modification to your original IBM i 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 source members, converting the source, and deploying the new browser-based interface to your program as a Web application. The conversion creates an index page, JavaServer Pages, and XML files that substitute for your original source code and make Web access possible. After your source 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. When you convert your source code, you select a Web style to determine the look and feel of the pages that will be generated for you. You can further customize the style after converting your source code using the Style properties pages. Styles allow you to define attributes for 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.

After conversion you can make further changes to your application’s user interface by editing the DDS members using the Web Settings view in the IDE and reconvert the changed members when you’re done. You can also enhance the generated index.jsp file using Page Designer. With Page Designer, you can further change the style, add graphics, update page properties, and create a more customized look for your Web-enabled application.

You can also use Rich Page Editor to enhance the generated index.jsp file. With Rich Page Editor, you can change the style, add graphics, update page properties, and create a customized interface for your Web-enabled application.

The following diagram show the high-level tasks and files involved in Web-enabling your applications:
Related tasks:
- Chapter 2, “Setting up the IBM WebFacing Tool,” on page 5
- “Opening the WebFacing perspective” on page 12
- Chapter 3, “Creating a WebFacing Web project,” on page 11
- “Selecting the source members to convert” on page 13
- “Specifying CL commands for invoking your application” on page 14
- “Choosing a Web style” on page 12
- “Finishing the project and converting your DDS source” on page 15
- “Analyzing the conversion logs” on page 15

Related information:
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

What’s new in WebFacing

About this task

New supported environments

HATS V9.5 extends the supported application development environment to
Rational® SDP V9.0. Supported WebFacing runtime environments now include WebSphere® Application Server V8.5.5.

HATS V9.5 supports Struts 2.3.16.3.

HATS V9.5 does not support IBM Portlet API. Portal API in WebFacing V9.5 has been removed.

WebFacing V9.5 supports Google Chrome.

Related reference:

“WebFacing support for multiple browsers” on page 96

WebFacing projects can be run in the Microsoft Internet Explorer, Mozilla Firefox, and Google Chrome browsers. This reference document describes the differences between the presentation of WebFacing pages in Internet Explorer and Firefox browsers.

HATS Information Center
Chapter 2. Setting up the IBM WebFacing Tool

About this task

The WebFacing Tool provides a GUI environment for selecting, converting, and testing the converted user interfaces of your original application.

Setting up your systems for the WebFacing Tool involves the following activities:

- Setting up WebSphere Application Server on your IBM i server
- Installing the WebFacing server PTFs on your IBM i server
- Starting the WebFacing server on the IBM i server
- Transferring converted applications to the IBM i server.

Related concepts:

- Chapter 1, “Introducing the WebFacing Tool,” on page 1

Setting up WebSphere Application Server on your IBM i server

About this task

WebSphere Application Server handles executing the JavaServer Pages and JavaBeans that are generated as part of the DDS conversion process. Set up WebSphere Application Server for IBM i using the documentation resources on the following Web site:

IBM WebSphere Application Server - Express V8.5 for IBM i

http://www-01.ibm.com/software/webservers/appserv/was/library/v85/e-ibmi/index.html

Minimally you will need to carry out the installation steps. Becoming familiar with the IBM WebSphere Application Server documentation, in particular the sections on installation and creating application server instances, is highly recommended.

You can optionally set up individual instances for the WebSphere Application server. If your system is to be used for test and production work, it can be helpful to create instances for testing and development and separate instances for production. As well, if more than one developer is doing development work, their work can be developed and tested separately by creating additional instances. The default HTTP port for WebSphere Application Server is port 9080. The default port for the administrative server is port 9060. The administrative servers are accessed through a browser-based console. If you want to create additional instances, please refer to the Web site links above.

Test the WebSphere Application Server setup. Use the links at the above sites to locate the appropriate Information Center. In the Information Center search on Verify the installation for information about testing your application server setup.
Installing the WebFacing server PTFs on your IBM i server

About this task

The WebFacing server acts as a gateway between WebSphere Application Server and your original IBM i application. Information about the WebFacing server PTFs can be found in Chapter 9, “Troubleshooting WebFacing applications,” on page 107.

The WebFacing server PTFs must be installed on the IBM i server where your application will be running. Except on some later-model IBM i servers, the WebFacing Tool consumes interactive cycles. Although WebSphere Application Server can run on the same system, it is not necessary for it to be running on the same system on which your application is running. The converted interface to your application is deployed as a WebSphere application; you can deploy the WebFacing WebSphere application to one IBM i server while the IBM i application is installed on another.

Starting the WebFacing runtime server on the IBM i server

Procedure

1. Start the WebFacing runtime server on your IBM i server. For OS/400® V5R2 and later, enter `STARTCPSVR SERVER(*WEBFACING)` from the command line. To stop the WebFacing server, enter `ENDTCPSVR SERVER(*WEBFACING)` from the command line.

2. To ensure that the server is running, enter `WRKACTJOB` from the command line.
   a. From a command line enter `WRKACTJOB JOB(QQF*)` to list all WebFacing jobs:
      - `QQFWFSVR` -- the WebFacing runtime server AND
      - `QQFVTSVR` -- the WebFacing virtual terminal server (More than one QQFVTSVR job may be running, which is normal)

Note: Check the system value QAUTOVRT. If this value is 0 then no browser sessions can be launched in place of VT (virtual terminal) jobs. The WebFacing runtime server relies on VT jobs. The value for QAUTOVRT determines the number of virtual terminal jobs that can be auto started. If necessary, set QAUTOVRT to *NOMAX or some value greater than 0. To change the value for QAUTOVRT, use the WRKSYSVAL IBM i server command to work with system values.

Transferring converted applications to the IBM i server

About this task

Application development is done on a workstation installed with Windows Operating System. Artifacts generated by the application must later be transferred to an IBM i server so they can be deployed by WebSphere Application Server. Set up a method for transferring your converted applications to your IBM i server. When you are deploying your converted applications to your IBM i server, you will need a method to transfer them to your server. Applications are transferred as .ear files, which are archived Web application packages created through the Export wizard. They facilitate deploying and installing new WebFacing Web applications to WebSphere Application Server. These are the primary methods for transferring files:

- Remote System Explorer perspective in the Rational Developer for IBM i workbench
• IBM i NetServer file sharing -- With file sharing, files can be transferred using the Export wizard.
• FTP -- With FTP, files are transferred manually.

NetServer file sharing allows you to map an IFS folder on your IBM i server to a drive letter on your workstation. This allows you to use the Export wizard to transfer files to your server in the same way that you can copy files from one folder to another on your workstation. To transfer your files by FTP, the FTP server must be running on your IBM i server. If FTP is not running, it can be started by entering the command STRTCP$VR *FTP.

Starting and stopping the WebFacing server on your IBM i host

Procedure
1. Start the WebFacing server on your IBM i server. Enter the CL Command STRTCP$VR SERVER(*WEBFACING) from the command line. To stop the WebFacing server, enter ENDTCP$VR SERVER(*WEBFACING).
2. To ensure that the server is running, enter WRKACTJOB from the command line.
   a. From the command line, enter WRKACTJOB JOB(QQF*) to list all WebFacing jobs.
      • QQFWFSVR -- the WebFacing server AND
      • QQFVTSVR -- the WebFacing virtual terminal server (More than one QQFVTSVR job may be running, which is normal.)

Note: Check the system value QAUTOVRT. If this value is 0 then no browser sessions can be launched in place of VT (virtual terminal) jobs. The WebFacing server relies on VT jobs. The value for QAUTOVRT determines the number of virtual terminal jobs that can be auto started. If necessary, set QAUTOVRT to *NOMAX or some value greater than 0. To change the value for QAUTOVRT, use the WRKSYSVAL IBM i server command to work with system values.

Related tasks
Chapter 2, “Setting up the IBM WebFacing Tool,” on page 5
“Configuring the WebFacing server for multiple interactive subsystems”

Configuring the WebFacing server for multiple interactive subsystems

About this task

Depending on the number of users that you have and the number of WebFacing jobs that are created on your system, you may want to configure additional interactive subsystems to handle WebFacing jobs. Using multiple interactive subsystems improves the scalability of WebFacing by:
• Increasing the total number of WebFacing jobs that can be run on your machine.
• Load balancing: WebFacing jobs are assigned in a distributed manner so that they are spread optimally among configured subsystems.

Up to 16 additional subsystems can be configured for WebFacing. If no other subsystems are configured, by default, WebFacing jobs run in the QINTER subsystem. Assess the need for using additional subsystems based on your knowledge of how many users you have and the capabilities of your hardware. Keep in mind that an interactive job is created for each user logon whether the
logon is through a 5250 session or a WebFacing session. Note, when additional subsystems are configured for WebFacing, QINTER is no longer used for WebFacing interactive jobs.

WebFacing interactive jobs are named by using a device name convention QQFn* where n represents which subsystem is being used in a sequence of up to 16. The value for n follows hex numbering conventions. In hex, the first 10 values for n are represented with the numbers 0 through 9 and the next six values (values 10 to 15) are represented with the letters A through F. For example, names for jobs in the first subsystem will begin with QQF0*, names for jobs in the twelfth subsystem will begin with QQFB*. If no additional subsystems are configured, WebFacing interactive jobs can be identified in the QINTER subsystem by names beginning with the string QQF0*.

Procedure
1. If necessary, modify the QINTER subsystem so that WebFacing jobs are no longer processed there. This is required if *ALL is the value currently used in the Work station name field for Work Station Entry values in QINTER. Use the DSPSBSD command to view your current settings. If required, replace *ALL with the display device name convention used for your system. For example, some systems use QPADEV*.

2. Create additional interactive subsystems for WebFacing jobs. These subsystems should be modelled after QINTER. Create these subsystems by creating subsystem descriptions using the CRTSBSD command.

3. When the WebFacing server is installed, a data area called QQFCONFIG is created for it in the library QQFTEMP. Change the SBS value in the QQFTEMP/QQFCONFIG data area to the number of interactive subsystems to be used for WebFacing. In this data area, a default name/value pair of SBS=01; is created to define the number of interactive subsystems to be used for WebFacing. To change the default value, use the command WRKDTAARA DTAARA(QQFTEMP/QQFCONFIG). Choose option 2=Change and then in the New value field enter SBS=nn; where nn is the number of subsystems you will be using for WebFacing. This can be from 1 to 16. For example, SBS=16; Although you enter the value in decimal notation, at runtime the SBS number is displayed in hexadecimal format. If you want to display your current values, use the command DSPDTAARA DTAARA(QQFTEMP/QQFCONFIG).

4. Using the ADDWSE command, add work station entries for WebFacing for each additional subsystem that you are going to use. Populate the Work station name field of the Add Work Station Entry screen with the WebFacing device name to be used for that subsystem. Follow the hex numbering conventions described earlier to determine what values to use. Hex values for the first 10 subsystems are 0 to 9. Hex values for the next six subsystems (10 to 15) are A to F. For example, in the ADDWSE screen for the first subsystem, enter QQF0*. For the second subsystem, enter QQF1*. For the sixteenth subsystem, enter QQFF*.

5. For these changes to take effect, stop and then start the WebFacing server. To stop the WebFacing server, use the command ENDTCPVR SERVER(*WEBFACING). To start the server, use the command STRTCPVR SERVER(*WEBFACING).
Changing the port used by the WebFacing server

The WebFacing server acts as a gateway between WebSphere Application Server and your original IBM i application. The WebFacing server will listen on a predefined TCP port to exchange data with WebFacing applications. All WebFacing applications connecting to the server on the same IBM i need to be configured to connect to the assigned TCP port. By default, the server listens on port 4004. This value can be changed using the IBM i WKSRTBLE command to modify the TCP service as-WebFacing. The server needs to be restarted for the change to take effect and all WebFacing applications need to be reconfigured accordingly in the runtime properties of the WebFacing projects and redeployed.

Changing the WebFacing runtime port on your IBM i server

1. Access the Work with Service Table Entries screen. From an IBM i server command line, enter the command: WRKSRVTBLE. The Work with Service Table Entries screen is displayed.

2. Page down to the as-WebFacing service.

Note: If you have not redefined the WebFacing port manually before, you will not see an entry for as-WebFacing. If you have not previously manually defined a WebFacing port, skip to the step Add a new entry for as-WebFacing.

3. Select option 5=Display to display the port information for the WebFacing server. Unless you have changed the port, the default is 4004. Press Enter or F12 to return to the Work with Service Table Entries screen.

4. To change the port, you first need to remove an existing entry for as-WebFacing, if you have one, by selecting option 4=Remove.

5. Add a new entry for as-WebFacing.
   a. Access the Add Service Table Entry screen. Select option 1=Add. The Add Service Table Entry screen is displayed.
   b. In the Service field, between the single quotes, enter as-WebFacing.
   c. In the Port field, enter the new port number that you would like to use.
   d. In the Protocol field, between the single quotes, enter tcp.
   e. Press Enter. Your new WebFacing service entry is added.

After you have changed the WebFacing runtime port on your IBM i server, follow the steps in "Changing the port used by your WebFacing applications" on page 34 to change the port used by your WebFacing applications so your WebFacing applications will work properly with the new port.

Note: This port is used only for communications between the WebFacing application and the WebFacing server, it is not the port that you may specify on the browser to access your WebFacing Web application.

Enabling licenses

Before you begin

You must purchase the IBM Rational Host Access Transformation Services (HATS) licenses to enable your WebFacing applications for use in a production environment. Without a valid license, you can run up to two concurrent trial sessions of your WebFacing applications.
To fully enable the WebFacing runtime for production in accordance with your licensed proof of entitlement, you must specify your license settings using the License Settings wizard.

**Note:** You must run the License Settings wizard even for projects whose runtimes were fully enabled in previous versions.

### About this task

To run the License Settings wizard, follow these steps:

1. Open the WebFacing perspective.
2. Right-click anywhere inside the WebFacing Projects view, and select **License Settings**.
3. On the Runtime Enablement File panel, use the **Browse** button to browse for the location of the file.
   - The name of the file used to fully enable HATS Web, HATS rich client, and WebFacing projects is `runtimeenablement95-webrcp.jar`. The wizard enables both the HATS and WebFacing runtimes in one step.
   - You must use the file intended for use with your version of WebFacing. An error message will display if you try to use a file for an older version.
4. Select the file and click **Open**.
5. After setting the location of the runtime enablement file, click **Next**.
6. On the License Options panel the types of projects are displayed with an indication whether they will be enabled or not enabled based on the selected runtime enablement file. Select the license option specified by your Proof of Entitlement. Options are **Authorized User** or **Value Unit**. If you specify Authorized User, you must also enter the number of licenses purchased.
   - Select **Apply to all new projects created in any workspace** to fully enable the runtime for all new projects, depending on your type of license, created in any workspace.
   - Select **Apply to all existing projects in the current workspace** to fully enable the runtime for all projects, depending on your type of license, that exist in the current workspace. If you have existing projects in other workspaces, you must run the License Settings wizard in each of those workspaces.

**Note:** The license settings of WebFacing projects in a workspace are checked on startup. If any of the projects do not match the master license settings, you are prompted to update them. To disable checking license settings at startup, select **Preferences > WebFacing > HATS/WebFacing** and deselect **Perform license settings check on workbench startup**.
7. Click **Finish**.
Chapter 3. Creating a WebFacing Web project

About this task

The WebFacing Web project is Struts-based and uses Struts 1.3. The Java™ EE level depends on the version of WebSphere Application Server targeted during project creation.

To create a WebFacing Web project:

Procedure

1. Click File > New > Project.
2. Expand WebFacing in the Wizards pane, select WebFacing Web Project, and then click Next to open the WebFacing Web Project wizard.
3. In the WebFacing Web Project wizard:
   a. Enter a name for your project in the Project name field.
   b. In the Project location field, when Use default location is selected, the project is created in the file system location where your workspace resides. To change the default file system location, clear the check box and locate the path using the Browse button.
   c. Select a server in the Target runtime combo box. The server you select determines the project's Java EE version. Note that this combo box is disabled if the EAR project already exists. To ensure that you can select a server, specify a new EAR project in step f below. Click New Runtime to define a new server.
   d. Notice the Dynamic web module version of your WebFacing project. The dynamic web module version adds support for the Java Servlet API and corresponds to the Java EE level. You can modify this setting on the Project Facets page. Access the page with the Modify button beside the Configuration field.
   e. Use the default configuration for the server you selected or select <custom> if you want to configure the project yourself. If you want to modify a project configuration, click Modify to display the Project Facets page. You can add or remove facets from your project using the Project Facets page.
   f. Ensure that the Add project to an EAR check box is selected. A default name for the EAR project is provided if this is the first project you create. If your workspace already contains an EAR project, it will be selected as the default. You can change the EAR project name or keep the default.
   g. Click Next. The next few screens allow you to select display file and UIM source members to convert. If you want to create the WebFacing Web project without converting, click Next until you see the Complete WebFacing Project page and click Finish to create the project.

   Note: If you select Configure advanced settings, the next few screens allow you to specify advanced project settings, such as, the Java build path and Web module settings.

Results

To add WebFacing support to an existing Web project:
1. Right click on the Web project and select **Properties**. This displays the properties dialog for the project.

2. Select the **Project Facets** node.

3. Click **OK** to create an unconverted WebFacing project, or click the **Further configuration available** link to specify the settings for your WebFacing project as described below.

   **Note:** WebFacing Web projects require Java EE 1.4, or later.

4. Click Next. The next screens allow you to select display file and UIM source members to convert.

**Related concepts:**
- Chapter 1, “Introducing the WebFacing Tool,” on page 1

**Related information:**
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

---

### Opening the WebFacing perspective

**Procedure**

1. When the IDE launches it displays the default perspective or the perspective you were last using.

2. To open the WebFacing perspective, select **Window > Open Perspective > Other > WebFacing**. If you have existing projects, you can view them by selecting the **WebFacing Projects** view.

3. Work with an existing project or create a new project with the WebFacing Project wizard. To start the WebFacing Project wizard, select **File > New > Project > WebFacing > WebFacing Web Project** and then click **Next**.

**Results**

**Related concepts:**
- Chapter 1, “Introducing the WebFacing Tool,” on page 1

**Related information:**
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

---

### Choosing a Web style

**About this task**

There are two approaches for customizing WebFacing Web projects:

- Use the Web Site Designer tools.
- Choose from one of the predefined styles that ships with the WebFacing Tool or customize one of the predefined styles.

**Enable for Web Site Designer**

Select **Enable for Web Site Designer** when you want to use the Web Site Designer tools for style customization of your WebFacing project. Web Site Designer will allow you to easily design a web site with a consistent theme and integrate it with WebFacing or other Web applications with the same layout and style. If you choose **Enable for Web Site Designer**, the predefined WebFacing styles will no longer be available for your project.

You can use the Style properties pages to modify the style, but you must use the Web Site Designer tools to maintain and save the style. See the related topic “Editing a Web Site style” on page 49 for more information.
Select classic WebFacing style

You can choose a predefined style or one that you have created using the Style properties pages. By creating your own styles, you can reduce the need to later customize the appearance of your converted application. For example, if there are specific graphics and color schemes that you want to use, by creating a style that incorporates these, you can affect the appearance of multiple pages rather than editing these pages individually after your application has been converted.

Results

Related concepts:
- [Chapter 1, “Introducing the WebFacing Tool,” on page 1](#)

Related information:
- [Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137](#)

Selecting the source members to convert

About this task

Use the Select Source Members To Convert page of the wizard to browse to and select the source members that you want to convert for the current project. You can select DDS and UIM help source members.

Procedure

1. Select your IBM i server. In the Connection field, select the name of the server connection to the IBM i system that has the source members that you will be converting. If no servers are available in the drop-down list, define a new connection using the New button.
2. Choose a library filter. In the Library field, you can choose a predefined filter, enter the name of a library directly, or create your own filter. The default filter is *LIBL. You can use asterisks to create your own filter. For example, if you entered API*, you would see all libraries that begin with the string API.
3. Choose or enter a file filter. In the File field, the default filter is *ALL. You can use asterisks to create your own filter. For example, if you entered *PG*, you would see all files names that have the character sequence PG (for example, RPGAPP).
4. Choose or enter a member filter. In the Member field, the default filter is *ALL. You can use asterisks to create your own filter. For example, if you entered ORD*, you would see all member names that begin with the character sequence ORD (for example, ORDENTR).
5. Choose member types. Use the Member types field to choose whether DSPF, MNUDDS, PNLGRP or all three types are displayed.
6. Click Refresh list to display the members that match the filter values you have chosen. Unless you have previously logged on, you will be prompted to log on to the server you have chosen in the Connection field.
7. Add the members that you want to convert. Browse to specific source members by clicking the + icons of the appropriate library and file. When the members that you want to convert are displayed, select them and then click the arrow to add them to the pane on the right. If you want to select several members at once, hold down the Ctrl key while you are selecting. To add all the members in a file to the conversion list, select the file and click the arrow. To remove a member from the right pane, right-click the member and select Delete from the pop-up menu.
Specifying CL commands for invoking your application

About this task

Use the Specify CL Commands page to indicate the CL commands that are used to launch your application. Each CL command is an entry point into your application and is displayed as a submit button in the welcome page that the WebFacing Tool generates. You can have as many entry points as you want.

For each project, an index.jsp file is generated. The links in the index.jsp file correspond to the text that you enter in the Command label field. After your project has been deployed as a Web application, when a user clicks one of the buttons, the application is launched for browser access using the corresponding CL command.

For each command, there are the following options:
- CL command that is used as an entry point into your application
- Command label that is used for the button text on the invocation page
- Invocation name that is used to uniquely identify the invocation
- Sign on preference

Note: After the project is created you can modify these options by editing the properties for your WebFacing project or by right-clicking the CL command folder of your WebFacing project and selecting Add. This displays the Specify CL Commands dialog.

Using the Specify CL Commands page:

Procedure

1. Enter the command in the CL command field. For example, if you were calling the program ORDENTR in the library APILIB, enter the command CALL APILIB/ORDENTR. If your program requires users to enter interactive parameters, you can supply a CL command in the following format: CALL PROGRAM PARM(&PARMNAME). Using this format, an entry field will be created on the invocation page. The value the user types in this field is passed when the user clicks the invocation button for the application.

2. Enter a label for the command in the Command label field. The text for the label becomes the text for the button users click to invoke your program. If you leave this field blank, the text you entered in the CL command field is used by default.

3. (Optional) Depending on your requirements, enter a unique value in the Invocation name field. In general you can accept the default value that is assigned. The default values are named INV1, INV2, and so on for each command that you enter. One reason to enter a unique name is if you are using programmatic invocation with WebFacing applications. Programmatic invocation allows you to call WebFacing applications from other Web applications using a defined set of arguments. One of the arguments is _inv
which refers to the invocation name of your CL command. If you have an
application that is using programmatic invocation and you require a specific
name for the inv argument, enter that name in the **Invoc**ation name field.

Please refer to “Programmatically invoking WebFacing applications from other
Web applications” on page 89 for further information on programmatic

4. Choose sign on method. Choose **Prompt for signon** or **Specify IBM i signon
values**. If **Prompt for signon** is chosen, users will be prompted for a user ID
and password when they use the application. If **Specify IBM i signon values** is
chosen, the values you enter in the **User ID** and **Password** field are submitted
automatically when the user invokes the program.

**Note:** Ensure that the library that contains the program object referred to by
your command is available in the library list for the user ID that is used to sign
on to the application.

5. Click **Add**. The values you have chosen are added to the command list. If you
want to change the order of the HTML links that launch your program, select
one of the items in the command list and click **Move Up** or **Move Down**. If
you want to delete or modify one of the commands, select the item in the
command list and click **Delete** or **Modify**.

**Results**

**Related concepts:**
- Chapter 1, “Introducing the WebFacing Tool,” on page 1

**Related information:**
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

---

**Finishing the project and converting your DDS source**

**About this task**

Use the Complete WebFacing project page when you have finished the previous
steps of the wizard. You can initiate the conversion at this time, or you can save
the project with the options that you have chosen and convert later. After you
select the conversion option you want to use, click **Finish**.

You will then be returned to the WebFacing perspective. The resources that have
been generated for the project are shown in the WebFacing Projects view. To begin
converting your DDS source, right-click the icon representing the name that you
gave to the project and click **Convert**.

**Related concepts:**
- Chapter 1, “Introducing the WebFacing Tool,” on page 1

**Related information:**
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

---

**Analyzing the conversion logs**

**About this task**

Once the DDS and UIM source has been converted, the conversion log appears in
the right pane. The following logs are produced:
**DSPF conversion log**
This log lists all of the display file members converted, along with the individual record format names in a given member.

**UIM help conversion log**
This log lists all of the UIM help members converted, along with help modules converted for selected UIM help.

General information about the conversion is displayed first on the Overview page. The Overview page displays the following areas:

**General**
This section contains the project name and total number of display file members and UIM help members converted.

**WebFacing Driver**
This section provides information on the WebFacing driver level used for the conversion. This information may be useful when receiving technical support.

**Conversion Information**
This section allows you to produce statistical data regarding the WebFacing conversion.

**Alert Section**
This section provides information on the success of the conversion. Check the individual logs for detailed information on errors and warnings produced during the conversion.

The Referenced Keywords page displays all keywords for each member, along with the level of WebFacing support for each keyword. Select a member in the left pane to display the keywords contained in that member.

The DSPF Conversion Log and UIM Help Conversion Log pages contain detailed information on all display file members and help file members converted. Each page includes a list of all compile and conversion errors and warnings. Select a file in the left pane to show detailed records (or help modules for UIM help) for the selected file. Select a member, record, or help module to display more detailed information on errors or warnings during conversion or at compile time.

**Related concepts:**
Chapter 1, “Introducing the WebFacing Tool,” on page 1

**Related information:**
Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137

---

**Working with WebFacing project resources**

**About this task**
Various resources are created with a sample WebFacing project. You can access these resources by clicking the WebFacing Projects tab from the main screen of the workbench.

A project has folders that correspond to the selections you made in the wizard. For example, there are folders for CL commands, DDS and UIM source, and Style. You can modify the values that you originally chose in the wizard. For example, if there is another CL command that you want to add to the CL commands folder of
your project, right-click the **CL Commands** icon and click **Add**. You will then work directly with the Specifying CL commands screen of the WebFacing Project wizard.

If you want to add a new DDS member to the DDS folder, right click the **DDS** folder and click **Add**. If you want to select a different style, right-click the **Style** folder and click **Select Style**.

If the style name that is displayed is Web Site, you must use the Web Site Designer tools to modify the style. To work with a Web Site style, right-click the **.website-config** icon under the **Navigator** tab and select **Open With > Web Site Designer**.

If you add new DDS or UIM members to a project that has already been converted, you will need to convert the new members. To do this, right-click the new source members and click **Convert**.

Some folders or objects will have properties that you can edit. If an object has specific properties defined, those properties will be displayed in the **Properties** pane beneath the **WebFacing Projects** pane. To edit the properties for your project, right-click the icon for your project and select **Properties**.

To see a file view of your project, click the **Navigator** tab.

### Handling non-WebFaced IBM i applications using dynamic data stream transformation

In a WebFacing project you may have applications where you have non-WebFaced pages. For example, you may not have the DDS source, or you may want to display non-DDS screens (e.g. UIM-based screens) or system screens for which you will not have the source.

By default, all screens that are not converted with WebFacing will be rendered using dynamic data stream transformation. In the case of DDS-based screens, when each display file is first opened by the application the WebFacing project is examined to determine if it has been converted. If all records to be shown on the screen are from display files that have been converted, then the WebFaced screen is shown. Otherwise, the dynamic data stream transformation is performed on the 5250 data stream for that screen and the result displayed.

Dynamic data stream transformation of unconverted DDS is only available when using a V6R1 (or later) host operating system. On earlier versions of the host operating system the dynamic data stream transformation will only present UIM-based application and system screens.

In addition to dynamic data stream transformation contained in WebFacing, unconverted screens can be presented using Host Access Transformation Services (HATS) by combining a HATS project with a HATS/WebFacing project. For more information see “WebFacing interoperability with HATS applications” on page 84 and Chapter 4, “Creating a HATS/WebFacing enabled Web project,” on page 21.

**Note:** UIM help panels that are referenced by DDS screens in your WebFacing application must be converted at the same time you convert the rest of your application screens.

**Related tasks:**

Chapter 4, “Creating a HATS/WebFacing enabled Web project,” on page 21
WebFacing performance considerations within Eclipse workbench

About this task

This section describes steps you can take to improve the performance when developing WebFacing applications within the Eclipse workbench.

1. Close any projects that are not currently in use or being developed to prevent them from being re-built unnecessarily. Only open them if necessary. From the Navigator view, right click the project and select Close Project.

2. Show the heap status and run garbage collection on demand. From the Window menu, select Preferences > General > Show heap status. Click the garbage can icon at the bottom of the workbench to run the garbage collector.

3. To eliminate unnecessary server publishing activity while making changes to an application, set WebSphere Application Server to not publish automatically. Double-click the server instance in the Servers view, expand Publishing, and select Never publish automatically. Note that if you do this, you have to manually click Publish each time you want to publish.

By default, all WebFacing projects have their JSP Compilation and Validation builders turned off. This improves the overall performance of WebFacing projects in the Eclipse workbench. If you write new JSPs for your project or want resources validated, such as XML files, you may want to turn these builders on so that you can see any errors that exist in these files. Steps to turn builders on or off for a WebFacing project are described below.

If you want to further improve the conversion and deployment performance of a WebFacing project, particularly when dealing with large number of DDS members, follow these steps:

Procedure

1. Create your WebFacing project without converting the DDS members in the last page of the project creation wizard.

2. After the project is created, switch to the Navigator view, right-click your newly created project, and select Properties. In the Properties Dialog, click on Builders.

3. Deselect all builders except for Java Builder and WebFacing Builder.

4. To find out what each of the builders are responsible for, please consult the list below. You may choose to turn on some of these builders after the conversion and deployment process is completed so that the project can take advantage of all the capabilities provided by the Eclipse workbench for a Web project.

Note that these changes have no impact on runtime performance of deployed WebFacing applications.

Builders

It is recommended to re-enable builders noted with an asterisk (*) after conversion and deployment.

- Faceted Project Validation Builder *

This builder validates the integrity of all the facets associated with the project.
• Validation
  This task performs validation on various files, including JSP and XML files.
  You can see the list of the validations by selecting Window > Preferences,
  and then clicking on Validation.
Chapter 4. Creating a HATS/WebFacing enabled Web project

About this task

The WebFacing and HATS interoperability feature allows you to easily integrate your WebFacing and HATS Web applications. For more information see “WebFacing interoperability with HATS applications” on page 84.

To take advantage of this feature, create a HATS/WebFacing enabled project and link it with a HATS Web project. You can then run the linked project as a single enterprise application.

The process of creating a HATS/WebFacing enabled project is similar to creating a new WebFacing Web project:

1. Click File > New > Project.
2. Expand WebFacing in the Wizards pane, select HATS/WebFacing Enabled Project, and then click Next to open the HATS/WebFacing Enabled Project Wizard.
3. The remaining steps are identical to the steps for creating a WebFacing Web project with the exception that an EAR project is not specified.

Note: Unlike standalone WebFacing projects, HATS/WebFacing enabled projects are designed to work in conjunction with HATS projects and cannot be run standalone.

Related information:
“WebFacing interoperability with HATS applications” on page 84

Linking a HATS/WebFacing enabled project with a HATS project

Using the Create Linked Project wizard, you can link a HATS/WebFacing enabled project with a HATS project.

By linking a HATS/WebFacing enabled project and HATS project, you can combine applications with converted source and unconverted source. This seamless interoperability allows you to leverage the advantages of both WebFacing and HATS.

Before a screen is displayed in the Web browser, the linked project is scanned for the converted source files. If these files are present, the WebFaced page is shown. Otherwise, the HATS data stream transformation is displayed.

If you do not have a linked HATS/WebFacing project, there is an internal dynamic data stream transformation for WebFacing projects that can be used to handle unconverted source. Please refer to “Handling non-WebFaced IBM i applications using dynamic data stream transformation” on page 17 for more information. To link a HATS/WebFacing enabled project with a HATS project:

1. In the WebFacing Projects view, right-click a HATS/WebFacing enabled project.
2. From the pop-up menu, select Link With -> HATS Project.
The Create Linked Project wizard opens, with your WebFacing project name
specified by default and a list of your HATS projects for you to select. Only HATS
Web projects with a connection type of 5250 or 5250W are displayed in this list.

**Note:** If you select a HATS project with a 5250 connection type, the connection
type for the linked HATS/WebFacing project will still be 5250W.

The Connection Properties page allows you to specify your host name, port, code
page, and screen size. Because you started the wizard by right-clicking the
WebFacing project, the connection settings of the WebFacing project will be
selected by default. Accept the default connection settings and click **Finish** to
create the linked HATS/WebFacing project.

**Tip:** If the connection information in the WebFacing project is different from the
HATS project, you can select a host name and port from values entered for either
project. (If you enter the wizard by right-clicking a HATS project, the HATS project
values will be selected by default as the host name and port.) You can also provide
new values for the host name and port. Code page and Screen size lists contain all
valid options, but are pre-set to the HATS project settings. For more information on
the settings on this wizard page, press **F1** while the page is displayed.
Chapter 5. Customizing your WebFacing applications

You can customize the look and feel of your converted WebFacing applications a number of ways, including the following:

- Use Web Settings to add customization to your DDS members and reconvert the members to see the changed Web interface.
- Use Style properties to change the user interface for the Web-enabled application after conversion, or update the style sheet directly.
- Use Rich Page Editor to change the style, add graphics, update page properties, and create a more customized look.

Related tasks:
- “Editing properties of a WebFacing project”
- “Editing a Web Site style” on page 49
- “Creating and modifying classic WebFacing styles” on page 47

Related reference:
- “Web Settings” on page 51

Editing properties of a WebFacing project

About this task

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. You can edit three types of properties:

- Conversion properties
  - Command Key Recognition Patterns
  - “Key Button Labels” on page 24
  - Edit Code Options
  - MNUDDS Options
  - XML Record Metadata
- Run Time properties
  - Project properties
  - <CL command> properties
  - DDS Object Mappings
  - UIM Object Mappings
  - “Java EE” on page 29
  - Command Key Actions
  - System Screens
- Style properties
  - DDS Field Color
  - DDS Field Display Attributes
  - Window
  - Subfile
  - Command Keys

Related concepts:
Conversion properties

About this task

Use these screens to view and edit conversion 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.

Command Key Recognition Patterns

About this task

Use the Command Key Recognition Patterns to specify how to detect the text for command key buttons to be displayed in a Web browser. When converted, command keys are displayed as buttons on your Web pages. Clicking these buttons is the same as pressing the corresponding command key.

You can define recognition patterns to control how command keys will be displayed. To define a recognition pattern, specify a prefix value in the Prefix field and a separator value in the Separator field, and then click Add. The pattern is added to the list of recognition patterns. When a pattern is recognized during conversion, the value for the text on the Web page button is taken from what is listed after the separator in your DDS source. For example, a recognition pattern with a prefix F and separator = converts the source F4=Prompt to a button with the text Prompt and the source F5=Refresh to a button with the text Refresh. To delete a recognition pattern, select the pattern from the list of recognition patterns and click Delete. You can also use the Move Up and Move Down buttons to alter the placement of recognition patterns in the table so that they appear in the order that you prefer.

Key Button Labels

About this task

Use the Key Button Labels to specify the default text for keys identified during conversion that do not match a defined recognition pattern. For example, if no recognition pattern has been defined that would match F1=Help, you could specify the value for the Key field as F1 and the value for Button label as Help. When converted, keys are displayed as 'Help' buttons on your Web pages. Users click these buttons to launch an event defined in your DDS source such as to refresh their page or to get online help.

Use the Key and Button label fields to specify the default text for keys and then click Add. The key and its label will then be added to the key list. If you want to delete a key label, select it from the key list and select Delete.
There are several ways you can define the label for a function key in your WebFacing applications. If more than one label is defined for a key, WebFacing conversion and run time will use the following priority table to determine which label will appear in the browser. The key label that will be used is the one with higher priority.

### Table 1. Label Priorities

<table>
<thead>
<tr>
<th>Priority</th>
<th>Source of key label text</th>
<th>Scope of setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Command Key Actions</strong> in WebFacing project Run Time properties</td>
<td>Project</td>
</tr>
<tr>
<td>2</td>
<td>Dynamic key labels Web Setting</td>
<td>Field</td>
</tr>
<tr>
<td>3</td>
<td>Key labels Web Setting</td>
<td>Record</td>
</tr>
<tr>
<td>4</td>
<td>Key labels Web Setting</td>
<td>File</td>
</tr>
<tr>
<td>5</td>
<td><strong>Command Key Recognition Patterns</strong> in WebFacing project Conversion properties Note: Function keys labels cannot be obtained using this method. Only command key text constants can be recognized in this way during conversion.</td>
<td>Record</td>
</tr>
<tr>
<td>6</td>
<td>DDS-defined &quot;text&quot; parameter for keyword</td>
<td>Record</td>
</tr>
<tr>
<td>7</td>
<td>DDS-defined &quot;text&quot; parameter for keyword</td>
<td>File</td>
</tr>
<tr>
<td>8 (lowest)</td>
<td><strong>Key Button Labels</strong> in the WebFacing project Conversion properties</td>
<td>Project</td>
</tr>
</tbody>
</table>

### Edit Code Options

**About this task**

Use the Edit Code Options tab to map edit codes. User-defined edit codes are not supported in WebFacing. However, you can map them to one of the available system edit codes.

- Each user-defined edit code can have only one mapping.
- The valid user-defined edit code values are 5-9.
- The valid system edit code values are 1-4, A-D, J-Q, and W-Z.

Select the edit codes to map for conversion. Click **Add**. The mapping appears in the list. Click **OK** when you have made your selection.

### MNUDDS Options

**About this task**

If you have selected DDS members of type MNUDDS for conversion, use the MNUDDS Options to 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. The option of automatically converting commands into hypertext links is selected by default. In the Separator drop-down list, specify a separator to identify the menu options. The default separator is the period (.). The other separators available for selection are the equal sign (=), colon (:), and dash (-).

Click **OK** when you have made your selection.
XML Record Metadata

About this task

The WebFacing conversion generates XML files that describe the data associated with each record format. By default, these files are archived in JAR files. Archiving the XML in JAR files can make deployment more efficient because when your project’s EAR or WAR file is installed on an application server, only a single JAR file is deployed for your DDS data. Some large projects can have thousands of XML files and deploying these as unarchived files increases the time required to install your application.

Depending on your project, though, you may not want the XML stored in a JAR file. Working with unarchived XML files can facilitate change management. A scenario where this might apply would be if more than one developer was working on a WebFacing project. For example, you might convert part of your application’s DDS while another developer converts the rest and the overall project is stored in a repository like CVS. If the generated XML was stored in JAR files, one JAR file would overwrite the other in your repository.

Run Time properties

About this task

Use these screens to view and edit runtime properties. Runtime properties determine the behavior of the converted Web application when it is being used by an end user.

Project Properties

About this task

Values for project properties are stored in the deployment descriptor file web.xml. The following properties appear under the General tab.

Note: Most changes to project properties require the Web application or the application server to be restarted to take effect.

Host name
The host name of the machine where your application is located. This field cannot be blank.

Note: To change the host name used by a linked HATS/WebFacing project, edit the wfhats.xml file in the project EAR.

Host port
The host port that the WebFacing server listens on. By default, port 4004 is used. If you want to use a different port, see the related topic “Changing the port used by your WebFacing applications” on page 34. A Host port is always associated with a specific Host name.

Note: To change the port used by a linked HATS/WebFacing project, edit wfhats.xml in the project EAR.

Handle as if DDS files are compiled with DFRWRT(*YES) option
The DFRWRT keyword controls the writing of records to the display device (for a WebFacing application, the display device is the browser). In 5250 applications, DFRWRT is sometimes set to *NO. This means that all records are immediately written to the display device with no apparent
delay. However, in a converted WebFacing application, the *NO parameter can cause significant delays. Using the property **Handle as if DDS files are compiled with DFRWRT(*YES)**, the default for WebFacing will be to defer writing. That is, WebFacing writes only when the application reads a record or when the application writes a record that has the FRCDTA keyword. You can disable this property in the properties page for a specified CL command or in the properties page for your project. The setting for the CL command overrides the setting for the project.

**Use insert mode for keyboard text entry**

This option controls the insert mode for text entry on input fields in the browser. In a typical Web application, insert mode is on. In a typical 5250 session, insert mode is off (that is, the keyboard is in overwrite mode). By default, in a converted WebFacing application, insert mode is off, so it behaves like the original green-screen application. This allows a user to enter data without selecting or deleting text.

**Include command key name in the key label**

The name of the command key will be included in the label on button used to represent the command key. For example, a return button could be labeled as F3=Return rather than just Return. By default, this option is not enabled.

**Enable Web page compression**

Select this option if you want to enable data compression for your WebFacing project. Data compression can facilitate the transfer of Web pages over your network. Slower networks will benefit the most from compression; for example, if your users are using dial-up connections. Enabling this option will require more system resources to compress the page data. This option is enabled by default.

**Field exit key**

Select this option, then select a key from the key list to enable a field exit key. When your user presses the enabled field exit key, the text after the cursor position in the current input field is deleted and the cursor moves to the next field.

**Override the maximum record wait time (WAITRCRD) specified for DDS files**

Select this option to override the time the browser waits for completion of a read-from-invited-devices operation. Specify the number of seconds or *NOMAX for no limit.

**Error detail level**

Sets the level for error reporting that is output to the end user's browser if there is a runtime error. 1 is the lowest level of error reporting; 3 is the highest.

**Cache Job Date**

Select this option to obtain the system date and time generated by the DATE(*SYS) and TIME keywords from the system the first time one of them is used and are subsequently calculated in the Web application (tier 2) instead of on the IBM i. The job date generated by DATE(*JOB) is cached in tier 2. This is to improve performance. You can use this property to turn off or on caching of the job date that results from DATE(*JOB). Otherwise, WebFacing will cache the job date by default. Note that when this option is set to true, the starting job date is retrieved from the IBM i just once when you first request this value using DATE(*JOB). Also note that general job attributes, such as date format, are only retrieved from the...
IBM i once at startup time before any Web page is displayed. The caching of these other job attributes cannot be turned off.

Show time-out dialog for INVITE
Select this option to enable the time-out dialog when the INVITE operation times out. If you do not specify this option, no timeout dialog will appear when a timeout occurs.

Convert text to uppercase for Katakana (W) keyboard shift
Select this option to convert all characters to uppercase for Japanese Katakana (W) keyboard shift.

The following properties appear under the Authentication tab.

Use single signon
Select Use single signon to enable users to sign on once, using one user ID and password, across multiple platforms and applications. Enterprise Identity Mapping (EIM) and network authentication service (NAS) must be configured to use single signon. For more information, see “Securing your Web applications with single signon” on page 38.

Specify EIM resource reference
Select Specify EIM resource reference to specify the name of the EIM resource reference. This value should match the name of the Resource Reference specified in the Web Deployment Descriptor.

Use IBM i signon
Select Use IBM i signon to use IBM i user profiles to authenticate users. When you select this option, the check boxes below are enabled.

Prompt once for user ID and password
Click this check box if you want to save the settings for user ID and password once the user enters them.

Specify signon values
Use Specify signon values to define a default user ID and password for your WebFacing application. The user ID and password that you set will be submitted automatically by the WebFacing application rather than prompting the user to manually sign on.

User id
The default User ID that will be used to connect to the machine where your original application is located.

Password and Confirm password
The password for the default User ID that will be used to connect to the machine where your original application is located. This password is encrypted and saved in the deployment descriptor file web.xml. The encrypted data is exported with your project when it is deployed to a WebSphere Application Server.

Enable users to change expired passwords
If this option is selected, if a user’s password has expired when they log on to a WebFacing application in the browser, they can change the password directly rather than opening a 5250 session to do so.
<CL command> properties

About this task

For each CL command of your WebFacing project, an entry is created in the invocations.jsp file, which is included in the index.jsp file. A corresponding section in the deployment descriptor file web.xml is also created for each link.

Command label
The text that will be displayed in the HTML link that a user will click to access your WebFacing application.

CL command
The actual CL command. For example, if in a 5250 screen you entered CALL APILIB/ORDENTR (that is, CALL <LIBRARY>/<PROGRAM_OBJECT>) to access your application, the same CL command would be used for your WebFacing application. You can override the project settings for the CL command by changing the properties available with Settings specified for this CL command.

Invocation name
The unique invocation identifier for the CL command.

Settings specified for this CL command
With this property, you can uniquely specify the values for a CL command for Host name, Host port, User ID, and Password. A field is marked with an asterisk if its value is different from its initial value. The initial values in this section are based on your project settings. Values in this section override those set for Project properties if they differ from the project settings. Values are stored in the deployment descriptor file web.xml.

DDS Object mappings

About this task

Values for DDS object mapping are stored in the file DSPFObjectMapping.properties, which is located under the \conf\ path in your Java source directory. Information about object mappings is available in the comments area of the corresponding screen. Mapping information is displayed in the editable text area of the screen.

UIM Object Mappings

About this task

Values for UIM object mapping are stored in the file UIMObjectMapping.properties. Information about UIM object mappings is available in the comments area of the UIM Object Mappings screen, and the actual mapping information is displayed in the editable text area of the screen.

Java EE

About this task

This option displays the Java EE (Java Platform, Enterprise Edition) specification level for your WebFacing application. To change the Java EE level, right-click the project and select Java EE > Specifications Upgrade Wizard.
Command Key Actions

About this task

See “Extending your WebFacing application with command key actions” on page 35 in the Developing WebFacing applications > Customizing your WebFacing applications section of the WebFacing online help.

System Screens

About this task

If you select the check box, all system screens are rendered at run time by dynamic data stream transformation (or by HATS for linked HATS/WebFacing projects), allowing all system screens to share the same look and feel. If you deselect the check box, WebFacing uses built-in Web pages for system screens like DSPSPLF (to display a spool file), which may provide faster response times.

Built in system screens include the following:

- WRKACTJOB
- WRKJOB
- WRKJOBQ
- WRKJRN
- WRKLIB
- WRKMNU
- WRKMSG
- WRKMSGD
- WRKMSGQ
- WRKOBJ
- WRKOBJLCK
- WRKOUTQ
- WRKOUTQD
- WRKPRTSTS
- WRKSBMJOB
- WRKSYSSTS
- WRKUSRJOB
- WRKUSRPRF
- WRKWTR

Style properties

About this task

Use the Style properties screens to customize the appearance of your converted pages. You can customize the look of the application area and the command keys using the Style properties.

For the classic WebFacing styles, 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. For more information on how to edit the layout or frame, see the related topic Creating and modifying styles.
For the Web Site Designer style, if you want to change the look of the layout and frame surrounding these areas, right-click the .websit-config icon and select Open With > Web Site Designer under the Navigator tab of the IDE. Then you can select the Index icon and edit the page template to change the look and layout. For more information on how to use the Web Site Designer tools, see the related topic Using the Web Site Designer style.

Changes to the styles made through the properties pages or by direct editing apply only to the current project.

For Classic WebFacing styles, save them for use in other projects, by right-clicking Style, selecting Style > Save as and giving your style a name. Then this named style becomes available for selection the next time you choose a classic WebFacing style during project creation or the next time you select a style to replace the current project style by right-clicking Style and selecting Style > Select Style. You cannot save a Web Site Designer style like this. You must use the Web Site Designer tools to import template and style files into your projects.

Note: You do not need to reconvert your source files to have any style changes take effect.

**DDS Field Color**

**About this task**

Use the DDS field color screen to indicate how you want DDS field colors to be treated when they are converted for Web use. For example, you can specify that a blue DDS field be displayed as some other color on the Web. You can also control the text and background color for your fields when you use the reverse image display attribute.

To change the mapping for a DDS field color in the Web interface:

1. Under Style in the left pane, select DDS Field Color.
2. In the DDS field color list, select the DDS field color that you want to map.
   The Field color and Reverse image color - DSPATR(RI) group boxes display the colors that will be used in the Web interface, for regular and reverse video text. The table displays the color keyword that is in use; the default row is used when no color keyword is specified.
3. To change the way the selected DDS field color will be mapped, click the push button to the right of one of the Text boxes or the Background color box and select the color you want from the chooser window that opens. Once you have selected a color and closed the chooser, the boxes show you what the text or background in that field will look like on the Web.
4. Click OK when you are done.

**DDS Field Display Attributes**

**About this task**

Use the DDS field display attributes screen to indicate how you want DDS field display attributes to be treated when they are converted for Web use. For example, you can specify that a blinking DDS field be displayed as italic font on the Web.

To change the mapping for a DDS field display attribute in the Web interface:
1. For **Fixed row height** you can specify the fixed row height for your web application. This setting is useful if you use the CLRL keyword or multiple display files using the KEEP and ASSUME keywords in your DDS.

2. For each of the other display attributes there is a description box and a sample box. In each description box you can view the description of the default mapping on the Web for that display attribute, and view the sample on the right.

3. You can change the default by clicking the push button beside the description box. A Font Editor opens. Make your choice and then click OK. The new description and sample choice are displayed in the boxes.

4. In the case of Column separators, you select an image and specify placement options as follows:
   - In the image filename box, you can type in an image filename to use as a background, or click the push button to browse for an image file.
   - In the **Repeat** drop-down list, you can specify to replicate the image horizontally (x-axis), vertically, in both directions or not at all.
   - In the **Position** drop-down list you specify where you want the image.

5. Click OK.

---

**Window**

About this task

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.

1. In the **Window areas** drop-down list, select the area of the window whose appearance you wish to customize.

2. In the **Foreground** group boxes, customize the text for the window area (title, body or shadow) you selected in the list.
   - In the **Color** box, specify the text color. Click the push button beside the box to bring up the Color chooser. Once you have selected a color and closed the chooser, the Color box shows you what the text in that field will look like on the Web.
   - In the **Font** box, you specify the text font. To modify the default font, click the push button beside the box. This launches a font selector. Once you have selected a font and closed the editor, the graphic display to the right shows you what the text in that window area will look like on the Web.

3. In the **Background** group boxes, customize the background for the window area (title, body or shadow) you selected in the list.
   - In the **Color** box, specify the color of the background. Click the push button to bring up the Color chooser. Once you have selected a color and closed the chooser, the Color box shows you what the background for that field will look like on the Web.
   - In the **Image** box, you can type in an image filename to use as a background, or click the push button to browse your image folder.
   - In the **Repeat** drop-down list, you can choose to replicate the image horizontally (x-axis), vertically, in both directions or not at all.
   - In the **Position** drop-down list, you can specify where you want the image.

4. In the **Border** color box, you specify the color of the border.

5. Click OK.
Subfile

About this task

Use the Subfile screen to indicate how the scroll bar and rows that are used to view a Subfile record will look after conversion.

1. In the **Subfile areas** drop-down list, select the area of the subfile display whose appearance you want to customize:
   - **Up arrow**
     The item that the user presses to scroll up.
   - **Slider**
     The item that moves up or down as the user is scrolling.
   - **Scroll bar background**
     The area behind the slider
   - **Down arrow**
     The item that the user presses to scroll down.
   - **Subfile record -- odd row**
     Odd rows in converted Subfile tables
   - **Subfile record -- even row**
     Even rows in converted Subfile tables

2. In the **Background** group, customize the appearance of the area you selected in the list. Modifications are made the same way here as for the **Window** Background group described above. The display area to the right will show what your choices will look like.

3. Click **OK**.

Command Keys

About this task

Use the Command keys screen to specify how the text and buttons for command keys defined in your DDS source will display in a Web browser. When converted, command keys are displayed as buttons on your Web pages. Users click these buttons to launch an event defined in your DDS source such as to refresh their page or to get online help.

1. In the Command key states drop-down list, select the command key state whose appearance you wish to customize:
   - **Default**
     The buttons as they are in their initial state
   - **Rollover**
     The buttons as they display when the user is holding the mouse pointer over them
   - **Button down**
     The buttons as they are being selected by the user

2. In the **Foreground** group, customize the appearance of text in the command key state (Default, Rollover or Button down) that you selected in the list.
   - In the **Color** box, you specify the text color for the command key state you selected. Click the push button beside the box to bring up the Color chooser. Once you have selected a color and closed the Color chooser, the **Color** box shows you what the text in that state will look like on the Web.
   - In the **Font** box, specify the font for the foreground of the window area you selected in the list. To modify the default selection, click the push button.
This launches a Font Editor. Once you have selected a font and closed the editor, the Font box shows you what the text in that state will look like on the Web.

- In the **Text indent in pixels** box, specify an amount to indent the text from the left edge of the command key button.
- In the **Text align** drop-down list, select an option for horizontal alignment of text on the button.
- In **Vertical align** box, select an option for vertical alignment of text on the button.

3. In the **Background** group, customize the appearance of the background for the command key state you selected in the list. Modifications are made the same way here as for the Window Background group described above. Note that you can select a background color as well as an image file that can be used as the background.

4. The three display areas on the right display samples of the choices you made. Click **OK**.

**Note**: If you prefer to edit the Cascading Style Sheet (CSS) files directly, you can use the CSS editor supplied with the workbench.

## Changing the port used by your WebFacing applications

### About this task

The WebFacing server acts as a gateway between WebSphere Application Server and your original IBM i application. The WebFacing server will listen on a predefined TCP port to exchange data with WebFacing applications. All WebFacing applications connecting to the server on the same IBM i need to be configured to connect to the assigned TCP port. By default, the server listens on port 4004. This value can be changed using the IBM i WKSRTVTLE command to modify the TCP service as-WebFacing. If you have changed the port WebFacing server listens to on your IBM i server (see “Changing the port used by the WebFacing server” on page 9), you will need to change the port used by your WebFacing applications so your WebFacing applications will work properly with the new port.

### Changing the WebFacing port number defined in your WebFacing projects

The port number used by a WebFacing application to communicate with the WebFacing server can be changed for the project as a whole or for individual CL commands. Ports defined for CL commands take precedence over the port defined for your project. Each CL command that your application uses will have a separate link for it in the index.jsp file that is created by the WebFacing conversion. If necessary, you can define a different host and associated port for each link. One reason for defining CL commands that use different hosts or ports is if you have a WebFacing project that accesses applications on more than one server. Port changes for your project or CL commands are stored in the deployment descriptor file web.xml.

**Note**: If you are changing a port setting used by a project that has already been deployed, generally, you must re-deploy and then restart the application in the WebSphere Administrative Console. Alternatively, you can search for the IFS location on your IBM i server where your WebFacing Web application’s web.xml file is installed and redeploy from the workbench just that one file. This alternative method still requires a restart of the application. However, it can be a convenient method of changing the ID and password if your application is large.
Changing the port for your project

1. Open the IDE and select the WebFacing Projects tab.
2. Access the properties page for your project. Select and right-click the icon for the WebFacing project whose port you would like to change and then select Properties. The Properties page opens.
3. Open the Run Time > Properties > Project section of the properties page.
4. Click the Change button next to the Host port field.
5. Click Edit in the Host Selection dialog to change the Host port. Click OK when finished. Changes are saved to the web.xml file.

Changing the port for a CL command

1. Open the IDE and select the WebFacing Projects tab.
2. Access the properties page for your project. Select and right-click the icon for the WebFacing project whose port you would like to change and then select Properties. The Properties page opens.
3. Open the Run Time > Properties > Project section of the properties page.
4. Click the label for the CL command that you want to work with.
5. Select the check box for Override project settings with this command.
6. Click the Change button next to the Host port field.
7. Click Edit in the Host Selection dialog to change the Host port. Click OK when finished. Changes are saved to the web.xml file.

Extending your WebFacing application with command key actions

Before you begin

You can define customized command keys in your WebFacing application to invoke a Web service, JSP, or html page. These command keys override the existing function keys and can connect to a Web application URI or an external URL. You might have already defined a JSP or service using a Web diagram, in which case, you can use command key actions to invoke that object or service.

For more information about command keys and related WebFacing tasks, see "Using Web Settings with your DDS source" on page 52

About this task

To define an overriding action for a command key:

Procedure

1. In the WebFacing perspective, right-click your project, select Properties and go to Runtime > Command Key Actions.
2. Click Add.
3. In the Command Key field, specify or select a function key (F1 to F24).
4. In the Action name field, describe the resulting action.
5. Enter the path of the internal or external service in the URI or URL field.
6. Specify or select the Target frame. Selecting *NEW targets a new browser window. Selecting *SAME targets the current window. Specifying an existing window name would target that window.
7. In the Button label field, enter a button label for the service.
8. Check the **Enable only when the command key is active** check box to have your button appear selectively according to the related function key. Otherwise, leave the box unchecked to have your button appear on all screens.

9. Click **OK**.
10. Redeploy your WebFacing application.
11. Restart the server.

**What to do next**

*Note: The WebFacing Tool does not give warnings if a function key is already in use.*

**Labels for command and function keys**

Use the priority table to determine the generated label for a command or function key.

**Purpose**

A command key such as F3 or function key such as HELP is displayed on a WebFacing Web page with a button for submitting F3 or HELP respectively. The text for the label is defined automatically during conversion or run time based on what you have in your DDS, in a Web Setting that you have defined, or in the project properties of your WebFacing application. If more than one label is defined for a key, WebFacing uses the *(Table 2)* table to determine which label will appear in the browser.

*Table 2. Label Priorities*

<table>
<thead>
<tr>
<th>Priority</th>
<th>Source of key label text</th>
<th>Scope of setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Command Key Actions</strong> in WebFacing project Run Time properties</td>
<td>Project</td>
</tr>
<tr>
<td>2</td>
<td><strong>Dynamic key labels</strong> Web Setting</td>
<td>Field</td>
</tr>
<tr>
<td>3</td>
<td><strong>Key labels</strong> Web Setting</td>
<td>Record</td>
</tr>
<tr>
<td>4</td>
<td><strong>Key labels</strong> Web Setting</td>
<td>File</td>
</tr>
<tr>
<td>5</td>
<td><strong>Command Key Recognition Patterns</strong> in WebFacing project Conversion properties Note: Function keys labels cannot be obtained using this method. Only command key text constants can be recognized in this way during conversion.</td>
<td>Record</td>
</tr>
<tr>
<td>6</td>
<td>DDS-defined “text” parameter for keyword</td>
<td>Record</td>
</tr>
<tr>
<td>7</td>
<td>DDS-defined “text” parameter for keyword</td>
<td>File</td>
</tr>
<tr>
<td>8</td>
<td><strong>Key Button Labels</strong> in the WebFacing project Conversion properties</td>
<td>Project</td>
</tr>
</tbody>
</table>

**Related tasks:**

“Editing properties of a WebFacing project” on page 23

**Setting authentication options**

**About this task**

Within the **Authentication** settings in the WebFacing project properties, you can change a number of authentication options for your WebFacing application. You
can prompt users to enter a user ID and password when accessing a WebFacing application, change the signon values originally entered using the **Sign-on with specified values** fields in the Specify CL commands screen, or you can use single signon for your Web application, enabling users to access multiple applications across multiple platforms using a single user ID and password.

**Note:** If you are changing the ID and password for a project that has already been deployed, generally, you must re-deploy and then restart the application in the WebSphere Administrative Console. Alternatively, you can search for the IFS location on your IBM i system where your WebFacing Web application's web.xml file is installed, and redeploy just the web.xml file from the workbench. This alternative method still requires a restart of the application. However, it can be a convenient method of changing the ID and password if your application is large.

**Related concepts:**
“Deployment descriptor” on page 45

### Prompting for user IDs and passwords

**About this task**

If you would like to prompt users to enter an IBM i user ID and password when accessing a WebFacing application, delete the entries for user ID and password in the Properties dialog for your WebFacing project. This will change the values in the deployment descriptor file web.xml in `WebContent > WEB-INF`.

**Procedure**

1. From the **WebFacing Projects** view, right-click your WebFacing project and select **Properties**. The Properties page appears.
2. Open the **Run Time > Project** section of the Properties page.
3. Click the **Authentication** tab to view the settings for authentication.
4. Select the **Use IBM i signon** radio button. This will enable you to change the **Prompt once for user ID and password** and **Specify signon values** boxes.
5. Uncheck the **Specify signon values** box.
6. If you want the user to be prompted once during the browser session, check the **Prompt once for user ID and password** box.
7. Click **OK**. The web.xml file is updated.

### Changing default user IDs and passwords

**About this task**

WebFacing applications can be run under any IBM i profile that has 5250 access. If initially a user ID and password for the application was entered using the **Specify signon values** fields in the Specify CL commands screen these can be changed using the Properties page. This will change the values in the deployment descriptor file web.xml.

**Procedure**

1. From the **WebFacing Projects** view, right-click your WebFacing project and select **Properties**. The Properties page appears.
2. Open the **Run Time > Project** section of the Properties page.
3. Click the **Authentication** tab to view the settings for authentication.
4. Under **Use IBM i signon**, change the values in the **User ID and Password** fields.
5. Click **OK**. The web.xml file is updated.
Setting password protection for individual CL commands

About this task

You can also set up user level password protection for individual CL commands rather than for an entire project. Changes made for CL commands are also stored in the file web.xml.

Procedure

1. From the WebFacing Projects view, expand your WebFacing project and open the CL Commands folder. The folder expands to display the list of CL commands used in your project.
2. Right-click a CL command and select Properties to open the Run Time > Project section of the Properties page.
3. Check the Override project settings for this command box. This will enable you to override the general user ID and password settings for the project.
4. If you want the user to be prompted for their user ID and password, deselect the Specify IBM i signon values check box. Otherwise, enter the default user ID and password to use for signon.
5. Click OK. The web.xml file is updated.

Enabling single signon

About this task

Single signon enables users to access more than one application and multiple platforms using one user ID and password. If you enable single signon, you must also configure Enterprise Identity Mapping (EIM), Lightweight Directory Access Protocol (LDAP), WAS security, and security for your application. EIM is a mechanism for mapping, or associating, a person or entity to the appropriate user identities in various registries throughout the enterprise. To enable single signon for your WebFacing application:

Procedure

1. From the WebFacing Projects view, right-click your WebFacing project and select Properties. The Properties page appears.
2. Open the Run Time > Project section of the Properties page.
3. Click the Authentication tab to view the settings for authentication.
4. Select the Single signon radio button.
5. Click OK. The web.xml file is updated.

What to do next

Note: Selecting single signon disables other authentication options on the Authentication settings page.

Securing your Web applications with single signon

About this task

Single signon enables users to access more than one application and multiple platforms using one user ID and password. For example, you can integrate secured WebFacing applications which are configured using single signon so that a user only needs to be authenticated once. Note that each system involved still requires a separate user ID.
If you want to use single signon for your applications, you need to perform the following tasks:

- Install and configure the Lightweight Directory Access Protocol (LDAP)
- Enable and configure WAS security
- Set up security for your application
- Configure Enterprise Identity Mapping (EIM)
- Configure your applications to use EIM

To perform these tasks, you should install the IBM i Navigator on a client PC. The following tasks use the IBM i Navigator, which is packaged with IBM i Access for Windows, which can be installed from your IBM i server. Ensure that you install all of the networking components, including TCP/IP.

The following describes how each of the main components are used for single signon:

**Lightweight Directory Access Protocol (LDAP)**
EIM configuration is stored in LDAP. WebSphere Application Server can also use LDAP to authenticate Web users. The tasks here assume that WebSphere Application Server is using LDAP for authentication.

**Enterprise Identity Mapping (EIM)**
EIM is required for mapping the ID used for WebSphere Application Server authentication to the profile used to invoke the application on the IBM i server. EIM configuration creates an association between these IDs. The ID used by WebSphere Application Server is the source and the IBM i profile is the target.

**Web application configured for EIM**
Your WebFacing application must be configured to use a token generated by EIM for authentication. This enables users of the application to authenticate to WAS using their LDAP ID and to let EIM map this ID (the source) to an ID on the target IBM i server (the target).

The following diagram illustrates the association between the source and target user identities on two systems. On System A, the user is authenticated by WebSphere Application Server as johnday in order to call an application on System B. On System B, the profile used to run the application on the IBM i is jsd1. The EIM identifier that is used to map the two IDs is John Day. Refer to the following figure while configuring single signon:
Installing and configuring LDAP:

About this task

A Lightweight Directory Access Protocol (LDAP) server is available as part of IBM i in the product Directory Services for IBM i. The server provides a network directory which can be accessed by network clients using the LDAP protocol. LDAP defines the transport and format of messages used by a client to access data in an X.500-like directory. Although LDAP does not define the directory service itself, a directory accessed using LDAP is typically called an LDAP directory.

The directory server allows access to a type of database that stores information in a hierarchical structure similar to the way that the IBM i integrated file system is organized. The LDAP directory server model is based on entries which consist of one or more attributes, such as a name or address, and a type. These attributes typically consist of mnemonic strings, such as cn for common name or mail for e-mail address. Refer to the example below to see how entries are stored in LDAP.

LDAP must be configured on an IBM i system to configure EIM. EIM configuration information is stored in an LDAP directory. These are the basic steps for installing and configuring LDAP. See "Networking > TCP/IP applications, protocols and services > Directory Services (LDAP)" in the IBM i Information Center for more details on working with LDAP. Note that although you can use LDAP on other operating systems, these steps describe configuration for IBM i systems only.

Note: Although a valid LDAP ID will allow you to use protected pages when securing your application, to use single signon an EIM user identity must be associated with that LDAP ID (as the source ID), and a target IBM i profile (as the target ID).
The directory server is installed with IBM i using a default configuration. To re-configure the directory server, you must have *ALLOBJ and *IOSYSCFG authorities. To configure LDAP:

**Procedure**

1. Using the IBM i Navigator, expand **Network > Servers > TCP/IP**.
2. Right-click **Directory Server** and select **Stop**.
3. Right-click **Directory** and select **Configure**. If you have previously configured the directory server, select **Reconfigure**.
4. Use the Configure Directory Server wizard to configure the directory server.
   - Note the settings you use, such as distinguished names (or suffixes), during configuration. Some of these values are required when you enable security for your applications.
   - When the wizard completes, your directory server is set up with a basic configuration. For more detailed information on configuring LDAP, see the "Networking > TCP/IP applications, protocols and services > Directory Services (LDAP)" in the IBM i Information Center.

**What to do next**

After you have configured your directory server, you can add entries to your LDAP directory using one of the following methods:

- Using the IBM i Navigator, right-click **Directory Server** and select **Tools > Import**. This will allow you to import the directory entries from a previously created LDIF (lightweight directory interchange format) file.
- In IBM i, use the ldapadd command from Qshell.
- Use the Directory Management Tool to add entries. The Directory Management Tool is in the IFS at \QIBM\ProdData\OS400\DirSrv\User Tools\Windows\setup.exe.

The following is an example of a simple LDAP file containing two entries:

```

dn: cn=John S. Day, ou=Rochester, o=Big Company, c=US
objectclass: top
objectclass: person
objectclass: organizationalPerson
cn: John Day
sn: Day
uid: johnday
telephonenumber: +1 408 555 1212
description: A big sailing fan.


dn: cn=Bjorn Jensen, ou=Rochester, o=Big Company, c=US
objectclass: top
objectclass: person
objectclass: organizationalPerson
cn: Bjorn Jensen
sn: Jensen
uid: bjenson
telephonenumber: +1 408 555 1212
description:Babs is a big sailing fan, and travels extensively in search of perfect sailing conditions.
title:Product Manager, Rod and Reel Division
```

For more information on adding LDAP entries, see "Networking > TCP/IP applications, protocols and services > Directory Services (LDAP)" in the IBM i Information Center.
Creating a Secured Websphere Application Server (IBM i):

About this task

To create a secured WebSphere Application Server, follow these steps:

Procedure

1. Start the IBM Web Administration for IBM i console. Typically, you can start the Web Administration for IBM i console at the address of http://MYSERVER:2001/HTTPAdmin.
2. Select Create a New Application Server.
3. Advance through the wizard, accepting the default values or setting them as needed until you reach the Configure Identity Token SSO for Web to IBM i Access pane.
4. Select Configure Identity Tokens and fill in the values that match your LDAP configuration.
5. Click Next.
6. Select your source registry from the drop-down.
7. Continue through the wizard to the end.
8. Click Finish to create the server.

Configure the connection factory:

About this task

To configure the connection factory, follow these steps:

Procedure

1. Start server you just created and open its administrative console. Click Launch Administrative Console.
2. Set the connection factory custom properties to match your installation, such as eimDomainName, ldapHostName, parentDomain, and sourceRegistryName.
3. Click Save to save the changes.

Enabling and configuring WAS security:

About this task

Follow these steps to configure WebSphere Application Server security. If you have not already created a WebSphere Application Server instance, you will need to create one before performing these tasks.

Procedure

1. Open the Administrative Console for your server.
2. Expand Security and click Global security.
4. Select the Enable application security check box, and ensure the Use Java 2 security to restrict application access to local resources is not selected. Click Next.
5. The Select user repository page opens. Select the Standalone LDAP registry radio button, and click Next.
6. The Configure user repository page opens:
   a. In the Primary administrative user name field, enter a user ID that will administer the LDAP configuration.
b. Select IBM SecureWay Directory Server in the **Type of LDAP server** field.

c. Type the host name or IP address of the IBM i in the **Host** field.

d. The **Port** field should remain at the default setting, 389.

e. In the **Base distinguished name (DN)** field, enter the domain name of the system in pair-value separated format. For example, if the domain name for your IBM i is torasbcc.torolab.ibm.com, then you would enter this text in the Base distinguished name (DN) field:

   dc=torasbcc,dc=torolab,dc=ibm,dc=com

f. Leave the **Bind distinguished name (DN)** and **Bind password** fields blank, and click **Next**.

7. The Summary page opens. Click **Finish**.

8. Save the changes and close the WAS Administrative Console.

9. Restart the server.

**Setting up security for your application:**

**About this task**

Setting up security for your application requires the following tasks:

- Setting up security roles and constraints for your application by editing the Web Deployment Descriptor.

- Gathering roles used for your application by editing the EAR Deployment Descriptor.

Security roles are logical groups of users (such as bankTeller or bankManager). When you deploy the application, these roles are mapped to specific individuals or groups of individuals. Constraints define which parts of the application to secure, such as servlets and JSPs, and which roles can access them.

To define security roles and constraints:

**Procedure**

1. In the Navigator view, expand the Web project folder and double-click the Web Deployment Descriptor file (web.xml) under WebContent/WEB-INF to open it in the editor. (If the Navigator view is not visible in the workbench, click **Window > Show View > Other > General > Navigator** to display the view.)

2. On the **Design** tab, select the Web Application (Application name) and click the **Add** button in the Overview area. In the popup window, scroll down to select Security Role and click **OK** to insert a new security role. For this application, the role specifies all users in the LDAP directory.

3. In the Security Role Details, enter **All application users** in the **Role Name** field.

4. On the **Design** tab, select the Web Application (Application name) and click the **Add** button in the Overview area. In the popup window, scroll down to select Security Constraint and click **OK** to define the security constraints for this new security role.

5. Type the name for the constraint in the Display Name field, for example, **Access Constraint**.

6. Click **Add** next to Authorization Constraints to add all authorized security roles you just created. Enter a description of the authorization constraint in the Description field. For example, **Allow all**.
7. In the Overview section, select Web Resource Collection under this new "Security Constraint". Type the resource name in the Web Resource Name field (for example, All resources), and click Add next to the URL Pattern box and enter /* for the URL pattern. This specifies that these particular users have access to all Web resources in the application.

8. In the Overview, select Web Resource Collection and click Add button, select the HTTP method in the popup window and click OK to specify the actions that you can perform, for example, GET and POST. If you add nothing, all actions are allowed. The updates to the Security page are complete.


10. In the Navigator view, expand the EAR project folder for your project and double-click the EAR Deployment Descriptor file (application.xml) under the META-INF directory.

11. On the Design tab, select Application and click Add button. Select Security Role and click OK to add all the security roles. Type the security role name details for added security roles.

12. Under WebSphere Deployment Descriptors, click the Open WebSphere Bindings Descriptor link. In the WebSphere bindings section, add all authorized security roles and all authorized users under each security role. Save changes.

13. Save and close the EAR Deployment Descriptor file.

14. Restart the application.

What to do next

With this configuration, you have allowed any user that can be authenticated to LDAP (All authenticated users) to access any resource in the project (/*). In your application, you may want to create additional roles and constraints to further refine which users (roles) can access which resources.

Related concepts:
“Deployment descriptor” on page 45

Configuring your application to use EIM:

Related concepts:
“Deployment descriptor” on page 45

Configuring authentication settings for your WebFacing application:

About this task

If you are using the WebFacing Tool, configure authentication settings in the Run Time properties for your project. See Setting authentication options for more information on setting authentication options for WebFacing applications.

Procedure

1. To open your WebFacing project properties, right-click the project in the WebFacing Projects view, and select Properties. The host name in the project settings is the system containing the target registry.

2. Click the Authentication tab to view the authentication settings.

3. Click the Use single signon radio button.

4. For this example, check the Specify EIM resource reference field and enter idTokenRR.
**Note:** The string `idTokenRR` can be any string. It is used when mapping this resource reference to the actual adapter name configured in WAS. This string is case sensitive.

5. Click **OK** to finish.

**Adding a resource reference to the Web Deployment Descriptor:**

**About this task**

Now update the Web Deployment Descriptor for WebFacing applications.

**Procedure**

1. In the Navigator view, expand the Web project folder and double-click the Web Deployment Descriptor (web.xml) file in the `WebContent\WEB-INF` folder.

2. In the Design tab select the **Web Application** (name of the application) and click the **Add** button in the Overview area. In the popup scroll down to select **Resource Reference** and click **OK** to configure the resource references.

3. In this example, type `idTokenRR` for the **Name** field, select `javax.jms.ConnectionFactory` in the **Type** list, and select Container from the **Authentication** list.

4. Click **Finish**.

5. On the WebSphere Deployment Descriptors Window, click **WebSphere Bindings Descriptor**. On the **Design** tab, select Web Bindings and click the **Add** button. Select Resource Reference in the pop up window and click **OK**. Enter `eis/idTokenRR` for the **Binding name** field. Enter `idTokenRR` in the **Name** field.


**Deployment descriptor**

A deployment descriptor is an XML file that describes how a module or application should be deployed, by specifying configuration and container options. `web.xml` is a deployment descriptor file for Web applications. WebFacing applications use `web.xml` to store information such as the properties you set for your project; for example, the properties you choose for handling user IDs and passwords are stored in `web.xml`.

WebFacing applications can be deployed to the WebSphere Application Server or the IBM i integrated Web application server.

- Use the WebSphere Application Server Network Deployment (ND) if you are deploying WebFacing applications requiring support for high availability and failover.

- Use the WebSphere Application Server Express or Base if you are deploying WebFacing/HATS linked applications or WebFacing applications that use the application bridge. This configuration does not support high availability and failover.

- Use the IBM i integrated Web application server if you are deploying WebFacing applications and have limited IBM i system resources that cannot optimally run the WebSphere Application Server. This configuration does not support high availability and failover.

**Related tasks:**

- "Configuring your application to use EIM" on page 44
- "Editing properties of a WebFacing project" on page 23
- "Setting up security for your application" on page 43
- "Setting authentication options" on page 36

Chapter 5. Customizing your WebFacing applications  45
Working with WebFacing styles

About this task

The visual appearance of your WebFacing application is controlled by a WebFacing style. You can choose one of the many predefined styles provided. You can also customize these styles to meet the unique needs of your application.

Example

The following figure shows a host screen using the predefined finance style. This style provides an example that uses drop-down menus with links to other Web sites.

The figure below shows a host screen using the predefined industry style. This style provides an example that uses menu bar links to other Web sites.
## Creating and modifying classic WebFacing styles

### About this task

To create a new style in WebFacing you must have a project with a supplied WebFacing style, which is a standard, uncustomized style. Modifications to the style are based on the selected style and apply only to the current project. To save the style, right-click your project and select **Style > Save as** and give the style a name. Once you have created a user-defined style, you can save subsequent modifications to the same style name. This style then 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. To replace a style, right-click your project and choose **Style > Select Style**. To see the directory structure for the location of a style within your project, select the **Navigator** tab. In the Navigator view, the path to the styles directory for a project is `<project_name>\WebContent\webfacing\styles`.

If you want to edit the style that is used for the application area and command key elements, edit the `apparea.css` file in the `\apparea` directory using the Style properties page or directly using tools supplied in the IDE. The Style properties pages make it easier to visualize the modifications that are being made and show 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. Note that not all style characteristics can be modified with the WebFacing properties pages. To use the IDE's CSS editor, right-click the file that you want to edit and select **Open With > CSS Designer**.

Properties for styles are stored in two separate directories of the styles folder inside the WebFacing project:

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
<th>Qty</th>
</tr>
</thead>
<tbody>
<tr>
<td>000072</td>
<td>Part number not found</td>
<td></td>
</tr>
<tr>
<td>000057</td>
<td>COBOL_Programmers_Guide</td>
<td></td>
</tr>
</tbody>
</table>

Part number not found.
Files that can be edited for style changes

<table>
<thead>
<tr>
<th>Directory where files are stored</th>
<th>Part of Web page they apply to</th>
<th>Editable using properties pages?</th>
</tr>
</thead>
<tbody>
<tr>
<td>\apparea</td>
<td>• application area</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>• command keys</td>
<td></td>
</tr>
<tr>
<td>\chrome\html and \chrome</td>
<td>• layout</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>• command keys</td>
<td></td>
</tr>
</tbody>
</table>

- apparea.css
- image files (.gif, .jpg, .png)
- PageBuilder.jsp
- CmdKeys.jsp
- User-defined .css files
- <stylename>.css
- <stylename>.js
- MenuKeys.jsp
- image files (.gif, .jpg, .png)

Note: You do not need to reconvert your source files to have any style changes take effect.

Related reference:
“Web Settings” on page 51

Related information:
Chapter 5, “Customizing your WebFacing applications,” on page 23

Using the WebFacing Styles view

About this task

The WebFacing Styles view provides a way to manage your classic WebFacing styles. You can use it to copy, delete, rename, import, and export styles. You can work with user defined styles (styles that you create or copy from existing styles). Or, you can work with IBM defined styles (styles that are shipped with the workbench). You cannot manage a Web Site style using the WebFacing Styles view. You must use the Web Site Designer tools to maintain the style. For information on how to use the Web Site Designer tools, see the related topics.

Note: To change the style for a WebFacing project, in the WebFacing perspective, right-click your project and select **Style > Select Style**. To see the directory structure for where a style is stored for your project, select the **Navigator** tab. In the Navigator view, the path to the styles directory for a project is <project_name>/WebContent/webfacing/styles.

Related topics:
- “Editing a Web Site style” on page 49
- “Web Site Designer style example” on page 51

Accessing the WebFacing Styles view

Procedure

To access the Styles view, select **Window > Show View > WebFacing Styles**. The view is opened in the WebFacing perspective.
Copying styles

About this task

To copy a style, right-click the style and select Copy as. Copy a style when you want to use an existing style as a starting point for creating a new style. You can then modify the new style.

Deleting styles

About this task

To delete a style, right-click the style and select Delete. IBM defined styles that are shipped with the workbench cannot be deleted.

Renaming styles

About this task

To rename a style, right-click the style and select Rename. IBM defined styles that are shipped with the workbench cannot be renamed.

Importing styles

About this task

To import a style from a different directory to the WebFacing styles directory, right-click inside the Styles view and select Import. Ensure that the style you are importing uses the same structure as the IBM defined styles that ship with the workbench. For example, ensure that it contains a chrome directory that includes a .css and PageBuilder.jsp file and that it contains an apparea directory that includes an apparea.css file. Also verify that any images referred to by PageBuilder.jsp or the .css files are in the appropriate directories.

Exporting styles

About this task

To export a style, right-click the style and select Export. You can then browse to the directory that you want to export the style to.

Editing a Web Site style

About this task

The Web Site style can be selected during project creation if you choose to Enable for Web Site Designer in the Choose a Web Style page of the project creation wizard. Once you have selected the Web Site style, a sample template is applied to your WebFacing pages that is consistent with the templates that are available for selection using the Web Site Designer tools. For more information, see the Web Site Designer documentation.

The Web Site style should be used if you have other Web applications that are using Web Site Designer to apply templates and organize the navigation of your Web site. The sample template supplied with the Web Site style does not include any navigation bars or site maps. You should not include these until support for running multiple WebFacing projects in a single browser session is available. The WebFacing project, however, can be included in the navigation of your Web projects and can be modified with the same look and feel as your Web site.

Note: If your application requires a relative URL, the Web Site Designer style may not be suitable.
To see the directory structure in which a style is stored for your project, select the Navigator tab. In the Navigator view, the paths to the styles directories for a project are: `<project_name>\WebContent\webfacing\styles` and `<project_name>\WebContent\theme`.

If you want to edit the style that is used for the application area and command key elements, edit the apparea.css file in the `\apparea` directory using the Style properties pages or directly using tools supplied in the IDE. The Style properties pages make it easier to visualize the modifications that are being made and show 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. 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 surrounding the application area and command keys, edit the user-defined files using Web Site Designer or directly by selecting the files in the file system within the `\theme` and `\webfacing\styles\chrome` directories.

To edit the files in the `\theme` directory using the IDE’s CSS editor, right-click the CSS file that you want to edit and select **Open With > CSS Designer**. To edit the layout and the frame, right-click the WFB_blue.jtpl file and select **Open With > Page Designer for Template**. To edit the PageBuilder.jsp file to which the style and template files are applied, right-click the PageBuilder.jsp file in the `\webfacing\styles\chrome` directory and select **Open With Page Designer**. To edit the template file associated with PageBuilder.jsp from within Page Designer, in the Design view right-click **Open Page Template With > Page Designer for Template**. This is the same as opening the WFB_blue.jtpl file directly using the Page Designer for Template. To edit the CSS style file associated with the template from within Page Designer for Template, in the Styles tab group choose the Styles view and right-click **blue.css > Edit**. This is the same as opening the CSS file directly using the CSS Designer.

To edit the files for the layout and frame using Web Site Designer, activate the WebFacing projects view and expand your project, then expand the **Style > Web Site** folders. Right-click the `.website-config` file, and select **Open With > Web Site Designer**. To edit the template, switch to the Navigator view, then right-click the *jtpl files under the `\theme` directory, and select **Open With > Page Designer for Template**. To edit the CSS style, right-click the *css files under `\theme` and select **Open With > CSS Designer**.

Properties for styles are stored in two separate directories of the styles folder inside the WebFacing project:

<table>
<thead>
<tr>
<th>Files that can be edited for style changes</th>
<th>Directory where files are stored</th>
<th>Part of Web page they apply to</th>
<th>Editable using properties pages?</th>
</tr>
</thead>
<tbody>
<tr>
<td>apparea.css</td>
<td><code>\styles\apparea</code></td>
<td>• application area</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• command keys</td>
<td></td>
</tr>
<tr>
<td>PageBuilder.jsp</td>
<td><code>\chrome</code></td>
<td>• layout</td>
<td>No</td>
</tr>
<tr>
<td>User-defined .css files</td>
<td><code>\chrome\html</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>\theme</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Web Site Designer style example

About this task

This example shows how files within a Web site can have different templates applied to them, as well as a sample template for a navigation bar for a Web site. The file with the navigation template includes links to your WebFacing pages that have a similar look and feel. The example also shows how to link other Web projects to your WebFacing project.

Here are the steps:

1. Create a WebFacing project and make sure you choose to Enable for Web Site Designer in the Choose a Web Style page of the project creation wizard.
2. Select the Navigator tab to see the directory structure for your project. In the Navigator view, right-click on <project_name>/WebContent/index.jsp and select Run As > Run on Server
3. You will see a sample of two navigation links, INV1 and INV2, that will open the first two CL commands that you defined for your project. These are provided as samples and will not work unless you have these CL commands defined with names INV1 and INV2. Click on the links to run your application. Notice that the original links from index.jsp are also available.
4. Select the WebFacing projects tab, expand your project, and select Style > Web Site. Right-click the .website-config file, and select Open With > Web Site Designer. To edit the template, right-click on the Index icon and select Set Template > Open Template With > Page Designer for Template.
5. To add a new navigation link, select the Navigator tab and copy the <project_name>/WebContent/inv2.jsp file and place it in the same directory with the name inv3.jsp. Edit the inv3.jsp file and change any text that refers to INV2 to INV3. Then drag the inv3.jsp file to the Web Site Designer navigation view and place it under the Index icon. Now you have navigation for a third CL command named INV3.

Web Settings

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 Page Designer.

Related concepts:
- “Using Web Settings with your DDS source” on page 52

Related tasks:
- “Editing properties of a WebFacing project” on page 23
- “Editing a Web Site style” on page 49
- “Creating and modifying classic WebFacing styles” on page 47

Related reference:
- “Web Settings considerations” on page 71
Using Web Settings with your DDS source

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 reconver a WebFacing project. Changes made directly to a generated JSP file are lost, however, when you reconver your project.

If you add or change the Web Settings for a project that has already been converted, reconver your project to make the changes part of your WebFacing application. Alternatively, you can reconver just the DDS members that you changed. To reconver specific members, open the WebFacing perspective, click the WebFacing Projects tab, expand the DDS folder, right-click the member that you have changed and select Convert. If the WebFacing Projects view is not open, select Window > Show view > Other > WebFacing > WebFacing Projects.

If more than one Web Setting of the same type is specified on a single field, record, or file, then the behavior is undefined. (In other words, there is no guarantee as to which of those duplicate Web Settings will take effect.)

Table 3. Web Settings

<table>
<thead>
<tr>
<th>DDS object</th>
<th>Icon</th>
<th>Available Web Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>File level</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Display Source Physical File</td>
<td>![Icon]</td>
<td>• Key labels</td>
</tr>
<tr>
<td>Records</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard Record</td>
<td>![Icon]</td>
<td>• Key labels and key order</td>
</tr>
<tr>
<td>Subfile Control Record</td>
<td>![Icon]</td>
<td>• Disabled keys</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Display size</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Insert script</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Application bridge</td>
</tr>
<tr>
<td>Constants</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Message Constant</td>
<td>![Icon]</td>
<td>• Appearance</td>
</tr>
<tr>
<td>Date Constant</td>
<td>![Icon]</td>
<td>• HTML overrides</td>
</tr>
<tr>
<td>Time Constant</td>
<td>![Icon]</td>
<td>• Location</td>
</tr>
<tr>
<td>User Constant</td>
<td>![Icon]</td>
<td></td>
</tr>
<tr>
<td>System Constant</td>
<td>![Icon]</td>
<td></td>
</tr>
</tbody>
</table>

Input-capable fields (including fields with Both specified for usage)
### Table 3. Web Settings (continued)

<table>
<thead>
<tr>
<th>DDS object</th>
<th>Available Web Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alphanumeric Field</td>
<td>• Appearance</td>
</tr>
<tr>
<td></td>
<td>• HTML overrides</td>
</tr>
<tr>
<td></td>
<td>• Location</td>
</tr>
<tr>
<td></td>
<td>• Option labels (only available if input field is specified</td>
</tr>
<tr>
<td></td>
<td>with VALUES keyword)</td>
</tr>
<tr>
<td>Zoned Field</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>Timestamp</td>
<td></td>
</tr>
<tr>
<td>Float</td>
<td></td>
</tr>
<tr>
<td>Continued field</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Appearance</td>
</tr>
<tr>
<td></td>
<td>• Dynamic key labels</td>
</tr>
<tr>
<td></td>
<td>• HTML overrides</td>
</tr>
<tr>
<td></td>
<td>• Hyperlink</td>
</tr>
<tr>
<td></td>
<td>• Image</td>
</tr>
<tr>
<td></td>
<td>• Location</td>
</tr>
<tr>
<td>DBCS Just</td>
<td></td>
</tr>
<tr>
<td>DBCS Either</td>
<td></td>
</tr>
<tr>
<td>DBCS Open</td>
<td></td>
</tr>
<tr>
<td>DBCS Graphic</td>
<td></td>
</tr>
<tr>
<td>Output-only fields</td>
<td></td>
</tr>
<tr>
<td>Alphanumeric Field</td>
<td>• Appearance</td>
</tr>
<tr>
<td></td>
<td>• Dynamic key labels</td>
</tr>
<tr>
<td></td>
<td>• HTML overrides</td>
</tr>
<tr>
<td></td>
<td>• Hyperlink</td>
</tr>
<tr>
<td></td>
<td>• Image</td>
</tr>
<tr>
<td></td>
<td>• Location</td>
</tr>
<tr>
<td>Zoned Field</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>Timestamp</td>
<td></td>
</tr>
<tr>
<td>Float</td>
<td></td>
</tr>
<tr>
<td>DBCS Just</td>
<td></td>
</tr>
<tr>
<td>DBCS Either</td>
<td></td>
</tr>
<tr>
<td>DBCS Open</td>
<td></td>
</tr>
<tr>
<td>DBCS Graphic</td>
<td></td>
</tr>
<tr>
<td>Hidden fields</td>
<td></td>
</tr>
<tr>
<td>Text constant</td>
<td></td>
</tr>
</tbody>
</table>
Table 3. Web Settings (continued)

<table>
<thead>
<tr>
<th>DDS object</th>
<th>Icon</th>
<th>Available Web Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alphanumeric Field</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zoned Field</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timestamp</td>
<td></td>
<td>* Send to browser</td>
</tr>
<tr>
<td>Float</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBCS Just</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBCS Either</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBCS Open</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBCS Graphic</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Editing Web Settings for a DDS member

About this task

When you first open the WebFacing perspective, the Web Settings view is displayed by default. The view will remain empty until you open a DSPF DDS source member in the LPEX Editor, or alternatively in the Screen Designer. To edit Web Settings for a DDS member:

Procedure

1. In the WebFacing perspective, expand **WebFacing project > DDS**.
2. Right-click the DDS member you want to work with and select **Open With > Remote Systems LPEX Editor**.
3. If prompted, sign on to the server. The file opens in the editor window.
4. Click on a DDS object in the editor. The list of settings available for that object is displayed in the Web Settings view.
5. Click on a setting in the list to edit it. The options for the Web Setting are displayed on the right side of the Web Settings view.
6. If the Web Settings view is not open, open it by selecting **Window > Show View > Other > WebFacing > Web Settings**.
7. To quickly navigate to specific objects, select **Window > Show View > Outline** to open the Outline view. As you click on objects in the Outline view, the focus will switch to that object in the editor window.

Web Settings descriptions

The Web Settings available for each DDS object vary depending on the object that you are working with. This section provides 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. The following is the syntax convention for Web Settings:

*%%WB <number> <level>
where <number> is the number of the Web Setting and <level> is its level
({FIL|REC|FLD}). You can span Web Settings over multiple lines by putting + as the
last non-blank character on the line and starting the next line with *%%WB +.

The following sections describe the Web Settings by file, record, and field level,
and include the syntax of the comments added to your DDS source for each
setting. Some Web Settings use a different syntax prior to WebSphere Development
Studio Client for iSeries® Version 6.0. Although the syntax for some of the Web
Settings is obsolete, it is converted to the updated syntax during WebFacing
conversion. Deprecated syntax is shown if applicable for each Web Setting.

File-level Web Settings

The only file-level Web Setting is Key labels.

Key labels

For function or command keys in your DDS source, you can specify the
text labels to be used in your Web pages to represent the keys. To change a
label, select a function or command key from the Key labels list,
double-click the New Label field, and enter the new text for the key. To
remove a label that you have defined, double-click the New Label field
and delete the text. You can also select the appropriate row and use the
Edit Label or Delete key to perform these functions.

Note: Function key labels cannot be overridden.

Syntax:

A*%%WB 9 FIL <key_label>[@<key_label2>[@<key_label3>[@...]]]

Example:

A*%%WB 9 FIL 12=Cancel|03=Exit

Record-level Web Settings

The following are the Web Setting categories available at the record level:

- Display size
- Insert script
- Key labels
- Key order
- Disabled Keys
- Application bridge

Display size

The Display size setting is available if a secondary display size has been
selected for your display file. Secondary display sizes can be specified
using the file-level DSPSIZ keyword. Use the Display size Web Setting to
change the size of the HTML table that will be used to display your
application in a browser. This Web Setting only applies to records that
have a conditioned DSPMOD keyword. These records could be displayed
in either the primary or secondary display size depending on the value of
the condition at run time, however, WebFacing conversion only targets one
display size. If you want the application to show this record in the
secondary display size rather than the default primary display size, you
need to specify this Web Setting.
Syntax:
A*%%WB 10 REC

Example:
A*%%WB 10 REC

Insert script

Use the Insert script setting to customize the JSP files that are created for regular records or subfile control records. For example, if you have a subfile control record LISTCTL, the WebFacing conversion will create a corresponding JSP file LISTCTLJavaScript.jsp. The code that you add in the text area of the Insert script setting will be added to the file LISTCTLJavaScript.jsp when it is generated. This setting allows you to enter arbitrary JavaScript or JSP scriptlets into your converted JSP files. This can be used to initialize fields or perform any work required for the specific record format. For example, you could add code that sets a field’s value if its initial value is null:

```java
if(&{CID.INITIALVALUE} == "")
{
 &{CID.REF}.value = "0001";
}
```

In this example, the code &{CID.INITIALVALUE} refers to a field CID that was selected using the Insert field attribute button. For more information on using field replacement variables, see "Specifying field replacement variables" on page 69. Other examples would be JSP code that determines the IP address of a WebFacing user or JavaScript that displays the time of day of the user’s system. You can use Insert script along with the Send to browser setting. For example, you can use values determined by your scripts to populate hidden fields that you have set with the Send to browser setting. If there is an existing field in your DDS that you want to use as part of your script, add it to your script using the Insert field attribute button. For example, you can select an H or hidden field and use it in your script so that a value can be assigned to it.

Syntax:
A*%%WB 18 REC 1 <script>

deprecated:
A*%%WB 18 REC <script>

For Insert script, there are two deprecated forms of replacement variables:

&$FLREC{FIELD} which becomes &{SFLREC.FIELD.REF}

and

&{FIELD} which becomes &{FIELD.REF}

For example, the old Insert script (REC):

A*%%WB 18 REC &{FIELD1}.value=1;&RECORD3{FIELD1}.value=4

becomes the new Insert script (REC 1):

A*%%WB 18 REC 1 &{FIELD1.REF}.value=1;&{RECORD3.FIELD1.REF}.value=4

Example:
A*%%WB 18 REC 1 &{CUSTCTL.field1.CURRENTVALUE}
Key labels

For function or command keys in your DDS source, you can specify the text labels to be used in your Web pages to represent the keys. To change a label, select a key from the Key labels and key order list, double-click the New Label field, and enter the text you would prefer for the key. The New Label value defaults to the File level key labels Web Setting if one is defined. To remove a label that you have defined, double-click the New Label field and delete the text. You can also select the appropriate row and use the Edit Label or Delete key to perform these functions. The Default button removes all related Web Settings (like Key labels, Key order, and Disabled keys).

Note: There are several ways you can define the label for a function key in your WebFacing applications. If more than one label is defined for a key, WebFacing conversion and runtime uses a certain priority to determine which label will appear in the browser. Refer to “Key Button Labels” on page 24 for the priority used for determining key labels.

Syntax:
A*%%WB 9 REC <key_label>["|"<key_label2>["|"<key_label3>[...]]]

Example:
A*%%WB 9 REC 2=enter|1=print

Key order

For function or command keys in your DDS source, you can specify whether or not a key label is hidden on your Web pages and the order in which the keys are displayed.

To hide a key, select the key from the Key labels and key order list and select Hidden from the drop down list in the State column. This removes the key from the list of buttons on the Web page but does not disable the corresponding key on the keyboard.

To disable a key, select the key from the Key labels and key order list and select Disabled from the drop down list in the State column. This removes the key from the list of buttons on the Web page and disables the corresponding key on the keyboard.

If more than one key is available in the Key labels and key order list, the order in which it is displayed can be changed by selecting the key and clicking the Move Up or Move Down buttons. The Default button restores the keys to their original order.

Syntax:
A*%%WB 11 REC <aid_key> [<aid_key2> [<aid_key3> [...]]]

where <aid_key> = CAnn | CFnn | SFLFOLD[(CAIn|CFnn)] | SFLDROP[(CAIn|CFnn)].

Example:
A*%%WB 11 REC CA02 CF14 SFLFOLD(CF10) SFLDROP(CA09)

If SFLFOLD or SFLDROP is specified with no aid key, for example:
A*%%WB 11 REC CA02 CF14 SFLFOLD SFLDROP(CA09)

the Web Setting will be updated to the following:
with the SFLFOLD keyword specified using the command function key 10.

**Disabled keys**

This Web Setting works in conjunction with key labels and key order. It allows you to disable an active aid key (command or function key) on the keyboard as opposed to just hiding its label from the Web user interface.

On the user interface for Web Settings you can set the state of a key to one of three values:

- Visible: Key label is visible and the related keyboard key is active.
- Hidden: Key label is hidden but the related keyboard key is still active.
- Disabled: Key label is hidden and the related keyboard key is disabled.

Keys are sequenced according to the Key order Web Setting. If a Key order Web Setting is not specified, the sequence is determined according to the numeric order for command keys and alphabetic order for function keys, with command keys always coming before function keys. If a key is defined in both the visible Key order Web Setting (A*%%WB 11) and Disabled key Web Setting (A*%%WB 41), the visible sequence takes precedence.

**Syntax**

```
A*%%WB 41 REC <aid_key> [/aid_key2 [aid_key3 [aid_key4]]]
```

**Examples**

Defined keys: CF03 CA01 PRINT HELP

1. Visible key order: CA01 HELP
   - Disabled: CA01 CF03
     - Results: (ordered visible keys: CA01 HELP; hidden but enabled keys: PRINT; disabled keys: CF03)

2. Visible key order: none
   - Disabled: CA01 CF03
     - Results: (ordered visible keys: HELP PRINT; hidden but enabled keys: none; disabled keys: CA01 CF03)

3. Visible key order: CA01 HELP
   - Disabled: none
     - Results: (ordered visible keys: CA01 HELP; hidden but enabled keys: CF03 PRINT; disabled keys: none)

4. Visible key order: none
   - Disabled: none
     - Results: (ordered visible keys: CA01 CF03 HELP PRINT; hidden but enabled keys: none; disabled keys: none)

**Application bridge**

To interoperate with another Web application, choose the Application bridge setting and select Use this record for Application bridge parameters. Note that this setting is only available for records which only have hidden fields (usage H), or which have no fields.

In the entry field, enter the URL of the target application. This URL follows the format "/*appContextRoot/appEntryPoint" where appContextRoot is the context root for the Web application you want to interoperate with and appEntryPoint is the entry point for that Web application. Refer to "Using
the WebFacing application bridge to interoperate with other Web
applications” on page 86 for details about how to interoperate with another
Web application.

Note:
• If you specify the Application bridge setting, all other settings on this
record and on all the fields in this record will not be available and will
be deleted if specified. If this is not desired, use Undo or Ctrl-Z to return
to the original settings.
• Field names in the record should not contain characters other than 'A'-'Z'
and '0'-'9'.
• The DATFMT and DATSEP keywords should not be used on a date
hidden field, and the TIMFMT and TIMSEP keywords should not be on
a time hidden field.

Syntax

A*%%WB 42 REC /appContextRoot/appEntryPoint

Example

To interoperate with a Web application where the context root is webApp
and the entry point is ProcessOrder:

A*%%WB 42 REC /webApp/ProcessOrder

Field-level Web Settings

The following are the Web Setting categories available at the field level:
• Appearance
• Dynamic key labels
• HTML overrides
• Hyperlink
• Image
• Location
• Option labels
• Send to browser

Appearance

Options available for the Appearance and Text category are Visibility,
Override style class, and Override field value:
• Visibility allows you to make the field hidden by using the Hidden
Web Setting. You can also use the Hide characters Web Setting to
indicate a range of characters in your field that you do not want to
display. For example, if your screen displays a menu of options for a
user to choose from and each option is listed numerically (such as: 1.
User tasks - 2. Office tasks), you might want to hide the first two
characters so that only the text for the option is displayed. In an
example like this, you would choose Hide characters and specify a
range of From 1 To 2. Note that if you use the Hidden Web Setting, no
other Web Setting is allowed on the same field.
• Override style class allows you to specify a style sheet class to be used
with the field. As an example, you may want to display all numeric data
with a particular font or color and you have defined a style sheet class
numeric for this purpose. If you had a customer number field in your
program such as custno, you could select the **Override style class** check box and enter the name for your numeric style sheet class.

- **Override field value** allows you to specify text that you want to display in place of the returned value for your field. For a constant field, the title of this check box changes to **Override text** and the field defaults to the text value of the constant.

**Syntax (Hidden):**
```
A***WB 4 FLD
```

**Syntax (Hide characters):**
```
A***WB 5 FLD <from> <to>
```

**Syntax (Override style class):**
```
A***WB 1 FLD <style> <field>
```

**Syntax (Override field value):**
```
A***WB 3 FLD <length>|<text>
```

**Example (Hidden):**
```
A***WB 4 FLD
```

**Example (Hide characters):**
```
A***WB 5 FLD 1 12
```

**Example (Override style class):**
```
A***WB 1 FLD wf_default wf_field
```

**Example (Override field value):**
```
A***WB 3 FLD 10|0000000000
```

**Dynamic key labels**

The **Dynamic key labels** setting can be used to determine the text label of a command key at runtime. This setting is only available for output fields. With this setting, a command key is associated with the output field and text for the command key is determined by the runtime value of the field. An example use is if your application is to support more than one language and you want the command key text to appear in the user's native language. One way of doing this is to have a MSGF per language that contains the text strings that your application uses and put each MSGF in a language specific library. If the output field’s value is read from the MSGF, you could change the language of the text displayed in the field by putting the library that contains the MSFG for your user’s language higher in the library list in the user’s profile. Subsequently, the command key label would be displayed in their language.

To use this setting, choose the **Dynamic key labels** setting and select Use the field value as the label for the function key. In the list, choose the key that will be associated with the text from the output field. *AUTO can also be chosen from the list. *AUTO means that the field value is parsed at runtime to determine which command key to assign the label to. This allows multiple labels to be parsed.

**Note:** There are several ways you can define the label for a function key in your WebFacing applications. If more than one label is defined for a key,
WebFacing conversion and runtime uses a certain priority to determine which label will appear in the browser. Refer to "Key Button Labels" on page 24 for the priority used for determining key labels.

**Note:** Only a limited set of string patterns are currently supported by *AUTO. Assess whether *AUTO will work with your application based on the patterns listed in the prefix and separator table below and your knowledge of the strings for command key labels that are used in your application. The table identifies patterns by a prefix and a separator. For example, if the string F3=Exit is the output value and *AUTO is the value chosen in the list, the string Exit will be used for the command key label. That is, F is treated as the prefix, = is treated as the separator, and the text following the separator is used for the label. If you specify *AUTO, you can define multiple labels in the output field.

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Separator</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>=</td>
</tr>
<tr>
<td>FP</td>
<td>=</td>
</tr>
<tr>
<td>CF</td>
<td>=</td>
</tr>
<tr>
<td>CA</td>
<td>=</td>
</tr>
<tr>
<td>PF</td>
<td>=</td>
</tr>
</tbody>
</table>

**Warnings**

- *AUTO can affect runtime performance. Use of *AUTO can be avoided by ensuring that there is a one-to-one mapping of output fields to command keys.
- If you are not using *AUTO and you are using MSGFs for the text of output fields, the MSGF text should not include the key. For example, for Exit, the text in a MSGF should be Exit not F3=Exit.

**Syntax:**

```
A*%%WB 20 FLD (<key_number> | *AUTO)
```

**Example:**

```
A*%%WB 20 FLD F12
```

**HTML overrides**

**HTML overrides** allows you to override the generated HTML by replacing it with user-defined HTML or inserting HTML along with the generated HTML.

Click the **Program-defined** radio button to specify that the value of the field itself should be used as the generated HTML. Use this setting if you want the application program to compute the HTML for this field rather than using something generated by WebFacing conversion.

Click the **User-defined** radio button to specify HTML text or tags that can be displayed in place of or with the value for your field. In the entry area, enter the HTML tags you would like to use. The default value for this Web Setting is the script representing the value of the field or the text value of the constant. If you want to use the value for your field as part of the HTML, click the button **Insert field attribute.** For more information on using field replacement variables, see "Specifying field replacement variables" on page 69. One example of how you could use this option would be if you wanted the value for your field to be displayed with a
To do this, use the <B> tag with the &{FIELD.INITIALVALUE} text. With this example, edit the entry area so that it displays <B>&{FIELD.INITIALVALUE}</B>. If the code for &{FIELD.INITIALVALUE} is not there by default, it can be added by clicking the Insert field attribute button. You can also add special values, such as the system date or time, to your user-defined HTML. See “Using special values” on page 70 for more information on using special values in your HTML.

Click the Insert radio button to customize your fields by including additional HTML. HTML can optionally be added Before, Inside, and After the HTML tag that is automatically generated for the field. As an example, you might have an input field for employee numbers and at your company all employee numbers are prefixed with 64-. You could make the field easier to use by adding some HTML such as:

**Before (%)**

```
<strong>64- prefix mandatory</strong>
```

**Inside (%)**

```
value="64-" title="Employee number"
```

**After (%)**

```
<a href="help.htm">Click here for more information on this field</a>
```

In this example, the HTML Before the field provides some information about the field and uses the <strong> tag so that the information is emphasized with a bold font. Since the HTML generated by conversion uses the <input> tag for input fields, Inside the <input> tag we could add the HTML attribute value="64-" so that the field will be pre-populated with the employee number prefix and your users will not be required to enter it. We also add the attribute title="Employee number" so that an informational tooltip appears when the mouse is hovered over the field. After the field is the HTML for a link that users can click to get more information about the field.

Ensure that you thoroughly test pages that you are altering with the Insert HTML setting. Changes that you make can conflict with the HTML that is generated by the WebFacing conversion. One way to work around this would be to convert your program without using Web Settings, look at the .html or .jsp source that is generated to see what changes can safely be made with Web Settings, apply Web Settings and then reconvert.

**Syntax (Program-defined):**

```
A*%%WB 16 FLD
```

**Syntax (User-defined):**

```
A*%%WB 17 FLD 1 <html>
A*%%WB 17 FLD 1 &{FIELD.INITIALVALUE}
```

If encountered, the deprecated %FieldValue is mapped to the newer form of &{FIELD.INITIALVALUE}:

```
A*%%WB 17 FLD <html>
A*%%WB 17 FLD &FieldValue
```

**Syntax (Insert before):**
**Hyperlink**

Use the **Hyperlink** setting to display your field as a hyperlink. You can format your hyperlink according to the DDS settings or browser settings, and link to a static page or perform actions. If you want to override the browser's settings for hyperlink appearance, select the **Use DDS settings** check box. If you use the **Hyperlink** setting along with the **Image** setting, your field can be displayed as a graphic that also acts as a hyperlink. Some of these settings accept field values as input. For more information on using field replacement variables, see "Specifying field replacement variables" on page 69.

**Note:** There are some conditions where Action hyperlinks that you create with Web Settings will not show up in a browser. This is done to prevent error conditions when the hyperlink does not apply to what is active or available on the browser. For example:

- The hyperlink is in a record that is not currently the top window or on the top layer.
- A function key that you have associated with the link is not currently active.
- A referenced field that is part of the link is not in an active record or is not readable. For example, the record is protected by other records which use the PROTECT keyword.

**Use DDS settings**

Most browsers follow a convention for the color and appearance of hyperlinks on Web pages or have settings for hyperlink appearance that users can change. For example, unvisited links may be displayed in blue text and visited links may be displayed in purple text. If you want the browser to display the hyperlink with the same appearance as is used for the field in your DDS, select the **Use DDS settings** radio button.
Go to URL

If there is a specific URL that you want to be displayed for a field, select Go to URL. The output value of your field will be the text that is displayed as a hyperlink. The location loaded by the browser when a user clicks the link is the value that is entered in the URL field. If you want the link to launch a separate browser window when it is clicked, enter a name for the window in the Target field. If you leave the Target field blank or enter *SAME, the link will open in the current browser window.

If there is a field value in your DDS that you would like to be part of the URL, click the button on the URL field and select the record, field, and attribute you want to use. For example, if you wanted to add parameters to the URL http://mysite.com/mywebapp/servlet/MyServlet and the servlet was expecting arguments for variables called parm1 and parm2, you could use Insert field value to help construct the URL. In this example, your final URL might look like http://mysite.com/mywebapp/servlet/MyServlet?parm1=\&{FIELD1.INITIALVALUE}&parm2=\&{FIELD2.INITIALVALUE} where &{FIELD1.INITIALVALUE} and &{FIELD2.INITIALVALUE} are parameters you added and that represent values that change at runtime.

Note: If you would like to use a field value from a different record format to be part of the URL for your hyperlink field, you must ensure that the field will be available on the Web page when the hyperlink is displayed. Fields that are in different record formats can be identified in the Insert field attribute dialog because they include the record format name, for example, &{RECORD2.FIELD1.INITIALVALUE}. If both record formats are not available on the Web page when the hyperlink is displayed, an exception will occur when the hyperlink is selected. For more information on using field replacement variables, see “Specifying field replacement variables” on page 69.

Syntax:
A+BWB 12 FLD 3 <url>"|"<target>

deprecated:
A+BWB 12 FLD 1 <url>"|"<target>

Example:
A+BWB 12 FLD 3 http://server/somepath?p2=&{FIELD2.INITIALVALUE}&p1=&{FIELD1.INITIALVALUE}&p3=&{FIELD1}

deprecated:
A+BWB 12 FLD 1 http://server/somepath?p2=&{FIELD2}&p1=&{FIELD1}&p3=&{FIELD1}
Perform actions

The **Perform actions** setting creates a link that when clicked can position the cursor to a field, perform a submit, or a combination of the two. The Position cursor to field and More actions functions both have values that you can specify.

Position cursor to field

When the link is clicked, the runtime value for cursor location that is returned to your application will be the location of the field you specify here. You can select a field from the drop down list or manually enter the name of a field. If you do not select Position cursor to field, the cursor is positioned to the field that was last selected by the Web user. This cursor value does not refer to the cursor as it is displayed by the browser, it refers to the WebFacing runtime's value for current cursor location of fields in the DDS record. If you want to assign a value to the field where the cursor is positioned, choose a field or select the Set field value to check box and enter a value in the text area. Some of these settings accept field values as input. For more information on using field replacement variables, see “Specifying field replacement variables” on page 69.

Syntax:

```
A*%%WB 12 FLD 5 fld=<field>"|"value=<value>
```

More actions

If the More actions check box is selected, you can choose a Function key or JavaScript function call submit action for the hyperlink. For example, from the Submit function key drop down list, you can choose ENTER, LOGOFF, or a function key that you have defined in your DDS. If there is a function key from another DDS record that you would like to use, manually enter the value for the function key.

When a user clicks a link that you have created for a function key, the program will execute that function. In the JavaScript function call field, you can enter the name of a JavaScript function that you want to be executed when the link is clicked. The function can be one that you have entered using the Call JavaScript Web Setting, a function that you have defined in an external JavaScript file, or a function that is included with WebFacing in the file webface.js. If you create your own external JavaScript files, put them in the usr directory of your WebFacing project. This directory is located in the path `<project name>\WebContent\ClientScript\usr`. Any scripts that you put in this directory are available to your WebFacing application at runtime.

You can combine the Position cursor to field option with the Submit function key option. Do this if you want to set the value for a specific field and submit that value back to your program. To do this, select the Position cursor to field check box, choose a field from the drop down list (or enter a field), select the More actions checkbox, select the Submit function key radio button, and choose ENTER from the drop down list. For information about the entry format...
to use for **Position cursor to field** and the **Submit function key** field, see the section of this document **Position cursor to the field**.

**Syntax:**

```
A*%%WB 12 FLD 4 fld=<field>"|value=<value>"|func=<function_key>
```

deprecated:

```
A*%%WB 12 FLD 2 fld=<field>"|value=<value>
```

**Example:**

```
A*%%WB 12 FLD 4 fld=&{FIELD3.REF}|value=1|func=ENTER
```

deprecated:

```
A*%%WB 12 FLD 2 fld=FIELD3|value=1
```

**Call JavaScript**

Enter the name of a JavaScript function that you want to be called when the link is clicked. For example, enter myFunction(). The function can be one that you have entered using the **Insert script** Web Setting, a function that you have defined in an external JavaScript file, or a function that is included with WebFacing in the file webface.js. If you create your own external JavaScript files, put them in the user directory of your WebFacing project. This directory is located in the path `<project name>\WebContent\ClientScript\usr`. Any scripts that you put in this directory are available to your WebFacing application at runtime. If you have a short inline JavaScript function, you can also enter the entire function in the **Call JavaScript** field.

**Syntax (Submit function key):**

```
A*%%WB 12 FLD 6 <function_key>
```

**Syntax (Call JavaScript):**

```
A*%%WB 12 FLD 7 <javascript>
```

**Syntax (Position cursor to a specific field with data and submit JavaScript):**

```
A*%%WB 12 FLD 8 fld=<field>"|value=<value>"|func=<javascript>
```

**Syntax (JavaScript hyperlink):**

```
A*%%WB 12 FLD 9 <javascript>
```

**Note:** JavaScript hyperlink is deprecated, and becomes Call JavaScript.

**Image**

Use the **Image** setting if you want to display an image in place of your field. An example use for this setting would be a product database that includes images of inventoried items.

Populate the **File name** field with the complete path to an image file on your workstation or with the complete URL to an image file (e.g. http://www.ibm.com/image.gif). The **File name** field can be populated by
entering a value directly or by using the Browse button to locate a local file. Local images that you specify will be copied to the ..\WebContent\images\generated\ directory of your WebFacing project during the WebFacing conversion. The images in this directory are later published to the server when you deploy your project with the Export wizard.

**Width** and **Height** can be used if you know the dimensions of an image. Adding width and height information can sometimes decrease the time required to render Web pages that contain images.

The **Alternative text** field allows you to specify the alternative text for the image.

If there are fields in your program that return values that can be used as part of the image file name, or the full URL to an image file, these values can be added to the **File name** field by clicking **Insert field attribute** and selecting a field from the **Field**. For example, if you had &{FIELD.INITIALVALUE} that returned the value bicycle1, you could easily refer to a graphic bicycle1.gif. In this example, first select the **Dynamic** radio button and then populate the **File name** field using **Insert field attribute** and then add the .gif extension. Files that you identify by using a field attribute must be manually copied over to the ..\WebContent\images\generated\ directory.

**Note:** The only field attribute allowed for this Web Setting is **Initial Value**. This is because images must be fully resolved at the time a Web page is first loaded.

For more information on using field replacement variables, see “Specifying field replacement variables” on page 69.

**Syntax:**
A*%%WB 13 FLD 1 <width>*"|"<height>*"|"<graphic>*"|"<alternative text>

**Example:**
A*%%WB 13 FLD 1 15|34|graphic.gif
A*%%WB 13 FLD 1 15|34|graphic.gif|IBM logo
A*%%WB 13 FLD 1 3|2|&{FIELD1.INITIALVALUE}.gif

deprecated:
A*%%WB 13 FLD <width>*"|"<height>*"|"<graphic>

**Location**
Options available for the Location setting are **Change span** and **Change position**. After your DDS source has been converted with the WebFacing Tool, your application is positioned on a Web page using an HTML table. The HTML table has the same number of rows and columns as your DDS screen -- that is, 24 by 80 or 27 by 132. With Location settings, you can change how your fields will be positioned on the HTML table after your source is converted.

- **Change position** allows you to indicate the row and column in the HTML table that will be the starting position for your field.
- **Change span** allows you to indicate the number of rows and columns in the HTML table that will be given to your field. A potential use for **Change span** would be using a graphic in place of the text values that your program originally used to populate the field. For example, the original program may have required 7 columns for text for the field but the graphic that you want to use only requires 2 columns.

The editor window shows the row and column layout for your DDS source. Use this view to get an idea of how your fields will be positioned relative to each other if you make changes with **Change position** and **Change span**.

An additional option for the Location setting is **CSS position**. The Cascading Style Sheet - Positioning setting enables you to place fields anywhere on a Web page, without restrictions.

- **Row** allows you to specify the row value for the field. It must be a positive integer greater than or equal to 1.
- **Column** allows you to specify the column value for the field. It must be a positive integer greater than or equal to 1.
- **Offset** allows you to specify the number of pixels the field is adjusted to, relative to the row or column value. This number can be 0, or a positive or negative integer.
- **Disable tabbing** allows you to indicate whether or not the field can be tabbed to after it is moved. If you select **Disable tabbing**, the user cannot tab to this field. Otherwise, the field remains in the same tab order it was in before it was moved.

**Syntax (Change position):**
```
A*%%%WB 2 FLD <row_number> <column_number>
```

**Syntax (Change span):**
```
A*%%%WB 15 FLD <row_number> <column_number>
```

**Syntax (CSS position):**
```
A*%%%WB 21 FLD <row_number> +/-<pixel_offset> | <column_number> +/-<pixel_offset> | <disable_tabbing>
```

**Note:** The pixel offset must be preceded by + or -. The value for disable_tabbing must be either true or false.

**Example (Change position):**
```
A*%%%WB 2 FLD 14 3
```

**Example (Change span):**
```
A*%%%WB 15 FLD 1 1
```

**Example (CSS position)**
To move an input field to row 30, column 20 (with no offset for either) and allow tabbing:
```
A*%%%WB 21 FLD 30 +0 | 20 +0 | false
```

To disable tabbing and move an input field to row 40 and column 10, with an offset of 20 pixels down and 5 pixels left, respectively:
```
A*%%%WB 21 FLD 40 +20 | 10 -5 | true
```
The **Option Labels** setting is available for input fields that are specified with the VALUES keyword. Fields that are specified with the VALUES keyword are displayed on Web pages as a series of radio buttons. The options displayed in the selection box are the values given as parameters to the VALUES keyword in your DDS source.

This Web Setting allows you to change the label or text that is displayed to the user. For example, if one of the VALUES parameters that your program is expecting is "item8", you might change the text that is displayed in a browser to something more descriptive such as "8 gauge wire". While the text displayed to the user would be "8 gauge wire" the value returned to your program would be the same "item8" that the program would expect.

To use this Web Setting, double-click the **Option Label** field and enter the text that you want.

**Syntax:**

```plaintext
A*%%WB 14 FLD <value>=<label>["|"<value2>=<label2>["|"<value3>=<label3>[...]]]
```

**Example:**

Using numeric values:

```plaintext
A*%%WB 14 FLD 0=value 0|1=value 1
```

Using text values:

```plaintext
A*%%WB 14 FLD 'value0'=value 0|'value1'=value 1
```

**Send to browser**

The **Send to browser** setting can be used for H or hidden DDS fields. This setting modifies the generated JSP files for your records. For example, if you have an H field in a record called LISTCTL, when this setting is used, an HTML hidden input form element will be generated in the JSP file LISTCTL.jsp. Hidden input form elements begin with the HTML code `<INPUT TYPE="hidden"`. Using **Send to browser** enables an H field to become part of the browser environment and for values to be assigned to it based on that environment. For example, the IP address of a user that is determined by a script you added using the **Insert into script** setting could be used to populate the hidden field.

**Syntax:**

```plaintext
A*%%WB 19 FLD
```

**Example:**

```plaintext
A*%%WB 19 FLD
```

**Specifying field replacement variables**

Field replacement variables are allowed in the **Insert HTML**, **User-defined HTML**, **Image**, **Insert script**, **Go to URL**, **Position cursor to field**, and **Call JavaScript** Web Settings. Here are the possible replacement variables, including the deprecated (but supported) variables for each:

**&{FIELD.CURRENTVALUE}**

This is the current value of an HTML field object for an input-capable field in the current record, which can be modified by user input in the browser, or is hidden but uses the **Send to browser** Web Setting.

The deprecated variable is &{FIELD.value}.
&{FIELD.INITIALVALUE}
This is the initial value from the WebFacing run-time for input or output
fields in the current record.

The deprecated variable is &{FIELD.value}.

&{RECORD.FIELD.CURRENTVALUE}
This is the current value of an HTML field object for an input-capable field
in the specified record, which can be modified by user input in the
browser.

The deprecated variable is &{RECORD.FIELD.value}.

&{RECORD.FIELD.INITIALVALUE}
This is the initial value from the WebFacing runtime for input or output
fields in the specified record.

The deprecated variable is &{RECORD.FIELD.value}.

&{FIELD.REF}
This is an HTML field object reference, for example,
document.SCREEN.11_REC1$FIELD1. This is only for input fields or fields that
are hidden but use the Send to browser Web Setting. For output-only
fields, use &{FIELD.ID}.

The deprecated variable is &{FIELD}.

&{FIELD.ID}
This is an HTML field ID, for example, 11_REC1$FIELD1. This is for input
and output fields.

The deprecated variable is &{FIELD}.

&{RECORD.FIELD.REF}
This is an HTML field object reference in the specified record. This is only
for input fields. For output-only fields, use &{RECORD.FIELD.ID}.

The deprecated variable is &{RECORD.FIELD}.

&{RECORD.FIELD.ID}
This is an HTML field ID in the specified record. This is for input and
output fields.

The deprecated variable is &{RECORD.FIELD}.

&{WEBFACINGFORMID}
This is the value for the ID attribute of the HTML form where all of the
fields reside. This can be used for customized JavaScript.

Using special values

You can use special values when you insert user-defined HTML using the
User-defined Web Setting in HTML overrides. The following values can be used
in the context of the User-defined Web Setting on the corresponding fields in your
DDS:

&{*SYSNAME}
This value inserts the name of the system on which the application is
running. It can only be used on the SYSNAME field.

The deprecated value is &SystemName.
&{*USER}
This value inserts the user ID of the current user. It can only be used on
the USER field.
The deprecated value is &UserID.

&{*DATE}
This value inserts the current system date. It can only be used on the
DATE field.
The deprecated value is &Date.

&{*TIME}
This value inserts the current system time. It can only be used on the TIME
field.
The deprecated value is &Time.

Web Settings considerations
Web settings may affect the behavior of your Web application (not the 5250
application). A field’s position, size, and visibility in the Web browser may be
changed through Web settings. These changes can affect how a field overlaps other
fields, and the visibility of fields. Changing a field’s position and size can also
increase the size of the record, which may affect visibility of records in a Web
browser. Be sure you are aware of these potential risks when using Web settings.

To determine the visibility of fields and records in a browser, WebFacing will first
make any changes to position, size, and visibility of fields by using Web settings.
The visibility of fields and records will then be determined using the same rules
that are outlined in the DDS Manual and the Application Display Programming Guide.
For more information on the visibility of overlapping fields, see Keywords for
Display Files > Positional Entries > Location in the DDS Manual. For more
information the visibility of overlapping records, see Keywords for Display Files >
OVERLAY in the DDS manual and Writing Output to the Display in the
Application Display Programming Guide.

Some Web settings, such as Hyperlink and HTML overrides, enable you to add
custom Javascript and HTML. If you add any code with these Web settings, ensure
that the code can run in supported browsers. See “WebFacing support for multiple
browsers” on page 96 for more information on supported browsers.

Visibility > Hide field
When a field is hidden using Web settings, it may no longer overlap other
fields, and the visibility of fields in the browser may be affected.

Example A: FIELD A overlaps FIELD B, as specified in the DDS source. On
the 5250 screen, FIELD B is not visible. If FIELD A is hidden using Web
settings, it will not overlap FIELD B in the browser and FIELD B will be
visible.

Location > Change span, Change position
These Web settings will change the screen area occupied by a field when
displayed in a browser. This may cause a field to overlap other fields. It
may also cause a field to no longer overlap fields that it previously
overlapped. In either case, the visibility of fields in a browser may be
affected.

Example B: As specified in the DDS source, FIELD A and FIELD B are two
fields that do not overlap. Both fields are visible on the 5250 screen. The
Web settings position of FIELDA is set so that FIELDA does overlap FIELDDB. In the Web browser, FIELDDB is no longer visible.

Modifying the span and position of a field using Web settings may also increase the number of screen rows occupied by the record. This may change how records overlap in a browser and affect the visibility of records.

Example C: In the DDS source, FIELDA is a field occupying row 13 in RECORD1. RECORD1 occupies the screen rows 2 to 15. Another record, RECORD2 occupies rows 16 to 23 on the DDS screen. Both RECORD1 and RECORD2 have the OVERLAY keyword specified. The row of FIELDA is changed to 18 using the position Web Setting. Now, in the Web browser, RECORD1 will occupy rows 2 to 18. The application writes RECORD1, and then writes RECORD2 to the screen. On the 5250, both records would be visible. In the Web browser, RECORD1 will be cleared from the screen, because it is overlapped by RECORD2.

Appearance and Text > Override constant text, Hide characters

Overriding constant text using Web settings may change the length of the text constant. Also, specifying the Hide characters Web Setting may make a text constant shorter. In either case, the overlapping of other fields in a browser may be affected. This may change the visibility of fields in a browser.

Example D: A text constant of length 10 is specified in the DDS. The last 5 characters of the text constant overlap FIELDA. On the 5250, FIELDA is not visible because it is overlapped. The constant text is overridden using Web settings so that the length is 5. In a browser, FIELDA is no longer overlapped by the text constant, so FIELDA becomes visible.

Overriding constant text using Web settings may also increase the number of screen rows occupied by the record. This may change how records overlap in a browser, and affect the visibility of records. The consequences of this can be seen in Example C.

Ignored Web settings

Web settings (particularly overriding text constant and changing span and position) will be ignored if they would cause any of the following to occur in the a browser:

- A field would be located beyond the screen boundaries.
- A field that spans more than one row or is located on the last screen row would extend beyond the screen boundaries.
- A subfile control field would overlap the screen area of the associated subfile.
- The number of rows occupied by a subfile record would be greater than on the 5250 screen.
- The width of a horizontal subfile would be greater than on the 5250 screen.

Multiple Web settings of the same type

If more than one Web setting of the same type is specified on a single field, record, or file, the behavior is undefined. In other words, there is no guarantee as to which of those duplicate Web settings will take effect.
Creating custom JavaScript library files

About this task

This topic shows you how to include your own JavaScript libraries in a WebFacing application. Creating your own JavaScript libraries is sometimes necessary when customizing a WebFacing application.

In order for your JavaScript to be published as part of your WebFacing application, it needs to be created in the usr directory. To locate the usr directory, when viewing your WebFacing project switch to the Navigator view, expand your project and expand the folders WebContent > webfacing > ClientScript > usr. Place the file containing your JavaScript library (.js file) in the usr directory. Files contained in this directory will be referenced as include files in the pages sent to the browser by your WebFacing application.

The list of files contained in the usr directory is determined once when the WebFacing application is started. This means that adding or removing files in the directory after the WebFacing application has started will not change the list of files included on each page. Additionally for two or more files there is no fixed order applied to the generated includes.

You can test that your JavaScript library is being included by implementing the function init_usr(). A WebFacing application will make a call to init_usr() after it has completed its load-time processing. To use this call, add an init_usr() function to your JavaScript library file. For example, if you add the following lines to your .js file, an alert will be displayed when the WebFacing application is accessed.

```javascript
function init_usr()
{
    alert("init_usr() called");
}
```

WebFacing Accessibility

The IBM WebFacing Tool provides WebFacing application accessibility that removes barriers for users with special needs, allowing them better to access your applications.

Since the WebFacing tool is unique in that it converts an original 5250 host application into a Web application automatically, the accessibility of the resulting Web application is limited by the information available in the DDS at the time of conversion. Unfortunately, this information is usually insufficient, making additional changes by the WebFacing developer necessary after conversion to further improve the application’s accessibility. This document describes different areas a developer should investigate and possibly modify to make the application more accessible.

Note: You should review your current customizations for accessibility and utilize these and other techniques where appropriate. Please see the Related information section for a more complete list of web accessibility considerations.

- Images
  A visually impaired user who accesses your application using an assistive technology (AT) such as a screen reader must rely on alternative text to associate
meaning with images or graphics. The **Alternative text** field has been added to WebFacing’s Image Web Setting that allows you to easily specify the alternative text.

![WebFacing Image Setting](image1.png)

**Note:** If no alternative text is specified, the field’s current value will be used. For guidelines on how to specify good alternative text, please see the Related information below.

- **Structures**

  For assistive technologies to navigate, interpret, and deliver content, it is important that an application’s markup correctly conveys the intended meaning. Using structural markup (rather than presentation markup) preserves the semantic meaning for users who do not use a visual browser.

  For example, HTML provides the elements `<h1>` to `<h5>` to identify headings. Since there is no equivalent DDS construct, however, WebFacing conversion cannot tell which fields or constants are headings and which are not.

  It may be clear that a heading such as “Parts Order Entry” is a heading when the screen is viewed by someone who is not visually impaired. However, someone using a screen reader, for instance, cannot tell that it is a heading simply by reading the text.

![Parts Order Entry](image2.png)

The challenge is similar for the WebFacing conversion engine. You can, however, use the **HTML overrides** Web Setting to insert heading tags around the field:

1. Open the DDS and find the text you want to identify as a heading.
2. In the Web Settings view, select the **HTML overrides** category, then check **Override generated HTML**, and select the **Insert** radio button.
3. Select the **Before** tab, and insert `<h3 style="display:inline;">`.  
4. Switch to the **After** tab and enter `</h3>` to close the tag.
5. Save and re-convert your DDS.
Similar techniques can be used enhance accessibility by making changes for lists, table headings, and quotations, among other things. Please see the related documents section for further details on making changes to structural markup in a web application.

- Forms

When assistive technologies encounter a form field, it is important that its context and description are available for the user to make sense of it. The recommended way achieving this is by associating fields with their descriptive text using the `<label>` element. In an original 5250 application, there is no mechanism with which to programatically associate a field with its label. Typically, the only available cues are visual, such as how fields are positioned in relation to each other.

This makes it impossible for the WebFacing conversion engine to automatically create such semantic associations in the generated Web application. Modifications for accessibility should be made by the WebFacing developer before deployment.

To define a string of text as a label for a form field, use the `<label>` HTML element. The general syntax used is as follows:

```
<form [...] >
  <label for="inputID" [...] >description</label>
  <input id="inputID" [...] />
</form>
```

For the application illustrated above, suppose that you would like to specify that “Customer number” is the label for the input field to its right.

1. Open the DDS and locate the field you would like to use as a label.
2. In the Web Settings view, select the HTML overrides category, then check Override generated HTML, and select the Insert radio button.
3. Select the Before tab, type `<label for="">`, and place the cursor between the quotation marks.
4. Click the Insert field attribute button, and select the identifier of the field for which you want to specify the label.
5. Click Insert. The Web Setting should now be in the following format:

```
<label for="&{FMT01.CUSTNO.ID}"/>
```
6. Switch to the After tab and enter `</label>` to close the tag.
7. Save and re-convert your DDS.

**Note:** It is possible to define multiple labels for a single form field. In the example we looked at above, the string “F4 for list” would be a good candidate for an additional label.

- Language

WebFacing Version 9.5 introduces changes to dynamically query and use the language used by a job at run time on supporting servers. Since this requires the
job to be started, however, the language information is not available prior to
launch on the application's index or logon pages, so the language defaults to
English.

To specify a different language on the index page, open the file index.jsp, and
add the lang attribute with the appropriate language code to the <html> tag, for
example:

<HTML lang="en">

The same technique can be used for the page logon.jsp.

Note: If your application includes pages that contain multiple languages on the
same screen, you can use the lang attribute on other tags to identify the
language. This can be done using the HTML overrides Web Setting by selecting
the Insert option and specifying lang="encoding=" on the Inside tab.

• Tables

When using a table to display data, relationships between headings and rows or
columns can get quite complicated, making it difficult for assistive technologies
to deliver content in a way that is easy to understand. Changes that a
WebFacing developer can make to improve the accessibility of tables in their
application are described below.

Making sense of data tables can be very challenging without knowing how that
data is organized. Using markup to identify the headings of rows and columns
will make it much easier for assistive technology to deliver content in a
meaningful fashion.

To identify a table cell as the heading of a column or row, use the table heading
tag, <TH>, in combination with the scope attribute.

Following is an example of how to make such a change for a subfile:

1. Open the appropriate JSP (in this example the JSP which was generated for
the subfile control record).

2. In the JSP, locate the table headings tags (indicated by <TD>).

```html
<TR id="1$=2Order">r6" class="trStyle">
  <TD class="trStyle">LastCol = 0;
  <TD colspan=3>&nbsp;</TD>
  <TD COLSPAN=3 ROWSPAN=1>
    <TH nowrap COLSPAN=3 ROWSPAN=1>
      <SPAN id='1$=2Order'>_L1ICTLUnnamed4' onclick="<WH> class="wft_field_d" onclick="Opt">Opt</SPAN>
    </TH>
  </TD>
</TR>
```

3. Replace the <TD> opening and </TD> closing tags with <TH> and </TH>
respectively.

```html
<TR id="1$=2Order">r5" class="trStyle">
  <TD class="trStyle">LastCol = 0;
  <TD colspan=3>&nbsp;</TD>
  <TH nowrap COLSPAN=3 ROWSPAN=1>
    <SPAN id='1$=2Order'>_L1ICTLUnnamed4' onclick="<WH> class="wft_field_d" onclick="Opt">Opt</SPAN>
  </TH>
</TR>
```
4. Add the attribute `scope="col"` to each instance.

```html
<TR id='1'&nbsp;onclick='order()' class='trStyle'>
    <TD colspan=3>&nbsp;</TD>
    <TH NOWRAP colspan=3 rowspan=1 scope="col">...
</TH>
</TR>
```

**Note:** There is a corresponding `scope="row"` value that can be used in situations where you may have horizontal headings. Both values may be used on the same table.

When changing a cell's markup from `<TD>` to `<TH>`, you may find that text has been centered and boldfaced, depending on the styles already applied. Testing your changes will quickly tell you whether any further modifications are needed.

- **Error Messages**
  - To access error messages using a screen reader a shortcut key for headers can be used. Reading out the whole page again will also read out the error message.

- **Windows**
  - For assistive technology to read out when a window comes up, a WDWTITLE keyword needs to be specified with the appropriate title text on the WINDOW record.

For more detailed and a complete list of web accessibility considerations, refer to the related links below.

**Related information:**
- IBM Web accessibility
- IBM Web Accessibility Checklist, W3C Web Content Accessibility Guidelines (WCAG) and US Section 508
- IBM Human Ability and Accessibility Center
- Guidelines, references and resources on Web accessibility
- W3C Web Content Accessibility Guidelines 2.0
- W3C Techniques for WCAG 2.0

**Bidirectional user interfaces in WebFacing**

WebFacing offers the ability to preserve the existing bidirectional (or bi-di) support from a 5250 application when it is Web-enabled.

Computer systems can process text and user interface elements left to right (as is the case in English) or right to left. Software that has support to handle these languages are said to have bi-di support.

Because existing 5250 applications are often already bi-di enabled, converting these applications requires some special steps to ensure that the WebFacing process interprets the direction of the input screens correctly.
Note: WebFacing bidirectional support is not available when using the Firefox browser.

**Bidirectional considerations**

The following are examples of things that can change from left to right or right to left:

- display of bidirectional text
- the relative position of HTML elements including table columns
- tabbing index order for cursor control through the user interface (focus switching from one widget to the next)
- cursor progression through a field
- position of a scroll bar or helper buttons in relation to the widget it controls

Web technology like HTML and XHTML support bi-di with the common base direction attribute `dir` which can have a value of `ltr` (left-to-right) or `rtl` (right-to-left). This attribute can be applied to a number of different tags and the effect depends on the type of widget. Child elements inherit from the parent element. For example, to set the base direction of the entire HTML document to right-to-left, you would apply the attribute to the `<html>` tag: `<html dir="rtl">`.

In DDS, there are three keywords that control these kinds of behavior:

- DSPRL for reversing the displays of screens symmetrically from left-to-right to right-to-left.
- CHECK(RLTB) to make field tabbing proceed right-to-left and top-to-bottom.
- CHECK(RL) to make the cursor progress right-to-left through a field.

The WebFacing Tool applies a unique set of rules to the conversion and enablement of bidirectional user interfaces. Here are some of the fundamental principles:

- The rarely used DSPRL keyword is not supported and the WebFacing HTML page always has a document direction of left-to-right.
- If the CHECK(RLTB) keyword is specified, then WebFacing assigns tab index values based on a grid (see below).
- Any field which had the CHECK(RL) keyword specified on it has the `dir="rtl"` attribute assigned to its generated HTML.

**The tab index layout grid**

Tabbable widgets (elements that should be accessible by using the Tab key like fields and hyperlinks) can be assigned a tab index. Most browsers assume a default tab index order that proceeds through all tabbable widgets from left-to-right and top-to-bottom.

On 5250 screens where the CHECK(RLTB) keyword is used, WebFacing assigns a tabindex to correctly implement right-to-left tabbing order within the left-to-right HTML page. These tabindexes are assigned based on the DDS row and column position. To leave room for moving fields outside of their DDS positions that are constrained to 27 rows and 132 columns at most, WebFacing uses a grid that has 132 rows and 250 columns. (The maximum value for the `tabindex` attribute is 32767.)
You need to understand this tab index grid in the following circumstances:

- adding your own tabbable widgets through Web settings or direct editing of WebFacing JSP files
- using CSS-P to move a tabbable widget enough to change its relative position in the tab index order

In the case of new tabbable widgets added through Web settings, you can use the substitution variable &{TABINDEX}. This will calculate the correct tab index order of your widget at conversion time.

**Web Setting Limitations**

**Note:** For information on further limitations, refer to the technote [Limitations when Converting and Running BiDirectional Applications with WebFacing](#).

The Arabic 420 EBCDIC code page does not have the following characters:

```
\ 
{ }
[ ]
```

Therefore, the tab index substitution variable is not supported for members using the Arabic 420 code page, nor are other Javascript constructs, like arrays, that rely on these characters.

In addition, Arabic becomes unshaped because it is stored in logical format in the member which is visual format.

**Related tasks:**

- "Setting tab order for bidirectional applications"

WebFacing automatically assigns right-to-left tab index order but if you customize with new elements, you should set their tab order too.

**Setting tab order for bidirectional applications**

WebFacing automatically assigns right-to-left tab index order but if you customize with new elements, you should set their tab order too.

**Before you begin**

Start with a 5250 application that uses the CHECK(RLTB) keyword. Convert it using the WebFacing Tool. The resulting Web pages will have a right-to-left tab order set for tabbable elements like fields and hyperlinks. This order is based on a 250 x 131 grid.

**About this task**

If you reposition elements with Web settings, the tab index reflects the customized position.
You can add your own elements like fields and hyperlinks. WebFacing handles the tab index for a Hyperlink Web setting, but if you create a hyperlink using any other Web setting (like Insert HTML After) or any other tabbable element, you need to add a tab index yourself.

To add a tab index:

**Procedure**

Select **Right-to-left tab index** from the Insert field attributes dialog.

This adds a substitution variable into your DDS source for any customizations you added yourself:

&{TABINDEX}

Alternatively, you can add the &{TABINDEX} substitution variable directly into your DDS source. This resolves to the appropriate number. An example for HTML would be:

```html
<input tabindex="&{TABINDEX}"/>
```

But note the following considerations:
- It is up to you to add this substitution variable.
- This variable is processed at conversion time and does not work for customizations done to the JSP after conversion.
- If you add more than one tabbable HTML element at the same row and column (for example, using Insert HTML After Web setting to add a hyperlink after a field) then both elements end up with the same tab index. The conflicting tab indices are visited from left to right since that is the document order.

**Results**

**Related concepts:**

“Bidirectional user interfaces in WebFacing” on page 77

WebFacing offers the ability to preserve the existing bidirectional (or bi-di) support from a 5250 application when it is Web-enabled.

---

**Querying user access to your WebFacing application**

**About this task**

The WebFacing Environment API enables you to check whether a user is accessing your application through a Web browser or through 5250 emulation. Use this API when you would like to change the behaviour of your program according to the type of access a user has. For example, there may be an extra field or different text..."
that you would like to display if your program is being accessed through a
browser, but you would like to suppress display of the field or text if 5250
eulation is being used.

The WebFacing Environment API is called QqfEnvironment and is part of the
WebFacing server runtime. The external procedure name QqfEnvironment is case
sensitive. It is a procedure packaged in a service program called QQFENV that is
located in the QSYS library. The API returns 1 if the application is running under
WebFacing and 0 if it is running under 5250 emulation.

The examples below show how to use this API. In the RPG sample, the external
procedure QqfEnvironment is defined with the DSpec QQFENV. In this example,
the QQFENV DSpec has been given the same name as the service program and has
been defined as an integer since the procedure returns 0 or 1. A DSpec rc has also
been defined to hold the value 0 or 1 when the Eval rc = QQFENV is performed.
The RPG program then uses the value of rc to conditionally determine the
behaviour of the program and what will get displayed on the DDS display.

In the DDS sample below, if the value for rc in the RPG module is NOT 1, the text
"Application is not running in the WebFacing environment" will be displayed. If
the value for rc is 1, the text "Application is running in the WebFacing
environment" will be displayed.

When you are creating a program to use this API:
1. Use the CRTRPGMOD command to create a module with your RPG code that
   is calling the API. An RPG module needs to be created because it is using a
   procedure not in the program.
2. When you create your program (CRTPGM) use the BNDSRVPGM keyword to
   bind your RPG module with the QQFENV service program in QSYS.

```
ACHINE CF E Workstn
DQQFENV PR 10I 0 Extproc('QqfEnvironment')
  Drc S 10I 0
  C Eval rc = QQFENV
  C Eval FLD001 = rc
  C Dow NOT *IN03
  C If rc = 1
  C Eval *IN01 = *on
  C Else
  C Eval *IN01 = *off
  C EndIf
  C Exfmt FMT01
  C EndDo
  C Eval *inlr = *on
```

*Figure 1. RPGLE sample*
Chapter 6. WebFacing interoperability with HATS and other Web applications

Both the IBM WebFacing Tool and the HATS Toolkit offer ways to enable your core business applications for the Web. The IBM WebFacing Tool converts your original DDS source files, providing a Web-based user interface for your 5250 host application. The HATS Toolkit transforms 5250 data streams into HTML dynamically, which enables applications to be migrated to the Web without accessing or modifying the original DDS source code.

The interoperability features between HATS and WebFacing allows you to leverage the advantages of both WebFacing and HATS by combining applications with converted and unconverted DDS source. With this technology your applications can be quickly and easily modernized using WebFacing and HATS.

If you would like to take your applications even further you may want to iteratively add new Web applications to take advantage of the latest Web technologies. The WebFacing application bridge allows you to seamlessly interconnect new Web applications (using Enterprise Generation Language (EGL) to create a new Web application, for example) with existing WebFacing and/or HATS applications.

This allows WebFacing applications to transfer control and data to and from other Web applications, such as one written in EGL, giving you the ability to combine HATS, WebFacing and EGL applications into a single application. This goes beyond getting data from IBM i in a Web application using direct database access or by calling a batch program via a Web Service. It allows you to link your new Web application to interactive IBM i WebFacing and HATS applications and share data.
For details about how to implement the WebFacing application bridge, see the Create an Application Bridge between WebFacing and an EGL application tutorial at the IBM Rational Host Access Transformation Services Education Assistant Web site.

WebFacing interoperability with HATS applications

The interoperability features between WebFacing and HATS provide the ability to combine WebFacing applications with 5250 dynamic data stream transformations using HATS, within the same Web application.

Interoperability, in this context, means that a single WebFacing Web application interoperates with, or works along side, a single HATS application. The combined Web application has the capabilities of both tools, with some limitations, but the capabilities are cleanly separated within the Web application. To create a combined Web application, a single WebFacing Web project (with the project type HATS/WebFacing enabled project) is linked with a single HATS project and packaged into a single enterprise application, or .ear file. This enterprise application contains the runtimes of both products, along with a common runtime component that is used to create and interact with a shared connection to the 5250 backend host.

For details about how to implement interoperating HATS and WebFacing applications, see the Create a Linked WebFacing and HATS application tutorial at the IBM Rational Host Access Transformation Services Education Assistant Web site.

Related reference:
“Considerations for linked HATS/WebFacing applications”
This reference document describes technical and security considerations when creating a linked HATS/WebFacing application.

Considerations for linked HATS/WebFacing applications

This reference document describes technical and security considerations when creating a linked HATS/WebFacing application.

Linked HATS/WebFacing applications have several technical and security considerations. In general, any limitations posed by a standalone HATS application or a standalone WebFacing application will also apply to linked HATS/WebFacing projects. In addition, you should consider the following information when designing and deploying your own projects.

Technical considerations

• Some changes made to the individual HATS and WebFacing applications will not affect the linked HATS/WebFacing project. Specifically, any configuration options that are specified in the Linked HATS/WebFacing project wizard (such as host name or port, code page, and screen size) can only be modified after
project creation by editing wfhats.xml, or by rerunning the project wizard.
Changing the connection settings in the HATS or WebFacing project after linking
will not affect the linked project.

- HATS connect macros will run when the HATS application is first accessed
  (from initial entry into the linked application, or from accessing a host screen
  that has not been converted by WebFacing). Therefore, you should not create a
  connect macro intended for the sign-on screen if your linked application will be
  started from WebFacing. Disconnect macros will only run if the linked
  application is terminated from the HATS runtime.

- Start and Connect events defined in a HATS project will run when the HATS
  application is first accessed. If the linked application is started from WebFacing,
  actions defined in these events will not be performed until a host screen that has
  not been converted by WebFacing is accessed. Therefore you should not, for
  example, play a macro intended for the sign-on screen in the Connect event if
  your linked application will be starting from WebFacing.

- If you use the HATS Administrative Console to change license settings while
  running in Run on Server mode, you must restart your linked application in
  order for the changes to take effect. If you use the HATS Administrative Console
  to change license settings while running in Debug on Server mode, the changes
  will not affect the linked application. This is because the HATS Administrative
  Console modifies the runtime-debug.properties file while running in Debug
  mode, and the linked application uses runtime.properties to determine license
  settings. Also note that if you are running in Debug on Server mode, you should
  verify that the license settings in runtime.properties match the license settings in
  runtime-debug.properties.

- For linked HATS/WebFacing projects, the only supported application server for
deployment is IBM WebSphere Application Server.
- For the list of supported Web browsers for linked HATS/WebFacing
  applications, refer to the HATS prerequisites section of the HATS Knowledge
  Center.

- A linked HATS/WebFacing application must consist of one HATS/WebFacing
  enabled project and one HATS Web project. Other types of HATS projects (Rich
  Client, Portal, and Enterprise JavaBeans) cannot be linked.

- The HATS applet is not supported for linked HATS/WebFacing applications. Do
  not configure your project to use the applet.

- Any screen customization made using HATS will have no effect if the screen is
  also converted using WebFacing. This is because the WebFacing conversion will
  be used at runtime.

- When running a linked HATS/WebFacing application, HATS connection pooling
  is not used, even if it is configured in the HATS project.

- When running a linked application, the connection is persistent. Nonpersistent
  features, such as fail-over, will not function correctly.

- Workstation ID support is not available for the linked HATS/WebFacing
  application. You can link a HATS application that is configured for a special
  workstation ID; however, this setting will be ignored at runtime.

- HATS projects can have both a main transformation connection and one or more
  background connections to the same or different hosts. Linked project features
  and connection settings only apply to the main transformation connection.

- WebFacing supports limited-capability users on IBM i V6R1 and later. To use a
  profile with limited capabilities on V6R1, you must start the linked application
  from the HATS interface.
In some cases, you may want to deploy your HATS application more than once to the same server. For example, the same HATS application can be deployed as a standalone application and part of a linked application. In order to deploy your HATS application more than once to the same server, you must provide a unique context root for each instance. To provide a unique context root:

1. Right-click the project and select Properties.
2. In the left pane, select Web Project Settings.
3. Enter a new, unique context root in the Context Root field, and click OK.
4. The Modifying Context Root dialog appears. Select Yes to fix links that reference the context root.
5. Change the <display_name> value in the project's web.xml file.

If you update the context root of either project after you have linked them together, you will need to update the wfhats.xml file to reflect the change. The wfhats.xml file is located in the root folder of the linked HATS/WebFacing project.

- You cannot access multiple linked applications at the same time in the same browser instance. For example, opening the browser and launching application A, creating a child browser window, and then accessing application B in the child window is not supported. A child browser is opened in Internet Explorer by doing Ctrl-N or File->New->Window. The same rule applies for tabs in Internet Explorer V10 and V11. In addition, if you wish to access application A and then application B in the same browser, you must first properly disconnect and exit application A before accessing application B. This only applies if applications A and B have the same context roots.

- You may see unexpected results when accessing a linked HATS/WebFacing application if you use the browser refresh and back buttons. See the HATS FAQs for details.

- Though the HATS toolkit allows you to create macros that navigate through DDS-based screens that have been WebFaced, the macros may fail when the application attempts to display the WebFaced screens. Macros cannot handle WebFaced screens if their display files were opened prior to the start of the macro execution, and the application will end in error.

Security considerations

When securing your linked HATS/WebFacing application using Tivoli® Access Manager, you must set two configuration options to allow your application to function properly.

- In the WebSEAL daemon configuration file (webseald.conf), enable the preserve-base-href setting:
  ```
preserve-base-href = yes
  ```
- Create all junctions with the -j parameter, to allow WebSEAL to provide a junction identifier cookie to the browser.

For more information on these options, see the Tivoli Access Manager documentation for details.

Using the WebFacing application bridge to interoperate with other Web applications

The WebFacing application bridge enables you to seamlessly interconnect new Web applications (written with EGL, for example) with existing WebFacing and/or HATS applications.
With this technology your applications can be quickly and easily modernized using WebFacing and/or HATS, and new Web applications can be added iteratively to take advantage of the latest Web technologies.

The application bridge allows WebFacing applications to transfer control and data to and from other Web applications such as EGL, giving you the ability to combine HATS, WebFacing and EGL applications into a single application. This goes beyond getting data from IBM i in a Web application using direct database access or by calling a batch program via a Web Service. It allows you to link your new Web application to an interactive IBM i application and share data.

The WebFacing application and the corresponding Web application can be packaged as an EAR file for deployment. The EAR file can be created using the Java EE perspective of the IDE. However, the WebFacing application and the Web application can also be deployed as separate EAR files within the same application server instance.

To support the interoperability between a WebFacing application and another Web application, a DDS linkage record should be created and the Application bridge Web Setting should be set for that record. A DDS linkage record is standard DDS record containing fields defined as hidden fields (usage H), that you can use to transfer data. The DDS linkage record is never actually displayed. It is simply used to transfer data and control to the Web application specified in the Target application URL of the Application Bridge Web Setting when your IBM i application does a WRITE on the record.

Please refer to the Application Bridge Web Setting in “Using Web Settings with your DDS source” on page 52 for further information on creating a DDS linkage record.

The WebFacing application bridge supports the transfer of control and data in the following scenarios:

1. A WebFacing application calls another Web application with data; the other Web application returns control back to WebFacing with data updates.
   - The IBM i application does a WRITE on the WebFaced linkage record (transfer control and Data to another Web application) immediately followed by a READ and waits for the data to come back.

2. A Web application calls a WebFacing application. The WebFacing application will execute, and then return data back to the other Web application.
   - The IBM i application does a WRITE on a WebFaced linkage record with the FRCDTA keyword and then terminates the application.

Scenario 1 passes data from the WebFacing application to the Web application specified in the Target application URL of the Application Bridge Web Setting when the IBM i application does a WRITE and READ on the DDS linkage record. The WebFacing application will then wait for a response from the Web application. The Web application will then read the data via the “LinkageData” request attribute, make any updates and send back the updates to the WebFacing application at which point the WebFacing application will resume control.

Scenario 2 is supported through WebFacing programmatic invocation, which allows you to specify any CL command with or without parameters to start a WebFacing application. The WebFacing application runs, then passes data back to the Web application specified in the Target application URL of the Application Bridge Web Setting when the IBM i application does a WRITE on the DDS linkage record.
record. For this scenario, the FRCDTA keyword should be specified on the DDS linkage record so that the application can terminate successfully. Please refer to “Programmatically invoking WebFacing applications from other Web applications” on page 89 for further information on programmatic invocation.

The WebFacing application bridge can be used to make WebFacing and/or HATS applications interoperate with other Web applications. HATS is enabled with WebFacing when a HATS and WebFacing project are linked using the interoperability feature. For linked HATS/WebFacing projects the application bridge can be used as in the following scenarios:

1. HATS and Web/EGL: only the linkage record is converted/WebFaced in your WebFacing project; allows HATS projects to pass control and data to Web/EGL.

2. WebFacing, HATS, and Web/EGL: part of your application is WebFaced and the other part runs as a HATS project; allows WebFacing and/or HATS projects to pass control and data to Web/EGL.

Passing control from WebFacing to another Web application

In both scenarios above, a WRITE is done on a DDS linkage record to pass control from WebFacing to another Web application. This is done using the Java EE forward method on the URL specified in the Target application URL of the Application Bridge Web Setting. The WebFacing application will use the first part of the target application's URL as the context root and forward the WebFacing application to the remaining part of the URL. If the target application’s URL is /appContextRoot/appEntryPoint, then the WebFacing application will use appContextRoot as the context root and forward to the appEntryPoint. The WebFacing application will add the request attribute "forwarded" with the value "WF" in the request object prior to forwarding. This will allow the receiving application runtime to take any special actions, if necessary, on receiving control from a forward.

Passing control from another Web application to WebFacing

In Scenario 1, when passing control back to the WebFacing application and in order to have proper data updates, the Web application should add the attribute "forwarded" in the request object prior to forwarding. The request attribute should be "EGL" for a request forwarded from an EGL application and "CUSTOM" for a request forwarded from other Web application. This will allow the receiving WebFacing application to take any special actions, if necessary, on receiving control from a forward. The entry point for the WebFacing application is "/webfacing/WebFacing.action".

Exchanging data between WebFacing and other Web application

Application bridge data delivered via the DDS linkage record to the WebFacing application is saved in a Java HashMap, with field name as the key in the HashMap and field value as the value for the associated key. A request attribute called "LinkageData" will be used to provide access to the HashMap.

Field values in the DDS linkage record will stored as Unicode string data in the HashMap. The other Web application will update the fields in the HashMap as necessary and send the HashMap back to the WebFacing application via the "LinkageData" request attribute. A field value should contain those characters which are allowed by the corresponding DDS data type and have the proper length; otherwise, it will be discarded by the WebFacing application.
Note the following:

1. Character fields and DBCS fields: no formatting is performed on these data types.
2. Numeric fields: a decimal point "." will be inserted into the proper location in the data if the decimal position for a field is greater than 0; a negative sign "-" will be added to the beginning of the data for a negative number. For example, -000123.45 is the proper format for a numeric with length 8 and decimal position 2.
3. Date fields: ISO format YYYY-MM-DD is used.
4. Time fields: ISO format hh:mm:ss is used.
5. Timestamp fields: ISO format YYYY-MM-DDThh:mm:ss.uuuuuu is used.

Note: When a DDS linkage record does not contain any fields, the request attribute "LinkageData" value will be set to null.

Related concepts:
"Using Web Settings with your DDS source" on page 52

Related information:
"Programmatically invoking WebFacing applications from other Web applications"

Programmatically invoking WebFacing applications from other Web applications

WebFacing applications can be invoked programmatically from other Web applications. This provides a way to integrate WebFacing generated user interfaces with existing Web applications.

WebFacing applications are launched from URLs. Typically, in a WebFacing application, each defined CL command URL is represented as a button that a user clicks to start the application. During WebFacing project creation, URLs are written to an invocations.jsp file, and after the application is deployed, these are the buttons users click to start the application.

With control of how WebFacing is invoked, it is also possible to use alternative methods of authentication. All user authentication can now be performed in a custom servlet before WebFacing is called. The authentication mechanism used must be able to provide the WebFacing application with IBM i user credentials so that it can access IBM i resources.

Here is a simple example for a determining the CL command to use to launch a program:

**URL constructed by a controller servlet:**

```
WFInvocation.action?clcmd=call%20ordentr
```

In this example, ordentr is the name of a program to launch. The value ordentr could be determined by a servlet and assigned to a variable such as `orderProgram`. Your servlet could construct the URL string using the value determined for `orderProgram` and assign it to a variable `newURL` using a line like:

```
newURL = "WFInvocation.action?clcmd=call " + orderProgram;
```

`newURL` could then be used as the forward or redirect URL for your forward() or sendRedirect() methods.
In this example, the complete URL used by the browser, if sent as a redirect, would look like:

http://<hostname>:<port>/<application>/WFInvocation.action?clcmd=call%20ordentr

The example shows the full URL beginning with http://<hostname>:<port>/
<application>/ . The value for newURL is the string after this. That is, the string:
WFInvocation.action?clcmd=call%20ordentr . In an example like this, the first part
of the URL: http://<hostname>:<port>/<application>/ , represents the host, port,
and context root for the application. If your controller servlet is in the same context root, it is not always required for the servlet to determine the entire URL. If necessary, though, you could code the servlet to construct a string for the fully qualified URL.

Note: The characters %20 in the URL represent a space character as encoded when sent to a browser. Space characters generally cannot be used explicitly in URLs. In the example where the URL string is being constructed and assigned to newURL , the space is present in the part of the string just after clcmd=call11 . The reason for the space in the string is that the example represents a CL command call ordentr . In the URL string that is being constructed , it is not necessary to add %20 directly. The server will add this encoding if necessary.

### URL parameters that can be determined dynamically

- **clcmd** CL command to launch the program.
- **host** Host name where the original 5250 application is located.
- **port** Port number of the WebFacing server on IBM i.
- **userid** userid is used to log onto the application. Note: If a forward() method is used in your controller servlet, URL parameters are only sent within the application server tier (middle tier). Using sendRedirect() instead exposes URL parameters to the browser. In this way, sendRedirect() is less secure because information such as user IDs and passwords can be revealed in a browser’s location field, or if a user views the properties for the page they are using.

  **password**

  Password used to log onto the application. Note: If a forward() method is used in your controller servlet, URL parameters are only sent within the application server tier (middle tier). Using sendRedirect() instead exposes URL parameters to the browser. In this way, sendRedirect() is less secure because information such as user IDs and passwords can be revealed in a browser’s location field, or if a user views the properties for the page they are using.

- **inv** The invocation name for the WebFacing CL command used to launch the application. If values such as host, user ID and password are defined for a CL command, these values will override the general values specified for a project. To view the invocation name for a CL command, open the WebFacing perspective in the workbench, click the WebFacing Projects tab, expand your WebFacing project, expand the CL Commands folder and click the label for the command. The value for the invocation name can be viewed in the Properties pane. (If the Properties pane is not displayed in the WebFacing perspective, to open it, click Window > Show View > Properties.) To edit the invocation name, right-click the CL command in the WebFacing Projects view, and select Properties.
Filtering programmatic invocation commands

You can specify the CL command prefixes that will be allowed for programmatic invocation using the clcmd parameter. Programmatic invocations that use the clcmd parameter and specify a value that does not start with a prefix that you allow will be blocked from running. The default is to not allow any programmatic invocations that override the CL command to be run.

For migration of projects from WebFacing V6, a special value of *ALL will be included to allow all CL commands to be run.

```xml
<context-param>
  <param-name>WFCLCMDAllowed0</param-name>
  <param-value>*ALL</param-value>
</context-param>
```

If the clcmd parameter is not used or if the clcmd values that are used are known, you should remove the *ALL value and provide values as indicated below.

To specify which command prefixes are allowed, edit the source of the web.xml file for your WebFacing application. Add parameter names consisting of WFCLCMDAllowed, followed by additional text to make each parameter distinct. Then add a parameter value for each to specify the command that is being allowed. The following example allows all commands that start with CALL MYCMD and GO MYMENU.

```xml
<context-param>
  <param-name>WFCLCMDAllowed0</param-name>
  <param-value>GO MYMENU</param-value>
</context-param>
<context-param>
  <param-name>WFCLCMDAllowed1</param-name>
  <param-value>CALL MYCMD</param-value>
</context-param>
```

Provide additional context parameters with values as required.

This will allow clcmd values such as CALL MYCMD isOK or CALL MYCMD PARAM(ONE), but disallow values such as CALL MY or CALL OTHERCMD. Similarly for GO MYMENU, allowed commands must start with the specified string. Case is ignored for the comparison.

Note: This only affects programmatic invocation using the clcmd parameter. WebFacing invocations using the inv parameter are unaffected.

Example URLs

```html
WFInvocation.action?clcmd=call%20ordentr
  Host and port in the deployment descriptor file web.xml is used. Prompt the user to logon.

WFInvocation.action?inv=INV1
  Host, user ID, password, and CL command will be retrieved from the deployment descriptor file web.xml. CL command invocation name is INV1. Prompt only when user ID or password is missing or prompt is specified. Print error message if user ID or password are incorrect.

WFInvocation.action?inv=INV1&host=SYSTEM1&userid=WEBFACING&password=WEBFACING
  CL command invocation name is INV1. Host, user ID, and password are passed by the URL. Multiple parameters are separated by &.
```
WFInvocation.action?clcmd=call%20ordentr&host=SYSTEM1
&userid=WEBFACING&password=WEBFACING

CL command call ordentr is passed by the URL. Host, user ID, and
password are passed by the URL. Multiple parameters are separated by &.

**Fully qualified example**

http://<hostname>:port/<application>/WFInvocation.action?clcmd=call
%20ordentr&host=SYSTEM1&port=4004&userid=WEBFACING
&password=WEBFACING

**Note:** in this example the strings <hostname> and <port> refer to the
hostname and port for the application server where the WebFacing
application is deployed. <application> is the context root for the deployed
application. The example shows the following values being passed by the
URL: CL command is call ordentr. Host where the 5250 application is
located is SYSTEM. The user ID is WEBFACING. The password is
WEBFACING. Multiple parameters are separated by &.

---

**Servlet methods for calling a WebFacing application programatically**

There are two servlet methods for calling a WebFacing application programatically.
They are:

- forward() -- The forward() method is in the javax.servlet.RequestDispatcher class.
- sendRedirect() -- The sendRedirect() method is in the

The most important differences between these two methods are listed below:

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A Server-side call. This method calls the other resource, retrieves its output, and returns it to the client.</td>
<td>Sends the HTTP 302 status code to the browser. The browser will automatically re-connect to the URL of the resource. In this case, the browser knows that the output came from the other resource.</td>
</tr>
</tbody>
</table>
Chapter 7. Testing and deploying WebFacing applications

About this task

After you have created and converted a WebFacing project with the WebFacing Tool, you can test your WebFacing project before deploying it to an application server. See “Testing WebFacing applications” for more information on testing WebFacing applications.

After you have tested your WebFacing project, you will need to deploy it to an application server. WebSphere Application Server deployment involves three general tasks:

1. Export your WebFacing project as an EAR or WAR file. An EAR file is a compressed Enterprise Application Archive. A WAR file is a Web application archive file.
2. Use the WebSphere administration console to install the exported EAR file.
3. Start the Web application

For more information on deploying WebFacing applications, see “Deploying WebFacing applications” on page 94.

Testing WebFacing applications

About this task

The WebSphere Test Environment can be used as a convenient way to test your WebFacing applications. It allows you to test your WebFacing project on a WebSphere Application Server (WAS) running on your workstation. As WAS on IBM i requires significant system resources, some developers may prefer to use the test environment during their development stage.

To setup and use the test environment, follow these steps:

Procedure

1. After the WebFacing project has been converted, activate the WebFacing Projects view. Right-click the WebFacing project that you would like to test and select Run As > Run on Server.
2. The Run on Server dialog opens. You can create a new server, or choose to publish your project to an existing server. If you are using the test environment for the first time, you will be prompted to create a new server. If you have previously created servers, you can select from any servers that you have already created. The available server types depend on the options chosen during the installation of the IBM Software Delivery Platform.

If you want this project to always deploy to the same server, select the Set server as project default (do not ask again) check box. When you have selected your server, click Finish.
3. If you are creating a new server, the server is created and started. If you are using an existing server, the existing server is started (if it is not already running). Your project is also deployed to the test environment.
When the server startup has completed, the server console displays a message: "Server open for e-business" and a Web browser opens in the editor area, displaying the index.jsp page for your WebFacing application.

Related tasks
Chapter 7, “Testing and deploying WebFacing applications,” on page 93
“Launching your application in a browser” on page 95

Deploying WebFacing applications

About this task
WebFacing applications can be deployed to the WebSphere Application Server or the IBM i integrated Web application server.

Which application server should I use?
- Use the Websphere Application Server Express or Base if you are deploying WebFacing applications, WebFacing/HATS linked applications, or WebFacing applications that use the application bridge.
- Use the Websphere Application Server Network Deployment (ND) if you are deploying WebFacing applications requiring support for high availability, server failure bypass (failover), and server load-balancing.
- Use the IBM i integrated Web application server if you are deploying WebFacing Web applications and have limited IBM i system resources that cannot optimally run the WebSphere Application Server. The IBM i integrated Web application server does not support high availability and failover.

Deploying to WebSphere Application Server

About this task
1. Export your WebFacing project as an EAR file.
   a. From the WebFacing Tool IDE, select File > Export. The Export wizard is launched.
   b. In the Export wizard, under Select an export destination, select Java EE > EAR file. Click Next.
   c. In the EAR Export screen, select the WebFacing project that you would like to export in the field, EAR project.
   d. Next to the Destination field, click Browse to select the directory to which you would like to export the EAR file and to choose a name for the EAR file. Select a Target runtime. The browser-based Administrative Console enables you to install EAR files that are either on your workstation or that are already in the IFS on your IBM i server. If you would like to first transfer your files to IFS, you can do so by using Remote System Explorer in the workbench, by FTP, or by creating a file share on your server and mapping a network drive on your workstation to that file share.
   e. Click Finish to create the EAR file.
2. Use the application installation wizard to install the EAR file.
   a. Log on to the browser-based Administrative Console. Generally, after you have installed WebSphere Application Server on your server, the URL for the Administrative Console is http://server_name/admin_port/admin/ for WebSphere Application Server.
   b. In the navigation on the left side of the Administrative Console, select Applications > New Application > New Enterprise Application.
c. If you want to install the WAR or EAR file from your workstation, select the Local file system button and browse to where you have exported the WAR file. If you have transferred the WAR or EAR file to the IFS on your server, select the Server path button and enter the path where the file is located.

d. When asked, "How do you want to install the application?", select Fast Path.

e. If you are installing a WAR application, in the Context Root field, specify the name by which users will be able to access your application, prefixed with a forward slash. The default context root is the project name. For example, if you used the same name as you did for your WebFacing project, such as /webfacing1, users will be able to access your application through a browser with an address like: http://myservername/webfacing1.

   If you are installing an EAR application, the Context Root field should be left blank.

f. Click Next. The next series of pages will display additional information about your application; read the information; accept the default values; and click Next to continue.

g. After you have completed the installation steps, the console displays a link Save to Master Configuration. Follow this link then click the Save button.

3. Start the Web application.

   a. In the Administrative Console, select Applications > Application Types > WebSphere enterprise applications.

   b. Check the box next to the application that you have just installed.

   c. Click the Start button.

---

**Launching your application in a browser**

**About this task**

Before you can access your WebFacing application in a browser, the application must first be started in WebSphere Application Server. For information about starting your application in the application server, see the section of this guide: Chapter 7, "Testing and deploying WebFacing applications," on page 93.

Once your application is started, it can be accessed through a browser with an address of the form:

http://host_name:<host_port>/<context_root>/

For example, if your server is MYHOST, your port is 9080, and the context root value that you entered for the application in the administrative console is webfacing1, then you would access the application with an address like:

http://MYHOST:9080/webfacing1/

To view the host name for your IBM i system, run the command CFGTCP from an IBM i command line. On the Configure TCP/IP menu, choose option 12 Change TCP/IP domain information and make note of the Host name value. This is the value, without the quotation marks, that you must specify for your host name.

You can also include the file name index.jsp at the end of your addresses. For example, you can use an address of the form:

http://host_name:<host_port>/<context_root>/index.jsp
The index.jsp file is generated by the WebFacing conversion. Most Web server configurations do not require that it is included as part of a URL. That is, in most configurations, if a user enters a URL that ends with the <context_root>, the server by default displays the index.jsp file for that context.

When you access the page, you see links that correspond to the Command labels you chose for the CL commands that you created in the WebFacing Project wizard. By clicking these links, you launch the corresponding applications on your IBM i server and the output for the application is displayed in your browser.

You can create your own pages to launch your applications. If you want to do this, you can use the webfacing/services/invocation/html/invocations.jsp page as a starting point or add the links in invocations.jsp to your own pages. To use the links, open up a source editor and look for HTML tags that start with <a href= and end with </a>. The anchor tag <a> identifies an HTML link. The href parameter is a required attribute that specifies the link's destination. For example: <a href="WFInvocation.action?inv=INV1">CALL ORDENTR</a> Launch order entry.

Your links will vary from this depending on the command you are using to call the application. To use the link, copy and paste the link into the source for your HTML pages.

To use any of these links, you will need to copy the javascript source from the <head> section of index.jsp into the <head> section of your own page. This source begins with the tag: <script language="javascript"> and ends with the tag </script>. If you create your own pages, they must be included in the Web application that was created for your WebFacing project.

Related reference:
"WebFacing support for multiple browsers"
WebFacing projects can be run in the Microsoft Internet Explorer, Mozilla Firefox, and Google Chrome browsers. This reference document describes the differences between the presentation of WebFacing pages in Internet Explorer and Firefox browsers.

### WebFacing support for multiple browsers

WebFacing projects can be run in the Microsoft Internet Explorer, Mozilla Firefox, and Google Chrome browsers. This reference document describes the differences between the presentation of WebFacing pages in Internet Explorer and Firefox browsers.

#### Currently supported browsers

For the list of supported releases of Web browsers, refer to the detailed HATS system requirements from the [HATS System Requirements](#).

**Attention:** Unless client customizations are explicitly coded for multiple browser support, they might not work in Firefox. Avoid using functions only supported by a particular browser in your customizations; for example, using `document.all()` (an Internet Explorer function) in a customization would cause the application to fail when running it in a Firefox browser.
Differences in WebFacing behavior when running in Internet Explorer and Firefox

Different browsers can vary in how they present Web pages, affecting both behavior and rendering. The following list shows some differences in how WebFacing applications behave and are presented in Internet Explorer and Firefox.

- Closing WebFacing connections and session reuse:
  - When running WebFacing applications in the Firefox browser, closing of the browser window is not detected, and the WebFacing job continues to run.
  - Internet Explorer 8 or Firefox browser windows started from same shortcut share the same session. WebFacing allows only one connection per session and issues (by default) a WebFacing Session Reuse Error when multiple connections are detected. Expect this condition to occur more often in Firefox browsers unless you explicitly exit from your WebFacing applications.

- Rendering differences:
  - The layout of an HTML table might be rendered slightly differently from one browser to another.
  - The font of characters might be rendered differently from one browser to another.
  - In Firefox, invalid characters that appear as spaces, boxes, or not at all in Internet Explorer might appear as '?' or as hex values.

- In Internet Explorer, the cursor is positioned at the beginning of a filled input field. Since Firefox does not support overwrite mode, the cursor is positioned at the end of a filled input field.

- Formatted fields work differently in Internet Explorer and Firefox:
  - Initial caret position and html rendering in a date field:
    - In Internet Explorer, the caret appears at the beginning of the field and the month, day, and year characters are close together.
    - In Firefox, the caret follows the first character of the field and the month, day, and year characters appear to have spaces between them.
  - Character overwrite:
    - In Internet Explorer, when the caret is in the middle of an input field and you type a character, the typed character overwrites the next character in the field.
    
    Note: This is only if overwrite mode is on, which is the default for Web projects.
    - In Firefox, the caret does not change position, and no characters are changed. This is because Firefox only supports Insert mode.
  - Behavior of the caret when it is at the end of a field:
    - In Internet Explorer, typing a character, for example '4', with the cursor at the end of a filled EDTMSK segment causes tabbing to the next segment and overwrites its first character with '4'. If the next segment originally contained '00', it would now contain '40'.
    - In Firefox, typing the same character with the cursor at the end of a filled EDTMSK segment causes tabbing to the next segment and replaces the segment's entire contents with '4' in the previous example.

- WebFacing bidirectional support is not available in Firefox.
Transferring files using Remote System Explorer

About this task

You can use Remote System Explorer to transfer WAR and EAR files to your IBM i host. With Remote System Explorer you can move files between your workstation and your server without using FTP or NetServer file sharing.

The WebSphere Application Server administrative console for versions 8.5.5 and later are browser based. WAR and EAR files can be installed directly from your workstation or from a path in your server's file system if they have already been transferred there.

To transfer files using Remote System Explorer, follow these steps:

Procedure

1. First, export your WebFacing project as a WAR or EAR file. Using WAR files as an example: In the workbench select File > Export. Choose WAR file in the Select an export destination dialog. Click Next. In the WAR Export dialog, select your WebFacing project in the Web module field. Click Browse to choose a location on your local file system to export the file to (for example, X:\war-files). Click Finish.

2. Open the Remote System Explorer perspective. To open the perspective, select Window > Open Perspective > Remote System Explorer.

3. Create a new connection if necessary. When initially creating your WebFacing project you would have been prompted to create a connection to your IBM i server if one didn't already exist. You can use this connection if that is the server you want to deploy your WAR file to. If you want to create a new connection (for example, if the application server is installed on a different IBM i than where your original 5250 application is located), right-click in the Remote Systems view and select New Connection, then choose IBM i.

4. Transfer the file to your server.
   a. Expand the Local icon to explore your local file system and locate the WAR file you exported in the initial step. Right-click the WAR file and select Copy.
   b. Expand the connection icon for your server. For example, if your server is called MYSERVER and you used the same name for the connection name, you should have an icon MYSERVER in the Remote System Explorer perspective. For your server, expand the icon IFS Files and locate a folder that you want to transfer the WAR file to. For example, you might select IFS Files > Root file system > home > my_directory. Right-click the directory you want to transfer the file to can select Paste.

Results

With your WAR or EAR file transferred to your IBM i file system, you can specify it's location directly in the application server's administrative console. For example, if you exported your WebFacing project to a file called wfproj.war and transferred the file to MYSERVER > IFS Files > Root file system > home > my_directory, then in the administrative console application install dialog, you can specify the path for your WAR file as /home/my_directory/wfproj.war.
Creating file shares and mapping network drives

About this task

This document provides some basic information about setting up a file share on the IBM i server and mapping a workstation drive letter to the file share. File shares can be used as a method to transfer files from your workstation to your IBM i. For example, you can use file shares to transfer the WAR or EAR files that are created when you export a WebFacing project from the workbench. Alternatively, files that you want to deploy can be transferred by using Remote System Explorer or FTP.

Generally, these are the steps that are followed:

Procedure

1. NetServer must be installed on your IBM i server. Information about NetServer is available at iSeries NetServer.

2. NetServer is often installed by default on many systems, so you may not require an installation step. With NetServer installed, you will need to start it. NetServer can be started from an IBM i server command line with the command: STRTCPSVR *NETSVR.

3. A file share must be created in the IBM i Integrated File System (IFS) so that a workstation drive letter can be mapped to this later. File shares can be created in the Client Access program Operations Navigator. More information about Client Access and Operations Navigator is available at the i5/OS™ V6R1 Information Center.

   In Operations Navigator, you could create a file share of the /QIBM directory. To do this, in Operations Navigator, drill down to the /QIBM directory in the IFS, right-click /QIBM and select Sharing > New Share.

4. Map a workstation drive letter to the share. For example, if the drive letter Q: is available on your workstation, you could map that drive letter to the /QIBM directory in the IFS. That way, any WAS files that are in an IFS directory path like /QIBM/UserData/WebASAdv/<instance_name>/hosts/default_host can be accessed through the network by your workstation's Q: drive letter with a path like: Q:\UserData\WebASAdv\<instance_name>\hosts\default_host\

   After a share is created, there are several methods by which a drive letter can be mapped to it. A couple of methods are listed below. The NetServer and Client Access documentation can provide more details:

   • Select Search > Printers, computers, or people > A computer on the network from the Windows Start menu and, in the computer name field, enter the name of your IBM i server. When the server is found, double-click on it and supply a valid user name and password to access it. After this, you should be exploring the server's integrated file system which contains the WebSphere Application Server /QIBM directory that you created a file share for. The share is listed by the name you gave for it in the New Share dialog. Right-click on the share and select Map Network Drive. Choose an available drive letter for the share, for example Q:

   • Through Windows Explorer, select the Tools menu and choose Map Network Drive. In the resulting dialog box, choose a drive letter to map to, enter \<IBM i server name>\<folder name> in the path field and then supply a valid user name and password. After you have successfully signed on, choose the file share that you created with Operations Navigator.

   • You can also use the Windows net use command from a Windows command prompt window to map file shares.
Chapter 8. Migration of WebFacing projects

This information explains how to migrate your WebFacing projects from previous versions of the IBM WebFacing Tool. Migration is supported from V6 or later.

Migration is required most often because of a change in the underlying technology from one release to another. For example, new standards in Web application directory structure may require reorganization of your projects from one release to another.

Migrating WebFacing projects

Migration of old projects must be completed before you can continue development of your WebFacing applications. For example, you cannot convert DDS source or change your project's properties. You can run unmigrated projects as is, however. To differentiate between unmigrated and up-to-date projects, unmigrated projects are identified with unique icons in the WebFacing Projects view.

There are two stages of project migration. The first stage is performed by the base product for any type of Web project, and the second is performed by the WebFacing Migration wizard to handle updates specific to WebFacing projects.

Important:

- WebFacing only supports migrating projects at release level V7.0 and later.
- See the [WebFacing Migration technote](http://www-01.ibm.com/support/knowledgecenter/SSRTLE_9.5.0/com.ibm.rad.install.doc/topics/t_update.html) for a list of known issues and resolutions related to migration as well as updates to the migration process that may become known after this document has been released.

Base migration

Begin by making sure the latest updates and fix packs for all Rational products are installed using the IBM Installation Manager. This step is very important as it will eliminate many problems you may run into. See [Updating installed product packages](http://www-01.ibm.com/support/knowledgecenter/SSRTLE_9.5.0/com.ibm.rad.install.doc/topics/t_update.html) and [Updating the HATS Toolkit](http://www-01.ibm.com/support/knowledgecenter/SSXKAY_9.5.0/com.ibm.hats.doc/doc/gsinstall.htm%23update) for details.

WebFacing projects must be loaded into a current workspace before they can be migrated with the WebFacing Migration wizard. Migration is irreversible, so it is recommended that you back up your workspace or projects.

Previous WebFacing projects can be introduced into the current level workspace by opening the old workspace or importing old projects. Existing projects can be imported directly from folders or archive files, including project interchange files (Import > General > Existing Projects into Workspace, then choose Select archive file).

After a project is opened in the current level workspace, the Rational Desktop Migration wizard may appear. The wizard enables you to select which projects to update and shows you which files will be modified. Go through the wizard and make any necessary changes, and click **Finish** to start project migration. When...
workspace migration is complete, results are displayed in the Migration Results view. See Migrating workspaces and projects at http://publib.boulder.ibm.com/infocenter/radhelp/v8r5/index.jsp?topic=/com.ibm.etools.rad.migration.doc/topics/t_migratefrom70.html for more details.

If any errors are reported, or if the migrated project is flagged with errors, refer to Chapter 9, “Troubleshooting WebFacing applications,” on page 107 as well as the online document WebFacing Migration technote which is kept up-to-date.

Note: You can ignore the following dialog if it appears after the Workspace Migration wizard has completed. This error will be resolved later in this document.

- Migration validation completed with some errors. See the Migration Results view for additional information. The Migration Results view highlights errors similar to: The WebSphere facet "WebSphere Web (Extended) 5.1" is not compatible with the WebSphere runtime "WebSphere Application Server vX.Y."

After workspace migration is complete, you can manually reset the perspective by clicking Window > Reset Perspective.

Your projects are now ready for use in the new workspace; however, they still must be migrated using the WebFacing Migration wizard.

Using the WebFacing Migration wizard

Preparing a project for WebFacing migration

You must perform the following preparatory tasks before running the WebFacing Migration wizard.

1. Ensure that a valid server is selected on the Targeted Runtimes page. This is required if the server originally targeted is no longer supported.
   a. In the Navigator view, right-click your project and select Properties.
   b. Open the Targeted Runtimes page and select the server level where the project will be run.
   c. Click OK.

2. If your project is at the Java EE 1.3 level, it must be migrated to 1.4 or higher using the Java EE Specifications Migration wizard before WebFacing migration can proceed. Refer to Migrating the specification level of Java EE projects at http://publib.boulder.ibm.com/infocenter/radhelp/v8r5/index.jsp?topic=/com.ibm.javaee.doc/topics/tmgv6j2eewiz.html.

   Note: If you see a dialog titled Inconsistent Files during Java EE migration, click Yes to dismiss it. Similarly, if at the end of the wizard you see the message Migration finished with errors, click the Details button. The following message can be safely disregarded: Error: org.eclipse.core.runtime.CoreException: Runtime WebSphere Portal v6.0 stub does not support project facet WebFacing 1.0.

3. Resolve any classpath errors in your project.
   a. Open the Problems view (Window > Show View > Other > Problems).
   b. Check if there are any classpath or build errors reported, for example: ClassName cannot be resolved [to a type], The project cannot be built until build path errors are resolved, or Unbound classpath variable.
   c. Resolve any such errors as described in the WebFacing Migration technote.
Running the WebFacing Migration wizard

Next, use the **WebFacing Migration** wizard to update your projects to the current version. You can no longer migrate projects by reconverting DDS files.

1. To launch the **WebFacing Migration** wizard, right-click your unmigrated project in the **WebFacing Projects** view and select **Migrate WebFacing Project**.
2. Click **Finish** to migrate the selected project to the current version.
3. Click **OK** if a **Results** dialog appears notifying you that the Struts version of your project has been changed.
4. If your project was migrated from a version earlier than V7.5.0, it is recommended that you reconvert your DDS, especially if you plan to use the Firefox browser with your WebFacing applications.

**Note**: If the project being migrated is a HATS/WebFacing Enabled project, the corresponding HATS project must also be migrated. See [Migrating to HATS V9.5](#) for details.

Customized projects may require manual intervention. See [“Manual steps for migrating projects with customized files”](#) and [WebFacing support for multiple browsers](#) for further details.

Migrated WebFacing projects require a HATS license to be deployed, just as new WebFacing projects do. If you already have a HATS license applied, your migrated projects will also contain this license setting. Refer to [“Enabling licenses” on page 9](#) for more information.

Uninstalling older product versions

The installation process does not automatically uninstall earlier versions of the software. It is important to note, however that once you have migrated your workspace, you will no longer be able to open it using an older version of WebFacing.

Before uninstalling an older product version, we recommend that you note the location of your workspace, and make a backup copy of it. If you do not back up your workspace, you might lose your projects. By default, these projects are not uninstalled or deleted by the installation program.

Manual steps for migrating projects with customized files

After migration, a number of files that might potentially need customization will have names with the following format: `filename_version_bak`. This has been done so you don't lose your customizations due to migration.

Customized projects may require manual intervention. See [WebFacing support for multiple browsers](#) for further details.

Certain display files in projects created before V7.5.0 might need reconversion after migration to display in Firefox browser, such as DSPF fields using `.CURRENTVALUE` or `.REF` replacement values in their Web settings. It is recommended that a re-conversion be performed on the migrated WebFacing project. To convert a single resource, locate and select the DDS or UIM Help file in the **WebFacing Projects** view. To convert the entire project, locate and select the project in the **WebFacing Projects** view. Right click and select **Convert**.
If you customized your WebFacing project, you might need to merge your customizations from the original files into the following:

- apparea.css
- chrome\stylename.css
- PageBuilder.jsp
- CmdKeys.jsp
- MenuKeys.jsp (not available in all styles)
- logon.jsp
- index.jsp
- rtmessages.properties

If the previous project used Web Site Designer, then the following files must be updated:

- \WebContent\index.jsp
- \WebContent\INV1.jsp
- \WebContent\INV2.jsp
- \WebContent\theme\WFB_blue.jtpl

Note: If the project being migrated is a HATS/WebFacing enabled Web project, then the corresponding HATS project needs to be migrated also.

**Rendering of styles in browsers**

There are some differences in how different browsers render styles. Several changes have been made to the IBM supplied styles so they work in the Firefox browser. If you create a new WebFacing project, the styles will contain those changes. However, if you migrate an existing project created before V7.5.0 that uses an IBM supplied style, the five files that control the style are backed up, if found, and replaced with the new ones. The five files that control the style are:

- stylename.css
- apparea.css
- PageBuilder.jsp
- CmdKeys.jsp
- MenuKeys.jsp (not available in all styles)

You need to compare the new and previous style files and merge any changes you might have made. If you do not use an IBM supplied style, or you have additional style changes, you have to make those changes manually.

To see what changes were made to the new files, compare the old style files with the updated files. The following list shows the changes made:

- All values must have units. In Internet Explorer, values are considered as pixels unless another value is specified. For example:
  ```css
  .launchbutton
  {
    border-color:black;
    border-width:1px;
    text-align:right;
  }
  ```

- In some of the IBM supplied styles, an incorrect syntax was specified for the margins. Internet Explorer tolerated the syntax, but Firefox does not. For example, in stylename.css, margin : 0px,0px,0px,0px; was changed to remove the commas: margin : 0px 0px 0px 0px; For margins where the four margins had the same setting, the change would be margin : 0px;
CSS expressions that were valid only for Internet Explorer (shown in italics in the following example) were removed from the .wf_layer class and replaced with the background color of the body (shown in bold in the following example):

```css
.wf_layer {
    background-image: expression(document.body.currentStyle.backgroundImage);
    background-color: expression(document.body.currentStyle.backgroundColor);
    background-repeat: expression(document.body.currentStyle.backgroundRepeat);
    background-attachment: expression(document.body.currentStyle.backgroundAttachment);
    background-color: #E3C993;
}
```

The `cursor: hand;` notation changed to `cursor: pointer;`

To enable WebFacing applications to display properly when they are rendered using a strict doctype, additional changes were made to the following files:

- **apparea.css**
  
  ```css
  SPAN.wf_field {
    width: 100%;
    border-style: none;
    display: inline-block; //for all browsers
    display: -moz-inline-block; //for Firefox 2
  }
  .scrollbarUpArrow {
    background-color: transparent;
    background-image: url("UpArrow.gif");
    background-position: bottom;
    background-repeat: no-repeat;
    width: 16px;
    height: 16px;
  }
  .scrollbarDownArrow {
    background-color: transparent;
    background-image: url("DownArrow.gif");
    background-position: top;
    background-repeat: no-repeat;
    width: 16px;
    height: 16px;
  }
  .scrollbarSlider {
    background-color: #73a2bd;
    background-image: url("CurrentSegment.gif");
    background-repeat: repeat-y;
  }
  .scrollbarBackground {
    background-color: transparent;
    background-image: url("segment.gif");
    width: 16px;
    height: 100%;
    background-repeat: repeat-y;
  }
```

- **PageBuilder.jsp**
  
  The height for certain styles where page elements should fill the vertical space is added:
  ```html
  <html <wf:lang text="lang="/>
  <head>
    ...
  </head>
  <body style="height:100%">
    ...
  </body>
```

Chapter 8. Migration of WebFacing projects 105
Web Settings migration issues and considerations

In WebFacing projects the input field `.value` is treated as initial value, and is not quoted. This is specified in config\conversion.rules:

```xml
<WebSetting>
  <ValueMigration option="INITIALVALUE"/>
  <INITIALVALUETransform type="unquoted"/>
</WebSetting>
```

Tweaking these settings and reconverting will lead to different results.
Chapter 9. Troubleshooting WebFacing applications

The information in this section can help you troubleshoot problems that occur in WebFacing applications. You may also refer to the following links to learn more.

- WebFacing Troubleshooting Tips
- WebFacing Tool PTFs
- Configuring the WebFacing Server
- Chapter 10, “WebFacing Tool -- current DDS keyword support,” on page 137
- “Considerations for linked HATS/WebFacing applications” on page 84
- “Web Settings” on page 51

For additional information on troubleshooting, as well as helpful tips and updates, refer to WebFacing-specific Support Documents online.

Before contacting IBM Support, please read the Collecting data for IBM i WebFacing applications section.

WebFacing error messages

This section lists error messages that you might encounter when using WebFacing. Refer to the following different times of messages when troubleshooting.

- “Runtime error messages”
- “Error messages by using the INVITE keyword” on page 111
- “Error messages from the inactivity” on page 112
- “Error messages from the panel group” on page 113
- “Error messages from the display file” on page 113
- “Error messages from the logon” on page 116
- “Error messages from the password change” on page 117
- “Error messages from the CL command filter” on page 118
- “Error messages from the system screen support” on page 118
- “Error messages from WebFacing or HATS interoperability” on page 119
- “Error messages from the server” on page 119

Runtime error messages

The messages listed below are runtime messages.

- “WF0002” on page 108
- “WF0003” on page 108
- “WF0004” on page 108
- “WF0005” on page 108
- “WF0006” on page 108
- “WF0007” on page 108
- “WF0008” on page 108
- “WF0009” on page 108
- “WF0010” on page 109
- “WF0011” on page 109
WF0002 • WF0008

• “WF0012” on page 109
• “WF0013” on page 109
• “WF0014” on page 109
• “WF0015” on page 109
• “WF0016” on page 109
• “WF0017” on page 109
• “WF0018” on page 109
• “WF0019” on page 109
• “WF0020” on page 109
• “WF0021” on page 109
• “WF0022” on page 109
• “WF0023” on page 109
• “WF0024” on page 109
• “WF0025” on page 109
• “WF0026” on page 109
• “WF0027” on page 109
• “WF0028” on page 109
• “WF0029” on page 109
• “WF0030” on page 110
• “WF0031” on page 110
• “WF0032” on page 110
• “WF0033” on page 117
• “WF0034” on page 110
• “WF0035” on page 110
• “WF0036” on page 110
• “WF0037” on page 110
• “WF0038” on page 110
• “WF0039” on page 110

WF0002
Explanation: The UPDATE request has been inappropriately entered to the ADBD record header.

WF0003
Explanation: Invalid value has been entered as the request type &1 in the ADBD record header.

WF0004
System action: The specified ScrollbarBean {&1} cannot be found.

WF0005
Explanation:
System action: The specified class {&1} cannot be loaded as a class implementing IScrollbarBean.

WF0006
Explanation: The system attempts to read record &1 in DSPF &2 or &3 before it has been written.

WF0007
User response: Enter a value greater than zero to method allocateOneSideOfPages(), numSegments.

WF0008
The parameter of method allocatePages(), the numPage parameter must be greater than equal to the numSegments parameter.

User response: For the parameter of method allocatePages(), the numPages, enter a value that is greater than or equal to the numSegments parameter.
WF0009
Explanation: For method allocatePages(), value of the currentPage parameter must be less than it of the numPages parameter.
User response: Ensure the value of the currentPage parameter is less than it of the numPages parameter when using method allocatePages().

WF0010
User response: For method allocatePages(), enter a value greater or equal to three for the numSegments parameter.

WF0011
Explanation: Invalid value in DefaultScrollbarBean.getHTMLSource() : &1.

WF0012
User response: Define the transform type in HTMLStringTransform.transform(String,int).

WF0013
Explanation: Invalid request type has been sent.

WF0014
Explanation: Attempt to move backwards in input stream but ADBDOutputBuffer does not support non-sequential data access.

WF0015
Explanation: The specified class or definition file {&1} cannot be found.

WF0016
Explanation: The static initializer displays an exception when loading the specified class {&1}.

WF0017
Explanation: The specified class {&1} might be out of date.
System action: The specified class {&1} is not linked in.

WF0018
Explanation: The specified class {&1} could not be loaded because of internal error.

WF0019
Explanation: The null constructor does not exist for the class {&1}.

WF0020
Explanation: An attempt to load class {&1} causes the exception: &2.

WF0021
System action: The constructor for the class {&1} displays an exception.

WF0022
Explanation: An attempt to call the null constructor of class {&1} has caused the exception: &2.

WF0023
Explanation: The class name for the RecordBean class could not be mapped to a known bean.
System action: System cannot load the RecordBean class.

WF0024
System action: In &1, an exception occurs when the system is obtaining the response of PrintWriter.

WF0025
System action: In ControllerRequestHandler.displayScreen(), an exception occurs when forwarding to /&1/&2/PageBuilder.jsp.

WF0026
Explanation: The invocation name is not specified in URL.

WF0027
Explanation: The system cannot find the invocation properties for invocation &1.

WF0028
System action: Application Request Handler returns an unexpected value.

WF0029
System action: The system is not able to load properties file &1
User response: Check whether the class path for your web application is correct.
WF0030

Explanation: The host name has been specified in neither the invocation properties or the WebFacing application properties.

WF0031

Explanation: The user ID has been specified neither in the invocation properties or the WebFacing application properties.

WF0032

System action: When the system performs the level check, the DDS source converted contains a file-level INDARA keyword but the DSDF object referenced by the application does not.

WF0033

Explanation: The style name has been specified neither in the invocation properties or in the &1 file.

WF0034

System action: The style properties file for the &1 style is not obtained.

WF0035

System action: In HttpRequestHandler.processEndOfApplication(), an exception occurs when the system opens or writes to the response of PrintWriter.

WF0036

Explanation: For record &1 in &2, the system accepts &3 indicators but &4 indicators have been received. For &4, the system accepts &5 indicators.

System action: The system failed to perform the consistency check.

WF0037

Explanation: The output buffer has received number of records in the subfile &2 instead of &1.

User response: Enter a value between 1 and the number of records in the subfile &1 for SFLRCDNBR.

WF0038

Explanation: Subfile I/O buffer length does not match the defined subfile records for record &1.

System action: The system failed the consistency check.
Applications that use the INVITE keyword can cause the hanging problem for the converted WebFacing applications. Many applications using INVITE do not rely on it and it can be safely removed from the DDS source. The unnecessary use of the INVITE keyword will slow down an application that is independent from WebFacing.

Error messages displayed when using the INVITE keyword are listed below.

- **WF0040**
  
  **Explanation:** A requested has been sent to the system for a screen layer that is not contiguous.

- **WF0042**
  
  **Explanation:** Invalid indicator &1 has passed to isOptionIndicatorOn().

- **WF0043**
  
  **System action:** An exception occurs when the system is requesting another subfile page.

- **WF0044**
  
  **Explanation:** The record data bean has passed in for &1 is not a subfile control data bean.

- **WF0045**
  
  **Explanation:** SFLROLVAL is specified, but the field name is empty.

- **WF0046**
  
  **Explanation:** The field value can not be converted to an integer.

- **WF0047**
  
  **Explanation:** SFLROLVAL is specified, but the field name is null.

- **WF0048**
  
  **Explanation:** The record to be read is not specified.

- **WF0049**
  
  **Explanation:** Data type &1 is not valid.

- **WF0050**
  
  **System action:** Error occurs when the system retrieves the user ID and password for authentication.

- **WF0052**
  
  **Explanation:** The record data bean passed in for &1 is not a subfile control data bean.

- **WF0056**
  
  **System action:** When the system connects to host &1 on port &2, error displays : &3.

- **WF0062**
  
  **System action:** The WebFacing server cannot re-establish the session with the Web application.
Error messages from the inactivity

Sometimes, the application ends unexpectedly or the WebFacing session serving the client expires after &1 seconds of inactivity. The maximum interval of inactivity can be configured in the application server.

The inactivity error messages are listed below.

- "WF0063"
- "WF0064"
- "WF0065"
- "WF0066"
- "WF0067"
- "WF0069"
- "WF0070"
- "WF0071"
- "WF0072"
- "WF0073"
- "WF0074" on page 113
- "WF0075" on page 113
- "WF0076" on page 113

WF0063
System action: When the system re-establishes the session with the WebFacing server, the following exception displays: &1.

WF0064
System action: The client downgrades. Communication with the server has been terminated.

WF0065
System action: When the system does version handshake with the WebFacing server, the communication with the server terminates. The following exception displays: &1.

WF0066
Explanation: A serious communication problem occurs between the WebFacing server on the iSeries and the application server.
System action: The system should read &1 bytes from the host but &2 has been received.

WF0067
System action: When a session is created with WebFacing server on host &1:&2, the following exception displays: &3.

WF0069
Explanation: The parameter to &1 is in an incorrect format.

WF0070
Explanation:
System action: Properties file &1 cannot be opened for write.

WF0071 Consistency check failed.
Explanation: For record &1 in DSPF &2, &4 fields have been received instead of &3 fields.
System action: The system failed the consistency check.

WF0072
Explanation: For field &1 in record &2 in DSPF, &3 has been defined with a length of &4 at the time of conversion but data with a length of &5 has been received.
System action: The system failed the consistency check.

WF0073
System action: Data corruption occurs when the system is reading indicators from the output buffer.
System action: An exception occurs when the system tries to obtain the response of PrintWriter.

Explanation: An unexpected exception occurs when system displays help.

Error messages from the panel group

If the panel group &2 containing the help module has not been converted, the following errors might display.

- "WF0077"
- "WF0078"

If the panel group source containing the help module has not been converted, you might receive the following message.

- "WF0122"

System action: An unexpected exception occurs when the system processes the help request.

System action: An exception occurs when the system processes the help information for record &1.

Error messages from the display file

If the display file &2 containing the record has not been converted, the following errors might display.

- "WF0079" on page 114
- "WF0080" on page 114
- "WF0081" on page 114
- "WF0083" on page 114
- "WF0084" on page 114
- "WF0085" on page 114
- "WF0086" on page 114
- "WF0087" on page 114
- "WF0088" on page 114
- "WF0089" on page 114
- "WF0090" on page 114
- "WF0091" on page 114
- "WF0092" on page 114
- "WF0093" on page 115
- "WF0094" on page 115
- "WF0095" on page 115
- "WF0096" on page 115
- "WF0097" on page 115
WF0079

Explanation: A help request on an application with an invalid session has been sent.

---

WF0080

System action: An error occurs but the error page cannot be found.

---

WF0081

System action: A stack trace of the original exception occurs.

---

WF0082

Explanation:

System action:

User response:

---

WF0083

System action: An error occurs when the system is building the command keys.

---

WF0084

System action: The WebFacing internal error occurs when building the screen.

---

WF0085

System action: An error occurs when building the screen.

---

WF0086

System action: The invocation on the iSeries fails.

---

WF0087

Explanation: The CL Command, &1, specified in the WebFacing Tool might be incorrect or the user profile of your WebFacing project might be unable to call the program.

System action: The invocation on the iSeries fails.

---

WF0088

System action: An error occurs when the system is processing a request from the browser.

---

WF0089

System action: A level check exception occurs when the system is writing input buffer back to the WebFacing server.

---

WF0090

System action: An internal exception occurs when building Host Application Response.

---

WF0091

System action: An I/O exception occurs when the system is writing input buffer back to the WebFacing server.

---

WF0092

System action: An unexpected error occurs when the system is processing a request from the browser.
WF0093
System action: An unexpected exception occurs when the system is building Host Application Response.

WF0094
Explanation: You have an invalid session that times out in about &1 minutes.

WF0095
Explanation: WebFacing requires Internet Explorer 5.5 or higher versions.
User response: Download the latest version of Internet Explorer from &1.

WF0096
System action: An error occurs when the system is receiving data from the application.

WF0097
Explanation: The logon to the iSeries fails when the invocation has been initialized.
System action: Error occurs in the communication layer.

WF0098
System action: A runtime error occurs and is sent back to the application instead of resulting in the message &1.

WF0099
Explanation: The support to send errors back to the host application has not been written.

WF0100
Explanation: The record data bean is either missing or inconsistent with the DDS record in the display file.
User response: Re-covert and deploy your DDS.

WF0102
System action: An unexpected exception occurs

WF0103
System action: An error occurs when the system is establishing connection with the WebFacing server on the host.

WF0104
System action: An error occurs when the system is initializing an application.

WF0105
System action: The client version fails negotiating with the WebFacing server.
User response: Upgrade the server.

WF0106
System action: messages.js can not be generated.

WF0112
Explanation: Value must be less than or equal to &1.
User response: Enter a value less than or equal to &1.

WF0113
Explanation: Value must be greater than or equal to &1.
User response: Enter a value greater than or equal to &1.

WF0114
Explanation: The entered value in the filed starts and ends with signs.
User response: Enter valid field value.

WF0115
Explanation: The error occurs because the application does not have write authority to the disk or the disk is full.
System action: WebFacing is unable to open the trace log file for a requested tracing.

WF0116
Explanation: This might be caused by the data transferred from the browser to the application server not being encoded in UTF-8.
System action: The application has received an incorrect encoding check field.
User response: Refer to the WebFacing documentation about how to set up UTF-8 client encoding in WebSphere Application Server.
WF0117 • WF0147

System action: An exception occurs when the system is forwarding request from &1 to &2.

WF0119

System action: Error occurs when the system is refreshing the screen.

Error messages from the logon

The following error messages display when the error happens during logon.

- "WF0130"
- "WFC123"
- "WFC124"
- "WFC125"
- "WFC126"
- "WFC127"
- "WFC128"
- "WFC129"

The following error message displays in the single logon function.

- "WF0147"

WF0130

Explanation: The system fails to load properties for invocation &1.

System action: The CL command is not specified in the invocation properties.

User response: Specify the CL command in the invocation properties.

WFC123

Explanation: Password for user profile &1 is incorrect.

User response: Enter the correct password for user profile &1.

WFC124

Explanation: The user profile &1 is disabled.

User response: Activate the user profile &1.

WFC125

Explanation: The password for user profile &1 has expired.

User response: Change the password for the user profile &1.

WFC126

Explanation: No password associated with the user profile &1.

User response: Create a password for the user profile &1.

WFC127

Explanation: The user profile &1 is incorrect.

User response: Correct the information of the user profile &1.

WFC128

Explanation: The user profile &1 cannot be found.

User response: Correct the information of the user profile &1.

WFC129

Explanation: The user profile &1 cannot be allocated.

User response: Correct the information of the user profile &1.

WF0147

System action: SSO fails because of the incorrect configuration.

116 IBM Host Access Transformation Services: Developing WebFacing Applications
**Error messages from the password change**

The following error messages display when the password is changed improperly.

1. "WF0131"
   - **Explanation:** New password and the verify password are not the same.
   - **User response:** Reenter the new and the verify passwords again and ensure they are the same.

2. "WF0132"
   - **Explanation:** The password is shorter than the system admits.
   - **User response:** Enter a password that matches the system password rules.

3. "WF0133"
   - **Explanation:** The password is longer than the system admits.
   - **User response:** Enter a password that matches the system password rules.

4. "WF0134"
   - **Explanation:** Password contains two numbers next to each other.
   - **User response:** Enter a password that matches the system password rules.

5. "WF0135"
   - **Explanation:** Password contains a character repeated consecutively.
   - **User response:** Enter a password that matches the system password rules.

6. "WF0136"
   - **Explanation:** Password is the same as one of previous passwords.
   - **User response:** Enter a password that matches the system password rules.

7. "WF0137"
   - **Explanation:** Password must contain a number.
   - **User response:** Enter a password that matches the system password rules.

8. "WF0138"
   - **Explanation:** Password contains a character used more than once.
   - **User response:** Enter a password that matches the system password rules.
Error messages from the CL command filter

The following error message is when the CL command filter is not setting correctly.

- "WF0148"

Error messages from the system screen support

The following messages show the errors of system screen support.

- “WF0149” on page 119
- “WF0150” on page 119
- “WF0151” on page 119
- “WF0152” on page 119
- “WF0153” on page 119
- “WF0154” on page 119
- “WF0155” on page 119
- “WF0156” on page 119
**WF0149**

**Explanation:** Unexpected 5250 data stream requests are sent too frequently possibly a loop has been sent.

**WF0150**

**Explanation:** CCSID of &1 might be incorrect. The default value is set as 37.

**WF0151**

**System action:** A 5250 data stream has been encountered but the 5250 transform cannot be loaded.

**WF0152**

**Explanation:** The detected J2EE Level is lower than 1.3.

**User response:** Ensure the J2EE level is 1.3 or higher.

**WF0153**

**System action:** The 5250 data received from the host cannot be displayed on the browser.

**WF0154**

**System action:** Unexpected error occurs when the system is rendering a 5250 data stream into HTML.

**WF0155**

**System action:** Unexpected error occurs when the system is processing a 5250 data stream.

**WF0156**

**Explanation:** The license limit has been reached for dynamic transformation of non-WebFaced 5250 applications.

**System action:** The current screen cannot be displayed on the browser.

**User response:** Purchase additional WebFacing Deployment Tool licenses to support programs running on V5R4 or later releases of i5/OS.

---

**Error messages from WebFacing or HATS interoperability**

Error messages of WebFacing/Hats interoperability are listed below.

- **WF0180**
  - hatsContextRoot in wfhats.xml is not valid.
  - **System action:** Error occurs when the system is trying to forward to HATS for a linked HATS/WebFacing application.
  - **User response:** Fix wfhats.xml and try again.

- **WFC200**
  - **System action:** When the system is starting a session for a linked HATS/WebFacing application, the following exception displays: &1.

**WFC201**

**System action:** The WebFacing server cannot start a session for a linked HATS/WebFacing application.

**User response:** Ensure that the latest level of the WebFacing server is installed on host &1 and is running on V5R4 or later releases of i5/OS.

**WFC202**

**Explanation:** The license limit has been reached.

**System action:** No further connections are allowed.

---

**Error messages from the server**

Errors happen during the connection are listed below.

- **WFC054** on page 120
- **WFC055** on page 120
- **WFC056** on page 120
If the WebFacing server might be down, or the TCP port &2, specified by the web application, is not configured and assigned to WebFacing TCP services on host &1, the following errors display.

- “WFC057”
- “WFC058”
- “WFC059”

If the WebFacing server has ended unexpectedly, the following error messages display.

- “WFC060”
- “WFC061”
- “WFC062”
- “WFC063” on page 121
- “WFC064” on page 121
- “WFC065” on page 121
- “WFC066” on page 121
- “WFC067” on page 121
- “WFC068” on page 121
- “WFC107” on page 121
- “WFC108” on page 121
- “WFC109” on page 121
- “WFC110” on page 121
- “WFC111” on page 121
- “WFC118” on page 121

---

**WFC054**

**System action:** The specified target host &1 cannot be found.

---

**WFC055**

**Explanation:** The specified port &1 is an invalid number.

**User response:** Enter a valid number for the specified port &1.

---

**WFC056**

**System action:** When the system is connecting to host &1 on port &2, the following exception displays: &3.

---

**WFC057**

**System action:** When the system is signing off from the interactive job through the WebFacing server, the following exception displays: &1.

---

**WFC058**

**System action:** When executing the CL command &1 on the host &2 under the user profile &3, the following exception has been received: &4.

---

**WFC059**

**System action:** The system has lost the connection with the WebFacing server on host &1 when waiting for data. The following exception displays: &2.

---

**WFC060**

**System action:** The system failed to logon to host &1 with user profile &2.

---

**WFC061**

**System action:** When the user is logging onto the host &1 with user profile &2, I/O exception: &3 has been received.

---

**WFC062**

**System action:** The application ends unexpectedly or the WebFacing session serving the client has expired after &1 seconds of inactivity. The WebFacing server cannot re-establish the session with the Web application.

**User response:** Configure the maximum interval of inactivity on the application server.
WFC063

System action:  When re-establishing the session with the WebFacing server, the following exception displays: &1.

WFC064

System action:  The client downgrades. Communication with the server has been terminated.

WFC065

System action:  When the system is doing version handshake with the WebFacing server, communication with the server is ended. The following exception displays: &1.

WFC066

Explanation:  A serious communications problem occurs between the WebFacing server on the iSeries and the application server.

System action:  System expects to read &1 bytes from the host but &2 has been received instead.

WFC067

System action:  When the system is creating a session with the WebFacing server on host &1:&2, the following exception displays: &3.

WFC068

System action:  When the system is sending session timeout to the WebFacing server on host &1:&2 the following exception displays: &3.

WFC107

Explanation:  The client has detected a downgraded version of the WebFacing server.

System action:  Communication with the server is terminated.

User response:  Upgrade the WebFacing server.

WFC108

User response:  Apply the latest iSeries PTFs for the WebFacing server.

WFC109

User response:  Apply the latest service pack for the WebFacing Tool and republish the WebFacing web application to the application server.

WFC110

Explanation:  Either the WebFacing Tool on the workstation or the WebFacing server is down level, or the TCP port &2, specified by the web application, is not configured and assigned to WebFacing TCP services on host &3.

WFC111

User response:  Apply the latest WebFacing Tool service pack and iSeries PTFs for WebFacing server support, and republish the WebFacing web application to the application server. Ensure that the TCP port &2, specified by the web application, is configured and assigned to WebFacing TCP services on host &3.

WFC118

System action:  Error occurs when the system is handling the logon request.
Collecting data for WebFacing applications

If a problem with WebFacing is suspected, please first read through the WebFacing troubleshooting section and the links it references.

Collecting the following information will allow IBM Support to better diagnose and resolve problems.

1. Clear description of symptom. Accompanied by screen captures, reproducing instructions, circumvention if any, any other documents.
2. WebFacing project version. From the WebFacing projects view, right-click the WebFacing project, select View Log. The overview tab of the log contains all necessary version information.
3. Run time information, if applicable:
   - Job logs for all QQF jobs: QQFWFSVR, QQFVTSVR, and the QQF# job(s) of interest.
   - Eclipse .log file found in folder .metadata in your workspace.
   - WebSphere Application Server logs.

Related reference:
“Common Base Event logging support for WebFacing applications” on page 132

I cannot run my WebFacing application from a previous release

In this situation, the WebFacing application from an early release does not function. You need to migrate it to the current release.

About this task

WebFacing projects developed with previous versions need to be properly migrated. Follow the instructions in “Migrating WebFacing projects” on page 101.

You may also want to first backup your old projects and workspace before migration.

Results

You must also ensure that licenses for the current release have been activated. Refer to “Enabling licenses” on page 9 for more information.

Application does not start

In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

There might be a problem with one of the following:
- The library list
- The invocation command you specified in the WebFacing project
- The IBM i host setup itself
  - The use of limited capability user IDs
  - WebFacing PTF levels may not be in sync with the WebFacing run-time level
Check the application server console. If you do not see the Pagebuilder.jsp being invoked, this is a good indication that the program on the IBM i host cannot be found.

First, check to determine whether or not the command you specified in the WebFacing project, along with the user ID that you used to log on for the WebFacing application, works in a 5250 environment: Start a 5250 emulation session:

1. Sign on with the same user ID and password that you used in the WebFacing project to run the WebFacing application.
2. Enter the command that you specified in the WebFacing project exactly the same way on the 5250 command line.

If the application does not open in the 5250 environment, something is wrong with the library list or the command:

1. Thoroughly check your application to get it running in the 5250 environment.
2. With the same settings, start your WebFacing application.

If you need to change the user ID and password in the WebFacing project, use the WebFacing project Properties to change them.

If you need to change the invocation command, use the WebFacing Project wizard to remove the current entry and add a new one. To use the WebFacing Project wizard:

- Switch to the WebFacing Projects view.
- Expand the WebFacing project.
- Right-click the **CL commands** node and select **Add**.
- In the Specify CL Commands page, select the CL command in the **CL Command** list and click **Delete**.
- Specify a new command in the **CL command** field and click **Add**.
- Click **Finish**.
- Restart the server and run your application.

If the command in the 5250 environment worked fine and started the application without a problem, check the job log for the WebFacing job. You can do this in the Remote System Explorer under the IBM i jobs node, or you can start a 5250 session and issue a Work Active Job (WRKACTJOB) command.

Look for a QQF job started with the user ID you are using and then check the job log. If you cannot find a QQF job, the job might have ended. Use the Work Spool File (WRKSPFLF) command for the user ID you used in the WebFacing application to determine whether or not there are any job logs from the terminated job that could give you more information.

**Checking the WebFacing PTF level**

Make sure you have the latest PTF level on your IBM i host. Refer to the WebFacing Tool PTFs for more information. If you have a mismatch, end the WebFacing server, load and apply the PTFs, then restart the WebFacing server:

1. End the WebFacing server (ENDTCPSVR *WEBFACING)
2. Load and apply the desired PTFs
3. Start the WebFacing server (STRTCPSVR *WEBFACING)
Run the WebFacing application again.

**Related topics**

“Problem with using limited-capability user IDs”
A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

“WebFacing server does not start” on page 125
If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

“Page cannot be displayed” on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.

“Page cannot be found” on page 127
When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

“Invalid session” on page 129
The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

“DDS and WebFacing are out of sync” on page 129
If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconver this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.

---

**Problem with using limited-capability user IDs**

A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

For versions of the IBM i operating system before V6R1, you can circumvent the problem. See [Using a LMTCPB userid with WebFacing](#) for more information.

**Related topics**

“Application does not start” on page 122
In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

“WebFacing server does not start” on page 125
If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

“Page cannot be displayed” on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.
When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

"Invalid session" on page 129
The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

"DDS and WebFacing are out of sync" on page 129
If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.

**WebFacing server does not start**

If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

Use the Start TCP/IP server command to start the WebFacing server.

```bash
STRTCPSVR *WEBFACING
```

**Note:** Make sure that the IBM i system value QAUTVRT is higher than 0. Because the WebFacing server relies on Virtual Terminal jobs, you should check to make sure that this system value is set so that all WebFacing sessions are supported.

You will receive the same message if the IBM i server you have specified to connect to does not exist, so use the WebFacing project properties to ensure that the server name is correct.

Scroll down the rest of the error page to see which IBM i host the application attempted to access and which port number was used to connect to the WebFacing server. Check that these values are correct. The default port for the WebFacing server is 4004. Use the following command to check the ports:

```bash
NETSTAT *CNN
```

Press F14 to show the port numbers, and use option 8 to check the jobs using port 4004.

There should be one job listed (QQFWFSVR). That means the WebFacing server is using port 4004. If the job listed is not QQFWFSVR, then it is not the WebFacing server listening on port 4004. If this is the case:

- Check the port the WebFacing server is using by entering the Work with Service Table (WRKSRTBLE) command.
- Look for an "as-WebFacing" service entry in the list. If there is an entry, use the display option to see what port is being used.

**Note:** If the default port 4004 is not in use by some other service, there is no need to change the port number to something else.

If there are QQF jobs:
• Check the job logs in both of them.
• In case of any errors, check the PTF level. If your system has the current WebFacing PTFs applied, this is most likely a problem and should be reported to IBM support.

**Determining whether or not the WebFacing server responded to a request**

You can easily check to determine whether or not the WebFacing server on the IBM i host has been responding to a request by checking the idle time in the **Work with TCP Connections Status** list.

If the idle time is close to zero and you are the only user running WebFacing applications, then your last request has been received by the WebFacing server and the job logs should help you find the problem.

If the idle time is high, then the request did not make it to the WebFacing server. Check the IBM i server name you specified in the WebFacing project, and check the authentication. You can try removing the automatic authentication in the WebFacing project properties to force a signon dialog. If the signon dialog appears, you know that you reached the IBM i server and you can check the WebFacing server environment as described before.

**Related topics**

“Application does not start” on page 122
In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

“Problem with using limited-capability user IDs” on page 124
A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

“Page cannot be displayed” on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.

“Page cannot be found” on page 127
When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

“Invalid session” on page 129
The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

“DDS and WebFacing are out of sync” on page 129
If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.
Page cannot be displayed

This error is most likely caused by an application server problem, or an incorrectly entered URL.

When you receive this error:
- Check whether the application server is started.
- If the application server is not started, start it.
- If the application server is already started, restart it.

The application server might be in a problem state, so restarting it will reset it. Close the browser and then try to run the application again.

Related topics
- “Application does not start” on page 122
  In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.
- “Problem with using limited-capability user IDs” on page 124
  A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.
- “WebFacing server does not start” on page 125
  If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.
- “Page cannot be found”
  When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.
- “Invalid session” on page 129
  The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.
- “DDS and WebFacing are out of sync” on page 129
  If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.

Page cannot be found

When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

Make sure that the index.jsp page exists in your WebFacing project:
1. In the Navigator view, look in the WebContent folder of your project for the index.jsp file.
2. If the file is not there, rebuild the WebFacing project.
Check the web.xml file, which contains the project information. If it is not created correctly, this error may occur.

1. In the Navigator view, expand WebContent > WEB-INF and double-click the web.xml file.
2. Check that there are WebFacing servlets in the Servlets and JSPs list.
3. If there are no servlets, then the project is corrupted. Try rebuilding the WebFacing project.

If the index.jsp file exists and the web.xml file looks correct, then the application might not be installed correctly.

In the test environment:
1. Stop the application server.
2. Publish the server.
3. Make sure you receive a success message.
4. Start the server again.
5. Try to run the application.

If you are working with a remote application server on an IBM i host or any other platform outside your workstation:
1. Make sure the application is installed correctly and use the application server console to verify that everything is installed correctly.
2. Make sure the application is started.
3. Restart the server to make sure it picks up the application information.
4. Restart the application server.

**Related topics**

"Application does not start" on page 122
In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

"Problem with using limited-capability user IDs” on page 124
A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

"WebFacing server does not start” on page 125
If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

"Page cannot be displayed” on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.

"Invalid session” on page 129
The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

"DDS and WebFacing are out of sync” on page 129
If you change the display file DDS source member and recompile your program
to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.

Invalid session

The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

In this environment, the server cannot detect whether a session has been interrupted permanently by problems with communications or by WebSphere Application Server itself. To avoid having too many invalid sessions, the application server has a default session time-out of 30 minutes. By default, a session will be invalidated after 30 minutes of idle time. If you try to use the session after 30 minutes of inactivity, you will see this error.

If this time interval of 30 minutes does not suit your environment, you can change the application server settings using the Administration console to change the time-out values.

Related topics

"Application does not start" on page 122
In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

"Problem with using limited-capability user IDs" on page 124
A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

"WebFacing server does not start" on page 125
If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

"Page cannot be displayed" on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.

"Page cannot be found" on page 127
When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

"DDS and WebFacing are out of sync"
If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.

DDS and WebFacing are out of sync

If you change the display file DDS source member and recompile your program to work with the changed display file, you will also have to reconvert this DDS source member to create the corresponding JSPs and XML files in the WebFacing project.
If you do not do this, you may receive an application error (with error messages WF0096 and WF0100) when the changed screen would have been displayed. When a screen is submitted and there has been a change in the input buffer definition you may see a MCH1236 error from/to QWSGET in the user job. To fix either of these problems you will need to ensure that the version of DDS source that the display file object was created with is the same as the version of DDS source that was converted for the WebFacing application. If the green screen version of the application is working fine then simply reconvert the DDS members that have changed.

This problem can also occur if more than one display file object with the same name is used. In this case, the DDS Object Mappings must be manually updated to explicitly define the library in the mapping for the multiple display files, rather than using the default display file name mapping.

Sometimes this error will also occur if the publishing failed to finish successfully. To fix this, stop the application server, republish the project, and start the application server again.

Related topics

"Application does not start" on page 122
In this situation, the application does not start when you click the link on the index.jsp page. The page remains in the browser with no error message and the application appears to hang.

"Problem with using limited-capability user IDs" on page 124
A limited-capability user ID on IBM i is one that has the LIMIT CAPABILITIES (LMTCPB) parameter set to *YES or *PARTIAL. WebFacing only supports limited-capability users on IBM i V6R1 and later.

"WebFacing server does not start" on page 125
If you receive an application error after clicking the hyperlink to invoke the WebFacing application, check the error message just below it. The error message contains details about the cause of the error. One of the more frequent messages is that the WebFacing server is not started.

"Page cannot be displayed" on page 127
This error is most likely caused by an application server problem, or an incorrectly entered URL.

"Page cannot be found" on page 127
When this error displays, the WebFacing project might be corrupted, the WebFacing project has not been installed correctly, or the URL is being used incorrectly.

"Invalid session" on page 129
The Web environment is not based on persistent connections between the browser and the HTTP server. When you run the WebFacing application and leave the browser for an extended period of time, the application will eventually time out.

---

**WAS Test Environment fails initialization for WebFacing application if long workspace path is used**

You may experience a problem with WebFacing projects not running in the WebSphere Test Environment with an error in the console similar to: 6cb33777 SystemOut O Exception in generateFieldAttributes : java.io.FileNotFoundException: (The system cannot find the path specified)
Limitation of the server related to handling long directory names. If you use a workspace in a directory with a long path or choose long names for your projects, you may get this error when starting a server, or when testing files on a server.

You may take one of the following actions:

- Move your workspace to a location with a shorter path, for example C:/workspace.
- Give your Enterprise Application project or other project a shorter name.

**Context root with space does not run on WAS Test Environment**

A space is not allowed in context roots because it does not conform to Java EE specifications.

**Solution**

If you want to use a special character or a space character in your project name, you must ensure that the context root of your project does not contain a special character or space. To do this, select the navigator view in the WebFacing perspective, right click on the project, select properties, and edit the context root in the Web Project Settings page. This is required if you want to run in the WebSphere Test Environments.

**Application does not work properly with RSTDSP(*NO) specified**

WebFacing applications do not work properly when display files have RSTDSP(*NO) specified.

RSTDSP is a parameter on the Create Display File (CRTDSPF) or Change Display File (CHGDSPF) command. It specifies whether data being shown at a display device by this display file is saved at the time the file is suspended (made temporarily inactive) so that a different display file can be used to show different data on the same device. If the data for this file is saved, it is restored to the display of the device when the file is used again. When *YES is specified for the Restore Display (RSTDSP) parameter, an image of the current display is saved when the display is suspended. When the display file is activated again, the saved image is used to restore the display to its appearance before being suspended.

According to Application Display Programming, the RSTDSP(*YES) parameter must be specified for the following keywords. If the parameter is not specified, data on the display can be lost if the file is suspended. You must ensure that the records that are on the display are the records that these keywords apply to. If the display file is suspended, the data must be restored to the screen so that the write operations to the record formats that use these keywords are valid.

- CLRL
- OVERLAY
- PUTOVR
- PUTRETAI
- ERRMSG
- ERRMSGID

WebFacing does not have the previous screen state for the display file to restore to if the display file has RSTDSP(*NO) specified. This would result in an IndexOutOfBoundsException in some cases. In other cases, the application may not
be broken, however you may see displays different from 5250; for example, some
windows overlap each other in WebFacing.

If you experience these problems, please make sure RSTDSP(*YES) is specified for
the display files that are used in the application.

Absolutely positioned elements appear in unexpected locations in
WebFacing

Elements previously customized with an absolute position property show up in
unexpected locations after migration.

WebFacing includes support for positioning DDS elements using Cascading Style
Sheet Positioning (CSS-P) technology. With this support, the WebFacing application
area becomes a "containing block". In accordance with the CSS specification,
absolutely positioned elements are positioned relative to this containing block.

Take one of the following actions to place the elements in their desired positions:
• Use the WebFacing support for Cascading Style Sheet - Positioning instead of
  your own absolute positioning. (Recommended)
• Use relative positioning (instead of absolute positioning) since relative
  positioning is always relative to the field's original position.
• Adjust the absolute positioning coordinates, keeping in mind that the elements
  are now positioned relative to the containing block of the application area.
• Remove the two occurrences of style="position:relative" from
  ScreenBuilder.jsp in \WebContent\webfacing\jsp\common\html for your
  WebFacing project if you do not want to change your absolute positioning
  coordinates and do not want to use the WebFacing support for Cascading Style
  Sheet - Positioning.

Common Base Event logging support for WebFacing applications

Running in WebSphere Application Server

WebFacing applications use WebSphere Application Server Common Base Event
logging to diagnose runtime application problems. When you run a WebFacing
application on an application server, logging information is sent by default to the
console of the application server. For more information about logging and tracing
in WebSphere Application Server, see the information center for your level of
WebSphere Application Server, for example, http://publib.boulder.ibm.com/
infocenter/wasinfo/v8r0/index.jsp?topic=/
com.ibm.websphere.nd.multiplatform.doc/info/ae/ae/ttrb_addtrace.html

You can configure the logging level of the WebFacing application logger
(com.ibm.etools.iseries.webfacing.context_root_name) using the administrative
console in WebSphere Application Server. For information about configuring the
logger, see the Information Center for your level of WebSphere Application Server
and search for Configuring logging properties using the administrative
console.

You can also configure the logger details level as well as other settings for the
WebFacing application logger by using the webfacinglogging.properties
configuration file.
By default, the log message in WebSphere Application Server is written to the
console of the application server, and the log record is written to the WebSphere
Application Server activity.log file. If you turn on tracing through the
application server administrative console, the trace information is written to the
trace.log file of the application server instance. When tracing is enabled, all log
messages are logged to trace.log in simple text format, and all log records with
log level CONFIG and above, are also logged to activity.log in Common Base
Event format. For information about turning on tracing in WebSphere Application
Server, see the information center for your level of WebSphere Application Server
and search for Enabling trace at server startup.

You can also configure your settings for WebFacing applications to separate the
runtime logging and tracing information into different files in the system where
your application server is running. Log records, where the log level is above or
equal to CONFIG, are written to a log file and a trace file, if tracing is enabled for
this logger. Trace records, where the log level is below CONFIG, are written to a
trace file. See the "Configuration settings for a WebFacing application" section
below for the properties that enable this feature.

The log and trace files are organized by each HTTP session. The name of the log
file is sessionid.log, where sessionid is the session id of the browser accessing
the WebFacing application. Similarly, the trace file name is sessionid.trace. The
log file related to a WebFacing application is written into a directory with the same
name as the context root name of your Web project containing the WebFacing
application. See the logFileLocation property in the "Configuration settings for a
WebFacing application" section below for information about configuring the
location of your WebFacing log and trace files.

See the following topics for information about using the Agent Controller for IBM
i:

- Administering Agent Controller
- Publishing, running, and debugging applications using IBM Agent Controller

If you just want to look at the message text of your log record, you can configure
your settings to format your log record to be written as a simple string. See the
useXMLFormatForCBELogging property in the "Configuration settings for a
WebFacing application" section below for information about controlling the format
of your Common Base Event log record.

**Configuration settings for a WebFacing application**

You can configure the logging and tracing settings of a WebFacing application
through the webfacinglogging.properties file, which is located in the
WebContent/WEB-INF/classes for a WebFacing Web project. Here you see the
settings inside this file.

```
# Control the level of runtime logging
# The levels in descending order are:
# SEVERE (highest value)
# WARNING
# INFO
# CONFIG
# FINE
# FINER
# FINEST (lowest value)
# In addition, there is a level OFF that can be used to turn off logging,
# and a level ALL that can be used to enable logging/tracing of all messages.
# See the Java document for the java.util.logging class for more information.
```
# The levels above or equal to CONFIG are considered logging.
# The levels below CONFIG are considered tracing.
# Default is SEVERE.
com.ibm.etools.iseries.webfacing.level=SEVERE

# Control if the logging level set in this file will override
# the static configuration settings of log detail levels in WAS
# Valid values are:
#   true - default
#   false
com.ibm.etools.iseries.webfacing.overrideAdminLevelSetting=true

# Control which directory the log and trace files will be written to.
# A subdirectory with the same name as the web-app will be created
# to store all the log and trace files of a specific web application.
# This value will be used when property "logToFile" is set to true.
# "%WASTraceDir" - trace directory of the WAS server instance.
#   By default, it is the SERVER_LOG_ROOT variable defined.
#   in the WAS server. This value will be used as default value
#   if the web application is running in a WAS environment.
#   If this value is not available, "%h/iserieslogs" will be used as
#   default.
# "%h" - the value of the "user.home" system property
#   "%h/iserieslogs" will be used as default value
# "%st" - the value of the system temporary directory
# "." - the value of the WEB-INF directory
# "/" the local pathname separator
#
# Example for workstation full path:
# "c:\my_directory\my_log_dir"
# Please notice that double-backslash is needed here.
#
# Example for iSeries IFS full path
# "/home/my_id/my_log_dir"
com.ibm.etools.iseries.webfacing.logFileLocation=%WASTraceDir

# Control if log and trace information will be written to files.
# Valid values are:
#   true
#   false - default
com.ibm.etools.iseries.webfacing.logToFile=false

# Control if log and trace information will be written to console.
# Valid values are:
#   true
#   false - default
com.ibm.etools.iseries.webfacing.logToConsole=false

# Control if parent's output handler of this logger will also be used.
# If this property is set to false, the log or tracing information will
# not be written to the WAS activity.log or trace.log
# Valid values are:
#   true - default
#   false
com.ibm.etools.iseries.webfacing.useParentHandler=true

# Control the format of the Java log records written to a file.
# If this property is set to true, the log records written to the
# log file will be in XML format.
# See the Java documentation for the java.util.logging.XMLFormatter class
# for more information.
# If this property is set to false, the log records written to the
# log file will be in simple string format.
# Valid values are:
#   true - default
#   false
com.ibm.etools.iseries.webfacing.useXMLFormatForJSR47Logging=true
# Control the format in which Common Base Event log records are written to a file.
# If this property is set to true, the log records written to the
# log file will be in Common Base Event
# XML format.
# If this property is set to false, the log records written to the log
# file will be in simple string format.
# Valid values are:
# true - default
# false
com.ibm.etools.iseries.webfacing.useXMLFormatForCBELogging=true

# Control how many log files will be created for each web application
# before the oldest log file is deleted.
com.ibm.etools.iseries.maxlog=3

Note: If the application server is running on IBM i, you may be required to assign
permission to this directory so that the log files can be created in the IFS. For IBM
WebSphere Application Server, you need to give write authority to the QEJBSVR
user profile. Specifically the QEJBSVR profile should have read, write, and execute
authority to the trace directory in the table above.

**TFRJOB or RRTJOB with WebFacing ends user job**

If a user job is being run in the WebFacing environment, do not use the TFRJOB or
RRTJOB commands and do not apply RTGDTA.

The use of the TFRJOB or RRTJOB commands or the RTGDTA parameter for other
than the default QCMDI value is not compatible with WebFacing. If a TFRJOB or
RRTJOB command is performed on a user job while running in the WebFacing
environment, it causes the user job to end.

When launched in the WebFacing environment, there is control data associated
with the user program. However, as part of the routing action, any objects
allocated in the previous routing step are deallocated and any open files are closed.
When this is done to a WebFacing user job, that job is orphaned from WebFacing,
and then ended by the IBM i operating system.

**Cannot access extended help on dynamically rendered WebFacing screens**

By default, extended help cannot be accessed on WebFacing screens that are
rendered dynamically because cursor positioning is not turned on for protected
fields.

To enable this function, customize the transformation so that the cursor can be
positioned to protected fields such as help menu options.

1. From the Navigator view, open the renderingSet.xml file, under
   `<project_name>\src`.
2. Search the file for the `showLinksForProtectedFields` setting. It should be near
   the end of the file.
3. Change the setting’s value from `false` to `true`. This allows the cursor
   positioning on protected fields by rendering them as links. Clicking on one of
   these links sets the cursor position to that specific location.
4. Save the file.
5. Republish and run your project.
You should now be able to display extended help by clicking on an extended help option, and then pressing Enter.
Chapter 10. WebFacing Tool -- current DDS keyword support

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application.

<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALIAS *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTHELP *</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTNAME *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTPAGEDWN *</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEUP *</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALWGUI</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWROL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (don't clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLANKS *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>BLKFOLD *</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>CAnn/CFnn *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHANGE *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCavail</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB) *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(RAZ) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGOSDFEU *(FE)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGOSDFEU *(ME MF LC CS)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGOSDFEU *(RI HI)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGOSDFEU *(UL/BLI)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CLEAR</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLRL(mm</td>
<td>END)</td>
<td>Yes</td>
</tr>
<tr>
<td>CLRL(*ALL)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CMP</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CNTFLD</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>COLOR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>COMP</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CSRINPONLY</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>DATESEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>DFT (implicit)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DLTCHK</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTEDT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DSPATR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(MDT)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(OID)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PC)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PR CS)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>“DSPATR(SP)” on page 184</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(UL HI RI ND)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (unconditioned)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (with option indicators)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPRL</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DSPSIZ</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>“EDTCD(5-9)” on page 193</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTCD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTMSK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTWRD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>ERASE</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERRMSGID</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSC</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>FLDCSRPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLTFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>FLTPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>FRCDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GETRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDCLR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLIN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>“HELP (2nd lev msg)” on page 196</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (return to PGM)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (with HLPARA)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPARA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPBDY</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCLR</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDOC</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDDL</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPID</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRCD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRIN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPSEQ</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSELD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSTELF</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HOME*</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>IGCAUTYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCCNV</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INDARA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INDTXT*</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INVITE (multiple devices)*</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (single-device)*</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOGINP *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MDTOFF *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MLTCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNURAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCNL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MOUBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVERLAY *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRATR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PAGEDOWN/PAGEUP*</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PASSRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PRINT(*PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PSHTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PUTOVR *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>No</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Numeric) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>REFFLD *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REF *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>RETKEY/RETCMDKEY*</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RETLCKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>ROLLUP/ROLDOWN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RTNCSRLOC(*MOUSE)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSRLOC(*RENAME</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SFLCHCCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDLT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLEND</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLTCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXTCHG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGMQ/SFLMSGKEY/SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRSA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROLVAL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNGCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
</tbody>
</table>

Chapter 10. WebFacing Tool -- current DDS keyword support  141
<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TEXT*</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIMfmt</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMsep</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>UNLock</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>USRDFN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSpmgt</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRStDSF</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>VALnum*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>WDBorder</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDWtitle*</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(definition)</td>
<td>Yes (partial)</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(reference)</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WrdWrap*</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(29) Reference*</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(30-34) Length</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type/Keyboard shift*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(39-44) Location*</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
</tbody>
</table>

Related concepts:

Chapter 1, “Introducing the WebFacing Tool,” on page 1

Related tasks:

Chapter 3, “Creating a WebFacing Web project,” on page 11

“Opening the WebFacing perspective” on page 12

“Selecting the source members to convert” on page 13

“Specifying CL commands for invoking your application” on page 14

“Choosing a Web style” on page 12

“Finishing the project and converting your DDS source” on page 15

“Analyzing the conversion logs” on page 15
WebFacing Tool -- current DDS keyword support

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application. The DDS keyword survey tool can be downloaded from [IBM Support](https://www.ibm.com/support).

<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHGINPDFIT(FL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFIT(RI HI)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFIT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHGINPDFIT(UL/BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>COLOR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DATESEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(program-to-system-field)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(MDT)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(OID)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PC)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PR CS)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(UL HI RI ND)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTCDE(5-9)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTCDE</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTCDE</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTPRWD</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>FLTXFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>HLPDATA</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>HLPDATA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSCHIDX</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSCHIDX</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSEQ</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>HLPSHELF</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>ALIAS</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTNAME</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLITCHK</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTEDT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REFFLD</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REF</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TEXT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(29) Reference</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRNONLY</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLDCSRPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSRLC(&quot;MOUSE&quot;)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSRLC(&quot;RECNAME&quot;)</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSRLC(&quot;WINDOW&quot;)</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>BLKFOLD</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>(39-44) Location</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>DFT (implicit)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(SP)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>ALTHELP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEDWN</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEUP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CAHn/CFnm</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLEAR</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (2nd lev msg)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (return to PGM)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>HELP (with HLPARA)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPRTN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HOME</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MOUBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PAGEDOWN/PAGEUP</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT(PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RETKEY/RETCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ROLLUP/ROLLDOWN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSLT</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CNTFLD</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDCLR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLIN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MLTCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEF</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCND</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>NDARA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>NDTXT</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>BLANKS</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHANGE</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>FLIPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>LOGINP *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>ERRMSGID *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSC *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DSPRL *</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCALTYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCCNV *</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWGPH</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWROL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (don’t clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(nn</td>
<td>*END) *</td>
<td>Yes</td>
</tr>
<tr>
<td>CLRL(*ALL) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (unconditioned) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (with option indicators)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPSIZ *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERASE *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>FRCDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GETRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (multiple devices) *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (single-device) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>MDTOFF *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVERLAY *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRATR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PASSRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTOVR</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAI</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RTNLCKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>UNLOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDFN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSPMGT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SFLCHCCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDLT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLEND *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLTCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXCHC</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGMO/SFLMSGKEY/SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRNA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROLVAL *</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNGCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFL</td>
<td>*</td>
<td>Subfiles</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(RAZ)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDF(ME MF LC CS)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CMP</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>COMP</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTMSK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Numeric)</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALNUM</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>(35) Data type/Keyboard shift</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>USRRSTDSP</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDWBORDER</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDWTITLE</td>
<td>*</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(definition)</td>
<td>Yes (partial)</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(reference)</td>
<td>Yes</td>
<td>Window</td>
</tr>
</tbody>
</table>

**WebFacing Tool -- current DDS keyword support**

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** Note: You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application. The DDS keyword survey tool can be downloaded from [IBM Support](http://www.ibm.com/support).
<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASSUME (don’t clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLANKS</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CAnn/CFnn</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CHANGE</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(RAZ)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(FE)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT([FE])</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(ME MF LC CS)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDFT(RI HI)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHGINPDFT((UL/BL))</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CLEAR</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLRL(nn</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*END)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*ALL)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CMP</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CNTFLD</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>COLOR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>COMP</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CSRINONLY</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>DATESEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DFT (implicit)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DLTCHK</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTEDT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DSPATR(BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(MDT)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PC)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PR CS)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(SP)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(UL HI RI ND)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>DSPATR [program-to-system-field]</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (unconditioned) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPSIZ *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>EDTCDE</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTMSK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTWRD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASE</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERRMSGID *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSG *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL *</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>FRCDTIA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>HELP (return to PGM) *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (with HLPARA) *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPARA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPBDY</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCODE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPEXCLD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPID</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRL</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HOME *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>IGCALRTYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INDARAA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INDTXT *</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INVITE (single-device) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOGINP *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>MDTOFF *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>OVERLAY</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PAGEDOWN/PAGEUP *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT(*PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Numeric) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>REFFLD *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REF *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>RETKEY/RETCMDKEY *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>ROLLUP/ROLLDOWN *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RTNCSRLOC(*RENAME</td>
<td><em>WINDOW)</em></td>
<td>Yes</td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFDLDT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLEND *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXTCHG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGMQ/SFLMSGKEY/SFLMSGRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRNA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFL *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TEXT *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIMFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMSEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>VALNUM*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>WDWTTITLE*</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(definition)</td>
<td>Yes (partial)</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(reference)</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WRDWRAP*</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(29) Reference*</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(30-34) Length</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type/Keyboard shift*</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(39-44) Location*</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWGPHT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWROL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>BLKFOLD*</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DSPATR(OID)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (with option indicators)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPRL*</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>EDTCDE(S-9)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>FLDCSRPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLTTFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>FLTPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>GETRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDCLR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLIN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>HELP (2nd lev msg)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDOC</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSCIDX</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSHELF</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>IGCCNV</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INVITE (multiple devices)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MLTCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCNL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MOUBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRATR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PASSRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PSHBTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PUTOVR</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>No</td>
<td>Validation</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>RETLCKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RTNCSSRLOC(*MOUSE)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>SFLCHCCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLTCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROVAL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNGCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>UNLOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDFN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSPMGT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRRSTDSP</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDBORDER</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
</tbody>
</table>

**WebFacing Tool -- current DDS keyword support**

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** Note: You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application. The DDS keyword survey tool can be downloaded from [IBM Support](https://www.ibm.com/support).
<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDTWRD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>FLTFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>TIMSEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>WRDWRAP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(30-34) Length</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>HLPARA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPBDY</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCLRR</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDOC</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPEXCLD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPID</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPNNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSCIDX</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSEQ</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSEH</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>ALIAS</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTNME</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTC</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REFLD</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REFLD</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TEXT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(29) Reference</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRINPONLY</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLDCSRCPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSLOC(*MOUSE)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSLOC(*RENAME</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSLOC(*WINDOW)</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>DLKFLD</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>(39-44) Location</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>DFT (implicit)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(SP)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>ALTHELP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTUPEDOWN</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEUP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CAnn/CScr</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLEAR</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (2nd lev msg)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (return to PGM)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (with HLPARA)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPRTN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HOME</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MOBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PAGEDOWN/PAGEUP</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT('PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RETKEY/RETCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ROLLUP/ROLLDOWN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSLT</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CNTFLD</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLIN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>MLTCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCNL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>INDARA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INDTXT</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>BLANKS</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHANGE</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>FLTPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGINP</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>ERRMSGID</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSC</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>OSPRL</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCALTTYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCCNV</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWGPH</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWROL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td><strong>Sort by Keyword</strong></td>
<td><strong>Sort by current support</strong></td>
<td><strong>Sort by Category</strong></td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-----------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ASSUME (don’t clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(nn</td>
<td>*END) *</td>
<td>Yes</td>
</tr>
<tr>
<td>CLRL(*ALL) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (unconditioned) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (with option indicators)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPSIZ *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERASE *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>FRCDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (multiple devices) *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (single-device) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MTOFF *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVERLAY *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRAFTR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PASSRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTOVR</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RETLCKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>UNLOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDFTN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSPMG1</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SFLCHCCCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDLT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>SFLEND</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENCENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLTCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCRD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXTCHC</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAGE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGMQ/SFLMSGKEY/</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCRD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRNA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROLVAL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNCCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SF *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(ARZ)</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK *</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDFT(ME MF LC CS)</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>COMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTMKS</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Numeric)</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALNUM *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>(35) Data type/Keyboard</td>
<td>*</td>
<td>Validation</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>USRRSTDSP</td>
<td>No</td>
<td>Window</td>
</tr>
</tbody>
</table>
**ALARM**

Category
Screen Building

Additional information

**ALIAS**

Category
Compile-time values

Additional information
This keyword is supported by default. When a program is compiled, the alternative name is brought into the program instead of the DDS field name. This information is then resolved using the compiled DDS object before a request is made to any converted record formats. The converted record formats use the original DDS field names.

**BLANKS**

Category
I/O Buffer

Additional information
The same response indicator should not be used with a BLANKS keyword and another keyword, since the DDS manual warns against this and the results of doing so may be unpredicatable.

A BLANKS keyword on a field that has both DSPATR(PR) and DSPATR(ND) on a field that has been hidden using Web settings may not work as expected in Webfacing. In Webfacing, the BLANKS response indicators for such fields will not be turned on even when the fields are blank; they would be turned on for a 5250 application.

When BLANKS is specified on a field on a subfile record and the field is written to the screen as blank, but the no fields on subfile record are changed, the BLANKS response indicator will not be turned on. On the 5250, the response indicator would be turned on. This is only a problem if (1) BLANKS is specified on a subfile field and (2) no fields in the subfile record are changed.

**IGCCNV**

Category
NLS Enablement

Additional information
Because this keyword provides functionality that is duplicated by the PC's language platform, it is not necessary to implement conversion for this
keyword. The end-user can use the Windows system for creating DBCS characters rather than an emulation of the 5250 method for doing the same.

### INDARA

<table>
<thead>
<tr>
<th>Category</th>
<th>Indicators</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additional information</strong></td>
<td></td>
</tr>
</tbody>
</table>

### INDTXT

<table>
<thead>
<tr>
<th>Category</th>
<th>Indicators</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additional information</strong></td>
<td></td>
</tr>
</tbody>
</table>

The INDTXT keyword is ignored by conversion. This text has no function in the file or the program other than as a comment.

The related INDTXT information will still be available in the display file when using commands such as DSPFFD on IBM i. However, these comments will not be available after conversion in the JSPs or XML files.

### INVITE (multiple devices)

<table>
<thead>
<tr>
<th>Category</th>
<th>Screen Building</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additional information</strong></td>
<td></td>
</tr>
</tbody>
</table>

INVITE is supported for single display devices. Scenarios where the application invites multiple devices concurrently are not supported.

### INVITE (single-device)

<table>
<thead>
<tr>
<th>Category</th>
<th>Screen Building</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additional information</strong></td>
<td></td>
</tr>
</tbody>
</table>

The INVITE operation is used to send a request for input to a display device and return to the program without waiting for input from the user. Before you run WebFacing applications that use the INVITE keyword, ensure that you have installed IBM i PTFs for WebFacing INVITE support. You may not need to use the INVITE keyword if your applications do not use asynchronous I/O. In this case, you can remove the INVITE keyword.

The WebFacing Tool supports the INVITE operation differently from 5250 devices. The main difference is that the READ operation timer is handled by the browser. This difference is due to the fact that an application can overwrite the screen records displayed on a 5250 display device, but cannot overwrite the page displayed in a browser without the browser giving control back to the application. Implementing the timer in the browser allows the browser to hand control back to the application when the timer expires.

The WebFacing Tool, by default, uses the value of the WAITRCD parameter for the browser-side timer. You can, however, override this value in the WebFacing project properties. The following describes the timer
operation as implemented by the WebFacing Tool. Details of the behavior of INVITE are provided in the table below.

If the WAITRCD parameter is set to *IMMED, the browser will time out immediately. You can override the value for WAITRCD in the Run Time properties for your WebFacing project to change the time the browser waits for completion of a read-from-invited-devices operation. In the Run Time properties page for your WebFacing application, select Project and check the **Override the maximum record wait time (WAITRCD) specified for DDS files** check box and specify the number of seconds or *NOMAX for no time limit.

When the INVITE operation times out, an alert is issued in the browser if you have enabled the time-out dialog for INVITE. In the Run Time properties page for your WebFacing application, select Project and check the **Show time-out dialog for INVITE** check box. When the alert is issued in the browser, you can either reset the timer for that INVITE operation, or let the transaction time out. Note that when the transaction times out, any user input is lost. If you do not select the **Show time-out dialog for INVITE** check box on the Project properties page, no alert will appear when a timeout occurs.

**Note:** If you specify a value higher than the session time-out specified on the WAS server, the browser session will time out before the INVITE operation times out. You should specify a value lower than the session time-out setting on the WAS server.

You can change the content of the warning dialog by modifying the timeout.jsp file in \WebContent\webfacing\jsp\common\html for your WebFacing project. You can also modify the amount of time the warning is displayed. For example, if you want the warning to be dismissed immediately without user interaction, you can set the second parameter on the window.setTimeout method to 0.

You can modify the parameters for the warning dialog by updating the webface.js file for your release (for example, webfac512.js in version 5.1.2) in \WebContent\webfacing\ClientScript for your WebFacing project. In this file, you can modify attributes such as the warning dialog size and position. You can modify these attributes by updating the parameters for the window.showModalDialog method.

Note the following differences between INVITE processing for the 5250 and the WebFacing Tool:

<table>
<thead>
<tr>
<th></th>
<th>5250</th>
<th>WebFacing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>WAITRCD time</strong></td>
<td>WAITRCD time begins to expire when the read-from-invited-devices operation is performed by the application.</td>
<td>The WAITRCD time begins to expire when the write-with-invite operation is performed by the application.</td>
</tr>
<tr>
<td>Scenario</td>
<td>Description</td>
<td>Consequence</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>WAITRCD set to &quot;IMMED</strong>**</td>
<td>The user can key in data without interruption until pressing an AID key or the outstanding INVITE is cancelled. If an AID key is not pressed by the time the application performs a read-from-invited-devices operation, a time-out exception is returned to the application, although the device will still be input-capable.</td>
<td>The browser will immediately time out and the user will be unable to key in any data.</td>
</tr>
<tr>
<td><strong>WAITRCD set too low</strong></td>
<td>The user can key in data without interruption until pressing an AID key or the outstanding INVITE is cancelled. If an AID key is not pressed by the time the operation performs a read-from-invited-devices operation and the number of seconds specified for WAITRCD has passed, a time-out exception is returned to the application, although the device will still be input-capable.</td>
<td>The browser will time out after the number of seconds specified for WAITRCD has passed, and the user will be unable to key in any data.</td>
</tr>
<tr>
<td><strong>Application performs a read(wait) operation</strong></td>
<td>The user can key in data without interruption until pressing an AID key or the outstanding INVITE is cancelled. The device will still be input-capable.</td>
<td>The browser may time out and any data keyed in is lost. The read(wait) will cause the invited record format to be redisplayed and input-capable, without previously keyed-in data.</td>
</tr>
<tr>
<td><strong>Application performs a read-from-invited-devices operation in response to an expired read-from-invited-devices operation</strong></td>
<td>The user can key in data without interruption until pressing an AID key or the outstanding INVITE is cancelled. The device will still be input-capable.</td>
<td>The browser will time out and any data keyed in is lost. Each additional read-from-invited-devices operation will cause the invited record format to be redisplayed and input-capable, without previously keyed-in data.</td>
</tr>
</tbody>
</table>
### Cancelling an INVITE operation when WAITRCD is set to *NOMAX or too high

The outstanding INVITE request is cancelled almost immediately for an explicit request (such as ENDRCV) or implicit request (such as the write of another record format), as long as the request is processed before an AID key is pressed. The display will still be input-capable.

The outstanding INVITE request is cancelled when the browser times out or the user presses an AID key. An implicit request is completed only upon browser time-out. If no AID key is pressed, a cancel request cannot complete until the number of seconds set for WAITRCD has passed. If WAITRCD is *NOMAX and no AID key is pressed, the request will never complete.

###Suspending an INVITE operation when WAITRCD is set to *NOMAX or too high

The outstanding INVITE request is almost immediately suspended.

The suspend request cannot complete until the number of seconds set for WAITRCD has passed. If WAITRCD is *NOMAX and no AID key is pressed, the request will never complete.

---

**INZINP**

**Category**

Screen Building

**Additional information**

**INZRCD**

**Category**

Screen Building

**Additional information**

**KEEP**

**Category**

Screen Building

**Additional information**

**LOCK**

**Category**

Screen Building

**Additional information**

**LOGINP**

**Category**

I/O Buffer

**Additional information**

This keyword is fully supported. The input buffer is written to the WebFacing job log.
Note: To find the active jobs for WebFacing, look under the QINTER subsystem for QQFxxxxxxx.

BLINK

Category
Cursor

Additional information

LOGOUT

Category
I/O Buffer

Additional information
This keyword is fully supported. The output buffer is written to the WebFacing job log.

Note: To find the active jobs for WebFacing, look under the QINTER subsystem for QQFxxxxxxx.

LOWER

Category
I/O Buffer

Additional information

MAPVAL

Category
Field Values

Additional information

MDTOFF

Category
Screen Building

Additional information
MDTOFF is only partially supported. It is supported for the case where it is used to set a changed data tag off so that the CHANGE keyword response indicator will not be triggered. For example, imagine a record RECORD1 is written to the screen. The user then changes a field in RECORD1, so the changed data tag for that field is set on. When RECORD1 is read, if there is a CHANGE keyword specified on the record, the CHANGE response indicator will be turned on. After that, if another record RECORD2 with MDTOFF specified is written, the changed data tag for the field on RECORD1 will be cleared and when RECORD1 is read a second time, the CHANGE response indicator for the record will no longer be turned on.

However, MDTOFF is not currently supported to affect on-screen behaviour such as validity checking. For example, say a record RECORDA has a field FIELDA with both VALUES and DSPATR(MDT) specified. Before RECORDA is written to the screen, FIELDA is given a value that is not specified in its values keyword. Then RECORDA is written to the
screen and the user must change the value of FIELD A to a value specified in its VALUES keyword or validity checking will fail if the user presses "Enter" or a command function key. However, if a record RECORD B with MDTOFF is written to the screen, it will clear the changed data tag set on FIELD A and the validity checking will be bypassed if the user presses "Enter" or a command function key. In WebFacing, we do not currently support these type of on-screen behaviour consequences of MDTOFF.

<table>
<thead>
<tr>
<th>Category</th>
<th>Graphical Look</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLTCHCFLD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MNUBAR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MNUBARSW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLKFOLD</td>
<td>Field Position</td>
<td></td>
</tr>
</tbody>
</table>

On a 5250 display, the default is for the data to be folded at the end of the physical line. This field-level keyword is used for named, output-only
fields (but not message or program-to-system fields) that are defined so that they overflow onto subsequent display lines. The keyword causes folding to occur at a blank in the data rather than at the end of the display line. It is used to make long text fields easier to read. On the Web the data will not be folded.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Category</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNUCNL</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>MOUBTN</td>
<td>Function Keys</td>
<td></td>
</tr>
<tr>
<td>MSGALARM</td>
<td>Messages</td>
<td></td>
</tr>
<tr>
<td>MSGCON</td>
<td>Constants</td>
<td></td>
</tr>
<tr>
<td>MSGID</td>
<td>Field Values</td>
<td></td>
</tr>
<tr>
<td>MSGLOC</td>
<td>Messages</td>
<td></td>
</tr>
<tr>
<td>NOCCSID</td>
<td>NLS Enablement</td>
<td></td>
</tr>
</tbody>
</table>
OPENPRT

Category
Screen Building

Additional information

OVERLAY

Category
Screen Building

Additional information
This keyword is fully supported. However, it may be limited by the support level of other keywords. For example, if a record that has the OVERLAY keyword specified is already on the display, it is deleted and rewritten as a new record even when the PUTOVR, PUTRETAIN, or CLRL keywords are specified.

There is a problem when a record is positioned on the screen between a subfile control and its subfile. For example, RecordA occupies lines 10-12, SFLCTLB occupies lines 1-9, and SubfileB (the subfile record associated with SFLCTLB) occupies lines 13-23. If RecordA is written, then SFLCTLB is written, RecordA will be cleared from the screen. The regular 5250 behaviour would be to leave RecordA on the screen in this case.

OVRATR/OVRDTA

Category
Screen Building

Additional information

CAnn/CFnn

Category
Function Keys

Additional information
This keyword is fully supported except that the optional text is ignored by conversion. This text has no function in the file or the program other than as a comment.

The related indicator information will still be available in the display file when using commands such as DSPFFD on IBM i. However, these comments will not be available after conversion in the JSPs or XML files.

PAGEDOWN/PAGEUP

Category
Function Keys

Additional information
After conversion, the PAGEDOWN/PAGEUP keywords are represented as a push buttons on the Web page. These buttons will replace the use of the Page Down/Page Up keys on the user's keyboard. If PAGEDOWN or PAGEUP is disabled in the DDS source, no push button will be placed on the converted Web page for the disabled keyword.
The optional text is ignored by conversion, however, it is still included on the list created at program compilation time to explain the intended use of the indicator. This text has no function in the file or the program other than as a comment.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Category</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>PASSRCD</td>
<td>Screen Building</td>
<td></td>
</tr>
<tr>
<td>PRINT(*PGM/response ind)</td>
<td>Function Keys</td>
<td></td>
</tr>
<tr>
<td>PRINT(PRINT(lib/file))</td>
<td>Function Keys</td>
<td></td>
</tr>
<tr>
<td>PROTECT</td>
<td>Screen Building</td>
<td></td>
</tr>
<tr>
<td>PSHBTNCHC</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>PSHBTNFLD</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>PUTOVR</td>
<td>Screen Building</td>
<td></td>
</tr>
</tbody>
</table>
**Additional information**

PUTOVR is a record-level keyword used with OVRATR or OVRDTA to permit the override of either display attributes or data contents (or both) of specific fields. By using PUTOVR, you can reduce the amount of data sent to the 5250 display device. In WebFacing, all fields are sent to the Web browser whenever a page is displayed, and WebFacing does not support the partial update of a Web page. As a result, there is no performance gain by using PUTOVR in WebFacing. The keyword is not supported by WebFacing and will be ignored, so your application may not work the same as 5250. For example, on 5250 the output operation of a record that has PUTOVR specified will only update the PUTOVR record (that is, update the fields that have OVRATR or OVRDTA specified). Any existing records on the display will remain untouched. However, when the same output operation is executed in WebFacing, any existing records on the display will be removed if OVERLAY is not specified on the PUTOVR record. To avoid existing records being removed from the display, you can add the OVERLAY keyword on the record which has PUTOVR specified.

---

**PUTRETAIIN**

**Category**

Screen Building

**Additional information**

**CCSID**

**Category**

NLS Enablement

**Additional information**

**RANGE (Alphanumeric)**

**Category**

Validation

**Additional information**

**RANGE (Numeric)**

**Category**

Validation

**Additional information**

On the Web, for numeric fields, RANGE checking is always performed when the field is changed by the user or its changed data tag (MDT) is set on using DSPATR(MDT). The MDT tag is not turned off when the changed field is checked. As a result, data that does not fall into the values specified by the RANGE keyword cannot be submitted to the program if the MDT tag is never set on.

**REF**

**Category**

Compile-time values

**Additional information**
For the REF or REFFLD keywords, you must ensure that the library containing the referenced database fields is in the library list for the profile used during conversion. Otherwise, the following error will occur during conversion and record formats containing the referenced fields will not be converted:

File specified on keyword REF or REFFLD was not found.

**REFFLD**

Category
Compile-time values

Additional information
For the REF or REFFLD keywords, you must ensure that the library containing the referenced database fields is in the library list for the profile used during conversion. Otherwise, the following error will occur during conversion and record formats containing the referenced fields will not be converted:

File specified on keyword REF or REFFLD was not found.

**RETKKEY/RETCMDKEY**

Category
Function Keys

Additional information
Currently, the RETKEY and RETCMDKEY keywords specified on a record allow the current active keys to be retained on the screen when the record is written. However, the response indicators specified on the retained keys will not be communicated back to the RPG program as they would on the 5250.

**RETLCKSTS**

Category
Screen Building

**RMVWDW**

Category
Window

**ROLLUP/ROLLDOWN**

Category
Function Keys

Additional information
After conversion, the ROLLUP/ROLLDOWN keywords are represented as a push buttons on the Web page. These buttons will replace the use of the Page Down/Page Up keys on the user's keyboard respectively. If ROLLUP
or ROLLDOWN is disabled in the DDS source, no push button will be placed on the converted Web page for the disabled keyword.

The optional text is ignored by conversion, however, it is still included on the list created at program compilation time to explain the intended use of the indicator. This text has no function in the file or the program other than as a comment.

**RTNCSRLOC(*RECNAMEl*WINDOW)**

**Category**
Cursor

**Additional information**
Since a caret is often used in Web applications instead of a cursor, the third optional parameter which returns the position of the cursor within a field is supported as follows:

- output-only fields return a cursor position of 1 no matter where the user clicks
- input-capable fields return:
  - The integer 1 when the caret is before the first position
  - The integer 1 through $n$ when the caret is to the right of the corresponding cursor position

**RTNCSRLOC(*MOUSE)**

**Category**
Cursor

**Additional information**

**CHANGE**

**Category**
I/O Buffer

**Additional information**
This keyword is fully supported except that the optional text is ignored by conversion. This text has no function in the file or the program other than as a comment.

The related indicator information will still be available in the display file when using commands such as DSPFFD on IBM i. However, these comments will not be available after conversion in the JSPs or XML files.

**RTNDTA**

**Category**
Screen Building

**Additional information**

**SETOF/SETOFF**

**Category**
I/O Buffer

**Additional information**
The SFL keyword is supported. However, it is limited in that all of the related subfile keywords are not yet fully supported. See each of the individual subfile keywords for any additional limitations.

**Note:** WebFacing subfile implementation causes some differences in behavior between a 5250 application and a WebFacing application. The 5250 handles the subfile control record and subfile as separate entities, either of which can be overlapped. WebFacing handles them as one entity, which results in the subfile control record and subfile being removed if one of them is overlapped by other records.

**SFLCHCCTL**

**Category**
Subfiles

**Additional information**

**SFLCLR**

**Category**
Subfiles

**Additional information**

**SFLCSRPRG**

**Category**
Subfiles

**Additional information**

**SFLCSRRRN**

**Category**
Subfiles

**Additional information**

**SFLCTL**

**Category**
Subfiles

**Additional information**
WebFacing subfile implementation causes some differences in behavior between a 5250 application and a WebFacing application. The 5250 handles the subfile control record and subfile as separate entities, either of which can be overlapped. WebFacing handles them as one entity, which results in the subfile control record and subfile being removed if one of them is overlapped by other records.
The SFLEND keyword is supported as a scroll bar by default on the Web. The scroll bar indicates the following information about the subfile:

- Where the user is at in the subfile.
- How big the subfile is.
- What proportion of the subfile the user is viewing

The plus sign (+) or text (More... or Bottom) will not be displayed on the Web.

Note that the scroll bar shown on the Web is different from the scroll bar provided by SFLEND(*SCRBAR) on the 5250. On the 5250, if *SCRBAR is used as a parameter on SFLEND, a graphical scroll bar will be shown for a graphical display. In this case, clicking on the up or down arrow acts differently than pressing Page Up or Page Down. Clicking on the up or down arrow will only scroll one subfile record, while pressing Page Up or Page Down will scroll one page of subfile records. On the Web, clicking on
the up or down arrow will function the same as pressing Page Up or Page Down, and will scroll one page of subfile records.

<table>
<thead>
<tr>
<th>Category</th>
<th>Subfiles</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFLENTER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLFOLD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLINZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLLIN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLMLTCH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLMODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SFLMSG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHCAVAIL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**SFLROLVAL**

**Category**
Subfiles

**Additional information**
SFLROLVAL is not implemented in general. However, some DDS programmers purposefully set the SFLROLVAL field to 0 or negative so that the Page Up and Page Down keys can be disabled (an error is issued when they are used), and they can define their own function keys to control the paging. In this case, they normally do not show the SFLROLVAL field. We check if the SFLROLVAL is invalid (0 or negative) and the SFLROLVAL field has display attribute non-display conditioned on. In this case we do not display the scrollbar.

**SFLRTNSEL**

**Category**
Subfiles

**Additional information**

**CHCCTL**

**Category**
Graphical Look

**Additional information**

**SFLSCROLL**

**Category**
Subfiles

**Additional information**

**SFLSIZ**

**Category**
Subfiles

**Additional information**

**SFLSNGCHC**

**Category**
Subfiles

**Additional information**

**SLNO (*VAR)**

**Category**
Field Position

**Additional information**
**SLNO(n)**

**Category**
- Field Position

**Additional information**

**SGHCCHCFLD**

**Category**
- Graphical Look

**Additional information**

**SYSNAME**

**Category**
- Constants

**Additional information**

**TEXT**

**Category**
- Compile-time values

**Additional information**
- The related TEXT information is stored in the display file object. After the display file is compiled, the related TEXT information will still be available when using commands such as DSPFFD on the IBM i. However, these comments will not be available after conversion in the JSP or XML files.

**TIME**

**Category**
- Constants

**Additional information**

**TIMFMT**

**Category**
- Validation

**Additional information**

**CHCSLT**

**Category**
- Graphical Look

**Additional information**

**TIMSEP**

**Category**
- Appearance

**Additional information**
UNLOCK

Category
Screen Building

Additional information

USER

Category
Constants

Additional information

USRDFN

Category
Screen Building

Additional information

USRDSMPGT

Category
Screen Building

Additional information

USRRSTDSP

Category
Window

Additional information

VALNUM

Category
Validation

Additional information
On the Web, this keyword causes an error message to be returned if the user attempts to embed a SPACE, PLUS SIGN or MINUS SIGN between numeric digits in the field. However, when the PLUS SIGN or MINUS SIGN precedes the numeric digits, it is allowed on the Web but not on the 5250.

VALUES

Category
Validation

Additional information
The VALUES keyword is supported on the Web as a single choice select box listing the allowed values. The user will not be permitted to enter values that are not in this list.

If a program value is sent to the field, and it is not specified on the VALUES keyword, it may be submitted to the program if the user does not
change the field. Once the field has been modified or its data tag (MDT) changed, only entries defined by the VALUES keyword can be submitted to the program.

**VLDCMDKEY**

**Category**
Function Keys

**Additional information**

**WDWBORDER**

**Category**
Window

**Additional information**

**ALTHELP**

**Category**
Function Keys

**Additional information**
This keyword is supported by default. The ALTHELP feature of allowing a command attention (CA) key as an alternative Help key is not necessary in Web applications and is not supported. Since the HELP keyword is supported as a push button, help will be available on the Web Application and a second push button is not required.

**CHCUNAVAIL**

**Category**
Graphical Look

**Additional information**

**WDWTITLE**

**Category**
Window

**Additional information**
Parameters [title-text] and *[CENTER] [LEFT] [RIGHT] are supported. Parameters like [title-text-color], [title-text-display-attribute] and *[TOP] [BOTTOM] are not supported and will be ignored.

**WINDOW(definition)**

**Category**
Window

**Additional information**
Use this record-level keyword to specify that the record format you are defining will be displayed using a window. A window is information that overlays part of the display. A window is usually smaller than the actual workstation display, and can be positioned anywhere on the display. The
*NOMSGLIN and *RSTCSR parameters for this keyword are not supported. The *NOMSGLIN parameter specifies that the message line within the window should be removed and placed at the bottom of the display. The *RSTCSR parameter specifies that the user should be allowed limited function when the cursor is outside of the window.

**WINDOW**(reference)

Category
Window

Additional information

**WRDWRAP**

Category
Appearance

Additional information
Use this file, record, or field-level keyword for named fields that are defined such that they overflow onto subsequent display lines or for continued-entry fields. The keyword causes wrapping to occur at a blank in the data rather than at the end of the data line. It is used to make long text fields easier to read. On the 5250, the default is for data to be wrapped at the end of the physical line or continued-entry field segment.

The effect of this keyword will be the default behaviour in the converted Web applications.

**(7-16) Conditioning**

Category
Indicators

Additional information

**(29) Reference**

Category
Compile-time values

Additional information
For the REF or REFFLD keywords you must ensure that the library containing the referenced database fields is added to your library list before you convert your DDS source files.

**(30-34) Length**

Category
Appearance

Additional information
(35) Data type (Buffer)

Category
I/O Buffer

Additional information

(35) Data type/Keyboard shift

Category
Validation

Additional information
The use of this entry is supported for all Keyboard Shifts except currently for Floating Point (F).

Signed Numeric (S): Only numbers 0 through 9 and '-' are allowed.

Numeric shift (N) and Numeric Only (Y): WebFacing ignores leading and trailing spaces but allows embedded spaces. We allow for +/- signs before and after the numbers, any spaces embedded between the numbers, and the + or - signs being treated as trailing spaces. We deviate from 5250 behavior when displaying the data. For example, a field with value -5 (length 4 and 0 decimal position) will be shown as -0005 instead of 000N.

(36-37) Decimal Position

Category
I/O Buffer

Additional information

CHECK

Category
Validation

Additional information
The following values will NOT be supported:
• M10F (modulus 10 checking as you type) will be supported as M10
• M11F (modulus 11 checking as you type) will be supported as M11
• M10/M10F and M11/M11F are ignored for alphanumeric fields

(38) Usage M

Category
Messages

Additional information

(38) Usage (I/O/B/H/P)

Category
Field Values

Additional information
<table>
<thead>
<tr>
<th><strong>Category</strong></th>
<th><strong>Location</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Position</td>
<td></td>
</tr>
</tbody>
</table>

**Additional information**

Fields will be positioned using HTML Tables. There will be one table for the entire DDS screen and it will contain 24/27 rows or 80/132 columns according to the display size of the screen. The fields will be placed in the table cell corresponding to their DDS row and column.

Wrapping (fields that extend beyond the screen edge will wrap to subsequent lines):

For output-only fields (including constant fields), wrapping will occur at the same location in the field value as on the 5250 screen. A SPAN tag will be used for each line the field occupies on the screen. Wrapped output-only fields will occupy the same space in the HTML table that they do in the on the 5250 screen.

The BLKFOLD keyword has not been implemented.

For continued-entry fields, wrapping occurs at a blank in the data rather than at the end of the data line. This behavior would be the same as having the WRDWRAP keyword specified. On the 5250, the default is for data to be wrapped at the end of the physical line or continued-entry field segment.

If input fields extend beyond the right boundary of the display size (80 or 132 columns), they wrap onto multiple lines. These fields cannot be exactly represented in HTML which only allows rectangular fields. We will determine the largest rectangle that fits within the 5250 screen area of the field. If that rectangle is on only one line, then an INPUT tag will be used whose maximum is greater than its visual size. Otherwise, a TEXTAREA tag will be used.

**Display Size:** WebFacing will only convert a given record to a single display size. Additional DSPSIZ information is available in the DSPSIZ document of the WebFacing DDS keyword support tables.

### DSPATR(PR CS)

**Category**
Appearance

**Additional information**

The CS display attribute may change depending on the style that you have selected for your Web application.

### DSPATR(PO)

**Category**
Appearance

**Additional information**

The cursor may not be positioned to the correct field when DSPATR(PC) is specified with:

1. An input field with both DSPATR(ND) and DSPATR(PR) specified.
2. A field in a subfile that has SFLFOLD, when the subfile is in truncate mode, and that field is not visible as a result.

<table>
<thead>
<tr>
<th><strong>DSPATR(BL)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td>Appearance</td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>DSPATR(SP)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td>Field Values</td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>CHGINPDT(FE)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td>Appearance</td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

This keyword applies the field exit keyboard controls of the affected input-capable fields just like the field-level CHECK(FE) keyword.

On the 5250, field exit keys include the Field Exit, Field+, Field-, and cursor movement keys. On the Web, only the Field Exit key is supported. You can enable a field exit key for your WebFacing project. In the **Run Time** properties page for your WebFacing application, select **Project**, check the **Field exit key** check box and select a key from the key list to enable a field exit key. There is no equivalent for the Field+ and Field- keys. You need to enter the + or - in front of the number manually and then use the tab key, mouse, or the Field Exit key.

<table>
<thead>
<tr>
<th><strong>CHECK(ER)/AUTO(RA)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td>Validation</td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>CHECK(RB)/AUTO(RAB)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td>I/O Buffer</td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

On the Web, Right-justified with blank fill will happen when the cursor is no longer in the field where data has been entered. On the 5250, Field Exit, Field+, Field-, and cursor movement keys can be used to move to the next input field. On the Web, only the Field Exit key is supported. You can enable a field exit key for your WebFacing project. In the **Run Time** properties page for your WebFacing application, select **Project**, check the **Field exit key** check box and select a key from the key list to enable a field exit key. There is no equivalent for the Field+ and Field- keys. You need to enter the + or - in front of the number manually and then use the tab key, mouse, or the Field Exit key to move to another field.
**CHECK(RZ)/AUTO(RAZ)**

Category
Validation

Additional information
On the Web, Right-justified with blank fill will happen when the cursor is no longer in the field where data has been entered. On the 5250, Field Exit, Field+, Field-, and cursor movement keys can be used to move to the next input field. On the Web, only the Field Exit key is supported. You can enable a field exit key for your WebFacing project. In the Run Time properties page for your WebFacing application, select Project, check the Field exit key check box and select a key from the key list to enable a field exit key. There is no equivalent for the Field+ and Field- keys. You need to enter the + or - in front of the number manually and then use the tab key, mouse, or the Field Exit key to move to another field.

**CHGINPDFT(ME MF LC CS)**

Category
Validation

Additional information
The CS display attribute may change depending on the style that you have selected for your Web application.

**CHGINPDFT(RI HI)**

Category
Appearance

Additional information
The RI and HI display attributes may change depending on the style that you have selected for your Web application.

**CHGINPDFT/CHGINPDFT(UL/BL)**

Category
Appearance

Additional information
This keyword applies the specified display attributes on the affected input-capable fields. On the Web, input capable fields will always be displayed by default as input boxes with borders. On a 5250 display, input capable fields are displayed as entry fields and DSPATR(UL) is applied to input and both fields by default when CHGINPDFT is not specified.

CHGINPDFT will not be supported for input-capable fields on the Web for the following parameter values:

<table>
<thead>
<tr>
<th>Parameter value</th>
<th>Equivalent DDS keyword</th>
<th>Meaning</th>
<th>Web properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
<td>DSPATR(UL) specified but not selected</td>
<td>Remove underline</td>
<td>Input field border is not removed</td>
</tr>
<tr>
<td>BL</td>
<td>DSPATR(BL)</td>
<td>Blinking field</td>
<td>Not Supported</td>
</tr>
<tr>
<td>Parameter value</td>
<td>Equivalent DDS keyword</td>
<td>Meaning</td>
<td>Web properties</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------</td>
<td>---------</td>
<td>----------------</td>
</tr>
<tr>
<td>UL</td>
<td>DSPATR(UL)</td>
<td>Underline</td>
<td>Supported as a text attribute and not as a border attribute</td>
</tr>
</tbody>
</table>

**CHKMSGID**

Category
- Validation

Additional information

**CHOICE**

Category
- Graphical Look

Additional information

**ALTNAME**

Category
- Compile-time values

Additional information
- This keyword is supported by default. When a program is compiled, the alternative name is brought into the program instead of the DDS record format name. This information is then resolved using the compiled DDS object before a request is made to any converted record formats. The converted record formats use the original DDS record format name.

**CHRID**

Category
- NLS Enablement

Additional information

**CLEAR**

Category
- Function Keys

Additional information
- After conversion, the CLEAR keyword is represented as a push button on the Web page. This button will replace the use of the CLEAR key on the user's keyboard. If CLEAR is disabled in the DDS source, no push button will be placed on the converted Web page.

The optional text is ignored by conversion, however, it is still included on the list created at program compilation time to explain the intended use of the indicator. This text has no function in the file or the program other than as a comment.
CLRL(nn|*END)

Category
Screen Building

Additional information
CLRL records are supported as overlapping layers on the screen with opaque backgrounds by default. Although this is non-standard behaviour, it should be satisfactory in a browser environment. The background color of the CLRL layers will be the same as the background color of your selected style. You must use a style with an opaque background color otherwise the text in underlying records will show through and may be displayed in the same location as data from the overlapping layer. If you want to change the default layer background, you can update the wf_layer style class in the apparea.css file under the /WebContent/webfacing/styles/apparea directory.

It is possible with output-only CLRL records to occupy lines above and below the records that the application is expecting input from. This will not work in WebFacing because the entire CLRL record will be made opaque and the overlapped portions of the records below will not be visible and cannot be clicked on. The circumvention for this unsupported scenario is to split the output-only CLRL record into two records containing the fields above and below the input record respectively.

The CLRL keyword is not yet supported for record formats with the WINDOW keyword and will be ignored.

Window Simulation with CLRL: If the CLRL(*NO) or CLRL(nn) keywords are used to implement a window-like record in DDS, WebFacing will attempt to construct a window for that record format to support this common usage. For CLRL(nn), nn must be less than the record length for it to be recognized as a window.

In order for a record to be recognized as a window it must not have any fields outside of the rectangular area intended to be the window. A work-around for developers that have defined function key labels (or other nonessential constants) outside the window area, is to hide these fields using Web Settings. WebFacing will respect these Web Settings and detect the rectangle properly.

The CLRL window recognition algorithm is based on commonly used CLRL window patterns. The following rules, for the top and side borders of the simulated window must be true for the detection to succeed:

1. For the top border, WebFacing checks for a constant field or an output field occupying the space from (first row, first column) to (first row, last column) and uses it as the window title. If a constant field is used, and the same character is used repeatedly from (first row, first column+1) to (first row, last column-1), the characters used to form the top border will not be used as a title and will simply be removed.

2. For the side borders, WebFacing checks for a constant field from the record’s second row to the second last row. The same string must be used for the left and right border for the simulated window. If this is true, the strings used to form the side borders will be removed.

Note: The bottom border of the simulated window will not be altered because this row is sometimes used to display message information.
The following 5250 behavior is not intuitive and will not be emulated: A CLRL record with no input-capable field does not occupy space on the display. If a CLRL(*ALL) record with no input-capable fields is written, the screen is cleared first and then the record is displayed. However, records that were on the screen before are not removed from the Active Record Table and will be used for OVERLAY checking later. If an OVERLAY record is written, it will first check if records in the Active Record Table are overlapped. If so then those records are removed from the Active Record Table and lines from those records are cleared from the display before the OVERLAY record is written to the display. The above rule also applies to CLRL(nn!*END!*NO). WebFacing deviates from this behaviour by only considering records that are on the display when doing overlap checking. Records that are in the Active Record Table but are not on the display are not considered.

**CLRL(*ALL)**

**Category**
Screen Building

**Additional information**
Additional CLRL information is available in the CLRL(nn!*END) document of the WebFacing DDS keyword support tables.

**CLRL(*NO)**

**Category**
Screen Building

**Additional information**
Additional CLRL information is available in the CLRL(nn!*END) document of the WebFacing DDS keyword support tables.

**CMP**

**Category**
Validation

**Additional information**
The CMP/COMP keywords for fields defined with a data type of character are supported only for the relational operators EQ and NE. Numeric values are fully supported.

**CNTFLD**

**Category**
Graphical Look

**Additional information**
Continued-entry fields are converted into single entry fields in the form of HTML text areas. In DDS, if a 60 character input field is specified with CNTFLD(10), then you will have 6 lines of 10 characters each for the field and the end-user will only be able to enter a maximum of 60 characters. The text area is similarly defined with 6 rows and 10 columns (e.g. `<TEXTAREA rows="6" cols="10">`) and the user will only be able to enter a maximum of 60 characters. However, the number characters in each line
and the number of lines that are displayed in the text area will differ depending on the font that is used for the Web application.

For example, on a 5250 display, some text entered into a field specified with CNTFLD(30) might look like:

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAA
```

On the Web, with a non-monospace font, you might see:

```
AAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAA
```

**Note:** The keyboard Enter key is not enabled for continued-entry fields.

<table>
<thead>
<tr>
<th>COLOR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COMP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CSRINPONLY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ALTPAGEDWN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Category</strong></td>
</tr>
<tr>
<td><strong>Additional information</strong></td>
</tr>
</tbody>
</table>
not necessary in Web applications and is not supported. Since the
PAGEDWN keyword is supported as a push button, it will be available on
the Web Application and a second push button is not required.

---

**CSRLOC**

**Category**
- Cursor

**Additional information**

---

**DATE**

**Category**
- Constants

**Additional information**

---

**DATEFMT**

**Category**
- Validation

**Additional information**

---

**DATESEP**

**Category**
- Appearance

**Additional information**

---

**DFT**

**Category**
- Field Values

**Additional information**

In standard use, the DFT keyword can be used on input-only, output-only,
and output/input fields. Under WebFacing, DFT only supports input-only
fields. The reason for this is that for output-only and output/input fields,
the user must specify PUTOVR at the record level and OVRDTA at the
field level. However, as PUTOVR is not supported by WebFacing, only
input-only fields are possible.

---

**DFT (implicit)**

**Category**
- Field Values

**Additional information**

Unnamed constant fields specified using the implicit DFT keyword are
represented as HTML text in the converted record JSPs. They do not affect
the dynamic data content in the record beans.
**DFTVAL**

**Category**  
Field Values

**Additional information**  
DFTVAL is fully supported but OVRDTA is not supported yet. Discrepancies with 5250 behaviour may be noticed in the following scenario: If the screen remains in PUTOVR mode for 3 or more writes of the same record but OVRDTA is not active for this field, the program value is shown instead of the DFTVAL due to the lack of support for OVRTDTA keyword.

---

**DLTCHK**

**Category**  
Compile-time values

**Additional information**  
The conversion process fully supports referenced fields and will ignore all validity checking and CHKMSGID keywords if DLTCHK is specified.

**Note:** When using referenced fields, you must ensure that the library containing the referenced database is in the library list for the profile used during conversion.

---

**DLTEDT**

**Category**  
Compile-time values

**Additional information**  
The conversion process fully supports referenced fields and will ignore any edit code or edit word keywords if DLTEDT is specified.

**Note:** When using referenced fields, you must ensure that the library containing the referenced database is in the library list for the profile used during conversion.

---

**DSPATR (program-to-system-field)**

**Category**  
Appearance

---

**ALTPAGEUP**

**Category**  
Function Keys

**Additional information**  
This keyword is supported by default. The ALTPAGEUP feature of allowing a command function (CF) key as an alternative Page Up key is not necessary in Web applications and is not supported. Since the PAGEUP keyword is supported as a push button, it will be available on the Web Application and a second push button is not required.
**DSPATR(OID)**

**Category**
- Appearance

**Additional information**
Since the card sweeper is specific to the 5250 hardware, this attribute would have no meaning in an internet application in a web browser.

**DSPATR(MDT)**

**Category**
- Appearance

**DSPATR(UL HI RI ND)**

**Category**
- Appearance

**Additional information**
The attributes specified using this keyword will be displayed according to the Style that is specified during conversion.

**DSPMOD (unconditioned)**

**Category**
- Screen Building

**Additional information**
Conditioned DSPMOD keywords are not yet supported. This means that it will not be possible to dynamically change which display size a record should be shown in at runtime. A record-level web setting exists to specify which display size a record should be converted to.

**DSPMOD (with option indicators)**

**Category**
- Screen Building

**Additional information**
Conditioned DSPMOD keywords are not yet supported. This means that it will not be possible to dynamically change which display size a record should be shown in at runtime. A record-level web setting exists to specify which display size this record should be converted to.

**DSPRL**

**Category**
- NLS Enablement

**Additional information**
Experts in Bidi development have indicated that typically the screens are already coded in a right to left fashion and this keyword is not necessary to flip the direction of strings. Since there is no demand for this keyword, there are no plans to implement it.
DSPSIZ

Category
Screen Building

Additional information
WebFacing will only convert a given record to a single display size. By default that will be the primary display size; that is, the first parameter of the DSPSIZ keyword. This can be overridden with an unconditioned DSPMOD keyword. However, for conditioned DSPMOD keywords, if the secondary display size is desired, it can be specified using the Web Settings view in the IDE.

The WebFacing Tool does not support System/38 DDS. IBM i supports the following display sizes:
- *DS3 - 24x80
- *DS4 - 27x132

System/38 supported two more display sizes:
- *DS1 - Console
- *DS2 - 12x80

Prior to WebSphere Development Studio Client for iSeries, V6.0, if a field contained multiple positions, the *DS3 or *DS4 position would be used in conversion. The following example contains a constant (Text constant) which is defined on the bottom line of the display:

```
A DSPSIZ(*DS2 *DS3)
A R RECORD1
A 12 1'Text constant'
A *DS3 24 1
```

If you converted this source and targeted the *DS3 display size, prior to V6.0 the constant would have appeared on line 24. Now it will appear on line 12. The current parser drops the *DS2 parameter from the DSPSIZ keyword and assumes *DS3 is the primary display size. Therefore, it uses the position on the defining line of the constant, which is 12, 1.

DUP

Category
Function Keys

Additional information

EDTCDE

Category
Appearance

Additional information

EDTCDE(5-9)

Category
Appearance

Additional information
User-defined edit codes are not supported in WebFacing. However, you
can map them to one of the available system edit codes using the Edit Code Options in the Conversion properties page for your WebFacing application.

**ALWGPH**

Category

Screen Building

Additional information

**EDTMSK**

Category

Validation

Additional information

**EDTWRD**

Category

Appearance

Additional information

**ENTFLDATR**

Category

Appearance

Additional information

**ERASE**

Category

Screen Building

Additional information

**ERASEINP**

Category

Screen Building

Additional information

**ERRMSG**

Category

Messages

Additional information

Input-capable fields will always be reverse-imaged, regardless of what other COLOR or DSPATR settings are in effect for that field. In Web applications, there is no concept of locking the keyboard. The message will simply be displayed in a select box with nothing preventing the user from continuing on.
ERRMSGID

Category
Messages

Additional information
Input-capable fields will always be reverse-imaged, regardless of what other COLOR or DSPATR settings are in effect for that field. In Web applications, there is no concept of locking the keyboard. The message will simply be displayed in a select box with nothing preventing the user from continuing on.

Second-level help for the message when pressing the Help button with the focus on the error is not yet supported.

ERRSFL

Category
Messages

Additional information
On a 5250 display, if the message line overlaps a record already displayed on the screen, the ERRSFL keyword is ignored. On the Web, since the message line can still be displayed, the ERRSFL keyword is not ignored. The MSGLOC keyword which positions the message line is ignored.

FLDCSRRPG

Category
Cursor

Additional information

FLTIXDEC

Category
Appearance

Additional information

ALWROL

Category
Screen Building

Additional information

FLTPCN

Category
I/O Buffer

Additional information
<table>
<thead>
<tr>
<th>Command</th>
<th>Category</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRCDTA</td>
<td>Screen Building</td>
<td></td>
</tr>
<tr>
<td>GETRETN</td>
<td>Screen Building</td>
<td></td>
</tr>
<tr>
<td>GRDATTR</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>GRDBOX</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>GRDCLR</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>GRDLIN</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>GRDRCD</td>
<td>Graphical Look</td>
<td></td>
</tr>
<tr>
<td>HELP (2nd lev msg)</td>
<td>Function Keys</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### HELP (return to PGM)
- **Category**: Function Keys
- **Additional information**

### ASSUME (data passing)
- **Category**: Screen Building
- **Additional information**

### HELP (with HLPARA)
- **Category**: Function Keys
- **Additional information**

### HLPARA
- **Category**: Application Help
- **Additional information**

### HLPBDY
- **Category**: Application Help
- **Additional information**

### HLPCCLR
- **Category**: Application Help
- **Additional information**

### HLPCMDKEY
- **Category**: Application Help
- **Additional information**

### HLPDOC
- **Category**: Application Help
- **Additional information**

This keyword specifies the document name to be used to deliver help. Since document help is not supported, this keyword is not supported.
<table>
<thead>
<tr>
<th>HLPEXCLD</th>
<th>Category</th>
<th>Application Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>Category</td>
<td>Application Help</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPID</td>
<td>Category</td>
<td>Application Help</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPPNLGRP</td>
<td>Category</td>
<td>Application Help</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>ASSUME (don't clear the screen)</td>
<td>Category</td>
<td>Screen Building</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPRCD</td>
<td>Category</td>
<td>Application Help</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPRTN</td>
<td>Category</td>
<td>Function Keys</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
<tr>
<td>HLPSCHIDX</td>
<td>Category</td>
<td>Application Help</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Additional information</td>
</tr>
</tbody>
</table>
Chapter 10. WebFacing Tool -- current DDS keyword support

HOME

Category: Function Keys

**Additional Information**

After conversion, the HOME keyword is represented as a push button on the Web page. This button will replace the use of the HOME key on the user's keyboard. If HOME is disabled in the DDS source, no push button will be placed on the converted Web page.

On the keyboard, if you press the Home key and the cursor is not already at the home position, the cursor returns to the home position, whether or not you specify the HOME keyword. This feature will not be enabled when the HOME push button is selected on the converted Web page.

The optional text is ignored by conversion. However, it is still included on the list created at program compilation time to explain the intended use of the indicator. This text has no function in the file or the program other than as a comment.
**IGCALTTYP (validation)**

| Category | Validation |

**Additional information**

**WebFacing Tool -- current DDS keyword support**

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application.

<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALIAS</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTHELP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTNAM</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTPAGEDWN</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEUP</td>
<td>Yes (partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALWGPH</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWRUL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (don’t clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLANKS</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>BLKFOLD</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>CAnn/CFrnn</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHANGE</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCCCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSLT</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(RAZ)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDFT(FE)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(ME MF LC CS)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDFT(RI HD)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>CHGINPDFT/</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(UL/BL) *</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CLEAR *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLRL(nn) *END) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*ALL) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CNTFLD *</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>COLOR *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>COMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CSRINPONLY *</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>DATESEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>DFT (implicit) *</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL *</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT *</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DLTCHK *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTEDET *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DSPATR</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(MDT)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(OID) *</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PC) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PR CS) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>“DSPATR(SP)” on page 184</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(UL HI R I ND) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (unconditioned) *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPMOD (with option indicators) *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPRL *</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DSPSZ *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>“EDTCDE(5-9)” on page 192</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTCDE</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTMSK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTWRD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERASE</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERMSGID</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSC</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>FLDCSRPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLTFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>FLTPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>FRCDA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GETRETAINT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDCLR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLINE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>&quot;HELP (2nd lev msg)&quot; on page 196</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (return to PGM)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (with HL PARA)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPARA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPBDY</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCLR</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDOC</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HPLEXCLD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPID</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPNLGRF</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTCD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTTN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPSCIDX</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSEQ</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSEHLF</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HOME</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>IGCA2ITYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>IGCCNV</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INDARA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INDTXT</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>INVITE (multiple devices)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INVITE (single-device)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCRD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOGINP</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MDTOFF</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MLTCCHEDLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCNL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MOUBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVERLAY</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRATR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PAGEOP/PAGEUP</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PASSRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PRINT(*PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PSHTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PUTOVR</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAIEN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>No</td>
<td>Validation</td>
</tr>
<tr>
<td>RANGE (Numeric)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>REFFLD</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REF*</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>RETKEY/RETCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RETLKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>ROLLUP/ROLLDOWN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RTNCSRLOC(*MOUSE)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>RTNCSRLOC(*RECNAME</td>
<td>*WINDOW) *</td>
<td>Yes</td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SFLCHCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDLT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSF</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLEND *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLICHIC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCRD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSC</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXTCHIC</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGMQ/SFLMSGKEY/SFLMSGRCRD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRNA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROLVAL *</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNGCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFL *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>Keyword</td>
<td>Current Support</td>
<td>Category</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TEXT *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIMFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMSEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>UNLOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>USRDEN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSPMGT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRRSTDSP</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>VALNUMI</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>WDWBORDER</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDTITLE</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(definition)</td>
<td>Yes (partial)</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(reference)</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WRDWRAP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(29) Reference</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(30-34) Length</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type/Keyboard</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(39-44) Location</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
</tbody>
</table>

Related concepts:
- Chapter 1, “Introducing the WebFacing Tool,” on page 1

Related tasks:
- Chapter 3, “Creating a WebFacing Web project,” on page 11
- “Opening the WebFacing perspective” on page 12
- “Selecting the source members to convert” on page 13
- “Specifying CL commands for invoking your application” on page 14
- “Choosing a Web style” on page 12
- “Finishing the project and converting your DDS source” on page 15
- “Analyzing the conversion logs” on page 15
**WebFacing Tool -- current DDS keyword support**

* indicates that the behavior of the keyword, as implemented by WebFacing, may have differences or limitations compared to the standard 5250 implementation. Click the keyword’s link for more information.

**Note:** You can use the DDS keyword survey tool to help evaluate the level of keyword support that the WebFacing Tool will provide for your application. The DDS keyword survey tool can be downloaded from [IBM Support](https://www.ibm.com).

<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALIAS *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTHELP *</td>
<td>(partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTNAME *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>ALTPAGEDWN *</td>
<td>(partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ALTPAGEUP *</td>
<td>(partial)</td>
<td>Function Keys</td>
</tr>
<tr>
<td>ASSUME (don’t clear the screen)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLANKS *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CAnn/CFnn *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CHANGE *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(ER)/AUTO(RA)</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(RB)/AUTO(RAB) *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>CHECK(RZ)/AUTO(RAZ) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHECK(FE) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(FE) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(ME MF LC CS) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CHGINPDFT(RI HI) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT/</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHGINPDFT(UL/BL) *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>CHKMSGID</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CLEAR *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>CLRL(\nn</td>
<td>*END)</td>
<td>Yes</td>
</tr>
<tr>
<td>CLRL(*ALL)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CLRL(*NO)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>CMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CNTFLD *</td>
<td>Yes</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>COLOR *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>COMP *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>CSRINPONLY *</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>CSRLOC</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>DATE</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>DATEFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>Keyword</td>
<td>Sort by</td>
<td>Category</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>DATESEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DFT (implicit)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFTVAL</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DFT</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DLCHK</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DLTEDT</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>DSPATR(BL)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(MDT)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PC)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(PR CS)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(SP)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>DSPATR(UL HI RI ND)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPATR(program-to-system-field)</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (unconditioned)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPSIZ</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>EDTCDE</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>EDTMSK</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>EDTWRD</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASE</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ERRMSGID</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRMSC</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>ERRSFL</td>
<td>Yes</td>
<td>Messages</td>
</tr>
<tr>
<td>FRCDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>HELP (return to PGM)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HELP (with HLPPARA)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPPARA</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPBDY</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPPCLR</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTNLGRP</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTCD</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTIN</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPSEQ</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPRTITLE</td>
<td>Yes</td>
<td>Application Help</td>
</tr>
<tr>
<td>HOME</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>IGCALITYP</td>
<td>Yes</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INDARA</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INDTXT</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>INVITE (single-device)</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>KEEP</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOGINP *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOGOUT *</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>LOWER</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>MDTOFF *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MSGCON</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>MSGID</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>OVERLAY *</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PAGEDOWN/PAGEUP*</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PRINT(*PGM/response ind)</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PROTECT</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Numeric) *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>REFFLD *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>REF *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>RETKEY/RETMDKEY*</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RMVWDW</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>ROLLUP/ROLLDOWN *</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>RTNCRSRLOC(‘RECNAME</td>
<td>Yes</td>
<td>Cursor</td>
</tr>
<tr>
<td>‘WINDOW)*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTNDTA</td>
<td>Yes</td>
<td>Screen Building</td>
</tr>
<tr>
<td>SETOF/SETOFF</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>SFLCLR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSRRRRN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCTL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDLT</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDROP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSPCTRL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLDSP</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLEND *</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLFOLD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLINZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLIN</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMODE</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGID</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGKEY</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSGRCRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMSG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLNXTCG</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPAC</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLPGQ/SFLMSGKEY/ SFLMSGRCRCD</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>SFLRCDNBR</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRNA</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSCROLL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSIZ</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SEL</td>
<td>Yes</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SLNO (*VAR)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SLNO(n)</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>SYSNAME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TEXT *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>TIME</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>TIMFMT</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>TIMSEP</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>USER</td>
<td>Yes</td>
<td>Constants</td>
</tr>
<tr>
<td>VALNUM *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VALUES *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>VLDCMDKEY</td>
<td>Yes</td>
<td>Function Keys</td>
</tr>
<tr>
<td>WDWTTITLE *</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(definition)</td>
<td>Yes (partial)</td>
<td>Window</td>
</tr>
<tr>
<td>WINDOW(reference)</td>
<td>Yes</td>
<td>Window</td>
</tr>
<tr>
<td>WRDWRAP *</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(7-16) Conditioning</td>
<td>Yes</td>
<td>Indicators</td>
</tr>
<tr>
<td>(29) Reference *</td>
<td>Yes</td>
<td>Compile-time values</td>
</tr>
<tr>
<td>(30-34) Length</td>
<td>Yes</td>
<td>Appearance</td>
</tr>
<tr>
<td>(35) Data type (Buffer)</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(35) Data type/Keyboard shift *</td>
<td>Yes</td>
<td>Validation</td>
</tr>
<tr>
<td>(36-37) Decimal Position</td>
<td>Yes</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>(38) Usage (I/O/B/H/P)</td>
<td>Yes</td>
<td>Field Values</td>
</tr>
<tr>
<td>(39-44) Location *</td>
<td>Yes</td>
<td>Field Position</td>
</tr>
<tr>
<td>ALARM</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWGPH</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ALWROL</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>ASSUME (data passing)</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>BLINK</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>BLKFOLD *</td>
<td>No</td>
<td>Field Position</td>
</tr>
<tr>
<td>CCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>CHCACCEL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCCTL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCSLT</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHCUNAVAIL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
</tbody>
</table>

Chapter 10. WebFacing Tool -- current DDS keyword support 209
<table>
<thead>
<tr>
<th>Sort by Keyword</th>
<th>Sort by current support</th>
<th>Sort by Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHOICE</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>CHRID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DSPATR(OID) *</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>DSPMOD (with option indicators) *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>DSPRL *</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>DUP</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>EDTCDE(5-9)</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ENTFLDATR</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>ERASEINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>FLDCSRPRG</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>FLTFIXDEC</td>
<td>No</td>
<td>Appearance</td>
</tr>
<tr>
<td>FLTPCN</td>
<td>No</td>
<td>I/O Buffer</td>
</tr>
<tr>
<td>GETRETAIN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>GRDATR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDBOX</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDCLR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDLIN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>GRDRCD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>HELP (2nd lev msg) *</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>HLPCMDKEY</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPDOC *</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPFULL</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLPSCIDX</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HLP SHELF</td>
<td>No</td>
<td>Application Help</td>
</tr>
<tr>
<td>HTML</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>IGCCNV *</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>INVITE (multiple devices) *</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZINP</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>INZRCD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>LOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>MAPVAL</td>
<td>No</td>
<td>Field Values</td>
</tr>
<tr>
<td>MLCCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARDSP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSEP</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBARSW</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUBAR</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MNUCNL</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>MOUBTN</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>MSGALARM</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>Sort by Keyword</td>
<td>Sort by current support</td>
<td>Sort by Category</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>MSGLOC</td>
<td>No</td>
<td>Messages</td>
</tr>
<tr>
<td>NOCCSID</td>
<td>No</td>
<td>NLS Enablement</td>
</tr>
<tr>
<td>OPENPRT</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>OVRATR/OVRDTA</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PASSRCRD</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PRINT/PRINT(lib/file)</td>
<td>No</td>
<td>Function Keys</td>
</tr>
<tr>
<td>PSHBTNCHC</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PSHBTNFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PULLDOWN</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>PUTOVR</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>PUTRETAiN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RANGE (Alphanumeric)</td>
<td>No</td>
<td>Validation</td>
</tr>
<tr>
<td>RETLCKSTS</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>RTNCSRLOC(*MOUSE)</td>
<td>No</td>
<td>Cursor</td>
</tr>
<tr>
<td>SFLCHCCTL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLCSRPRG</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLENTER</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLMLTCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLROVLVAL *</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLRTNSEL</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SFLSNGCHC</td>
<td>No</td>
<td>Subfiles</td>
</tr>
<tr>
<td>SNGCHCFLD</td>
<td>No</td>
<td>Graphical Look</td>
</tr>
<tr>
<td>UNLOCK</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDEN</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRDSPMG1</td>
<td>No</td>
<td>Screen Building</td>
</tr>
<tr>
<td>USRRSTDSP</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>WDWBORDER</td>
<td>No</td>
<td>Window</td>
</tr>
<tr>
<td>(38) Usage M</td>
<td>No</td>
<td>Messages</td>
</tr>
</tbody>
</table>
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Intellectual Property Dept. for Rational Software
IBM Corporation
5 Technology Park Drive
Westford, MA 01886
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.
If you are viewing this information softcopy, the photographs and color illustrations may not appear.

**Programming interface information**

This Developing WebFacing Applications guide contains information on intended programming interfaces that allow the customer to write programs to obtain the services of HATS WebFacing.

**Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml).

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
Readers’ Comments — We'd Like to Hear from You

IBM Host Access Transformation Services
Developing WebFacing Applications
Version 9 Release 5

Publication No. SC27-5905-01

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.
Submit your comments using one of these channels:
• Send your comments to the address on the reverse side of this form.
• Send a fax to the following number: 1-800-227-5088 (US and Canada)
• Send your comments via email to: USIB2HPD@VNET.IBM.COM

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

Email address
Readers’ Comments — We’d Like to Hear from You
SC27-5905-01

BUSINESS REPLY MAIL
FIRST-CLASS MAIL  PERMIT NO. 40  ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Rational Enterprise Modernization UAD
Department 67RA/Building 503
Research Triangle Park, NC
USA
27709-9990

SC27-5905-01