

Web Functions Wizard Guide

- [About this Guide](#)
- [Introduction to the Web Functions Wizard](#)
- [Wizard Home Page](#)
- [Customizing a Presentation Layout](#)
- [Customizing a Standard Menu](#)
- [Customizing a DHTML Menu](#)
- [Style Sheets](#)
- [System Wide Graphic Variables](#)
- [Customizing Browse Lists](#)
- [Application Import/Export](#)
- [Troubleshooting](#)
- [Wizard Tutorials](#)
- [Appendix A - Dynamic Buttons](#)

Edition date: November 12, 2013

© LANSA

About This Guide

Before you begin to use the *Web Functions Wizard*, you may wish to review the following:

- Who Should Use This Guide
- How to Use This Guide

Your feedback will help us improve the overall quality of the LANSAs documentation and training. Please email your comments to lansatraining@lansa.com.au.

Who Should Use This Guide

This guide has been created for Web Function developers who wish to use the Web Functions Wizard to customize their LANSAs Web Function or WEBEVENT applications. It has been designed to accommodate both novice and experienced LANSAs developers. For more details, please refer to [How to Use This Guide](#).

If you are a novice LANSAs Web Function developer, you should refer to *Web Functions and WEBEVENT Functions* of the LANSAs for the Web Guide for additional information about LANSAs processes, functions and Web technology.

How to Use This Guide

This guide has been designed for use as online documentation for the Web Functions Wizard. It includes information for installing, executing and troubleshooting the Wizard. A set of [Wizard Tutorials](#) has also been included to help you learn the fundamentals of using the Web Functions Wizard.

New Users

If this is your first time using the Wizard, it is strongly recommend that you use the [Wizard Tutorials](#) to help guide you through your first applications.

Experienced Web Developers

If you are an experienced Web Function developer, then you may already be familiar with the Web components and variables being manipulated by the Web Functions Wizard. You may find the Technically Speaking sections very helpful.

Technically Speaking Sections

There are Technically Speaking sections spread throughout this guide. You are not required to read these sections to execute the Web Functions Wizard. These sections provide technical details which may be of assistance to experienced LANSA developers who may wish to make manual alterations to the work created by the Wizard.

You need only read these sections if you are an experienced developer and want to understand how the Wizard has been implemented.

Tutorials

A set of Web Functions Wizard Tutorials have been included within this guide. The tutorials provide simple examples and instructions explaining the basic features of the Web Functions Wizard. Follow this link to the [Wizard Tutorials](#).



Look for this symbol for links to tutorials.

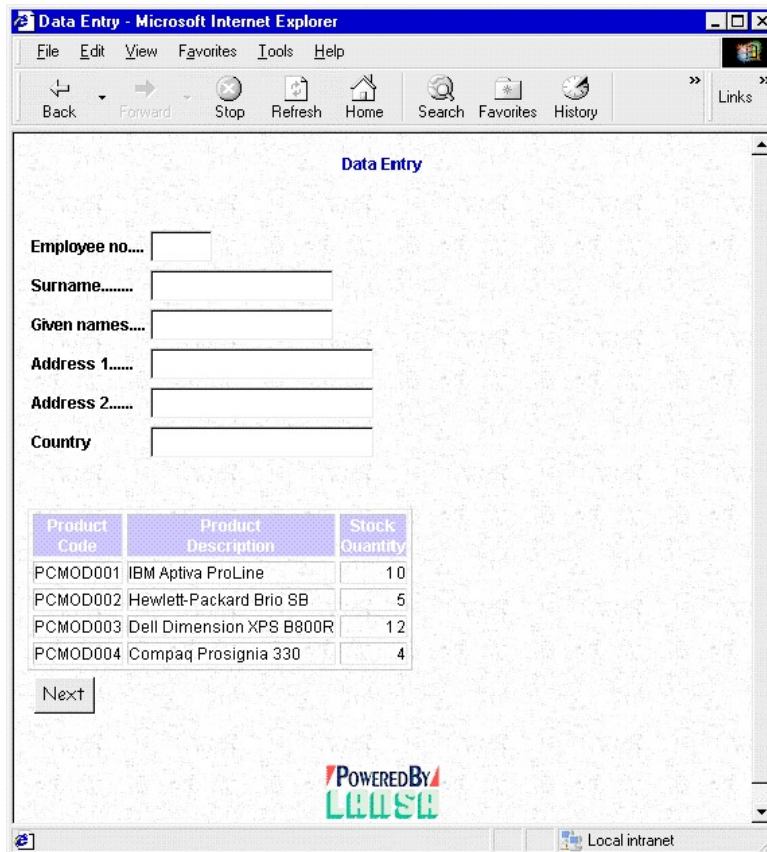
1. Introduction

The Web Functions Wizard allows you to quickly and easily customize the presentation of your LANSAs Web Function or WEBEVENT applications.

- It is a Web-based tool used to alter the graphical interface of your LANSAs Web Function Application. (It is not used with Web Application Modules.)
- It removes the need to manually define graphic variables and Web components.
- You can change the look-and-feel of your applications without having to recompile any of the Web functions in your application.
- The Wizard is not used to create your LANSAs Web functions.
- It allows the RDML programmer to concentrate on the functionality of the application, without having to worry about the graphical interface of the application. On completion of the RDML programming, the Wizard can be executed over the application to quickly build the menus and design a graphical interface to meet the user requirements.
- The Wizard gives you the flexibility of deciding which components to include or exclude from the presentation.
- The Wizard allows you to convert the default look-and-feel of a LANSAs Web function into a contemporary Web look-and-feel.

In-depth knowledge of HTML or JavaScript is not required to use the Web Functions Wizard.

For example, the screen shown is the default of applications generated by a LANSAs Web function. It has a 5250 look-and-feel.



Using the Web Functions Wizard, you can convert this 5250 look-and-feel to a modern, look-and-feel as shown. There is no manual modification to the HTML and no recompilation of the function.

Employee Details - Microsoft Internet Explorer

File Edit View Favorites Tools Help Links »

Update Details Delete Print

Employee Contract
Addnl Information ▶
Manager

LEOSA

Employee Details

Employee Number A1234

Surname

Given Names

Address 1

Address 2

Address 3

Skill Code	Date Sk Acquired	Grade	Comment
ADVPGM	1/11/01	D	
INTPGM	12/06/01	C	
MBA	15/07/00	D	Credit Average
MANAGE1	2/02/02	P	Internal Training

Next

Menu ready for use Internet

1.1 What can it do?

The Web Functions Wizard is a browser-based tool used to customize the presentation of your LANSAs Web function applications. It is an automated alternative to using the PC-based LANSAs Web utilities to modify the presentation variables and components. The Wizard is made up of LANSAs processes and functions in conjunction with HTML and JavaScript components.

The Web Functions Wizard allows you to:

- Define the boundaries of the application you want to customize.

It allows you to select a generic subset of processes to work with within the Wizard. The process drop down on the Home Page is modified to reflect your selection.

- Change the layout of your application.

A selection of layout templates provided can be used to define the background of your Web function application. You can change the layout style associated with a process by selecting from the schemas available with the Wizard. In addition, you can nominate the color scheme, the type and style of menus to be included and whether a banner, logo or click tracking should be incorporated in the interface.

- Customize Menu components.

The Wizard allows you to create Left, Right and Horizontal Menu components. DHTML and Standard Menus (buttons, hypertext, images) are supported. Each Menu Component may consist of many different menu items, which can be used to link the various functions or URL references.

- Define Style sheets.

The Wizard allows you to create a style sheet at a process level. Tags, Classes, Ids and Sub Classes can be built using a supported selection of font attributes. Additional details can be entered using the free-format text area.

- Change the presentation of your LANSAs browse lists.

Using the Wizard, you do not need to know the exact name of the graphic variable of the associated browse list attribute you want to change. All attributes of a browse list are presented to you in a single screen so that you

can quickly and easily identify what you want to change.

- Save changes in multiple languages.

The Wizard allows you to specify the language code used to save your customizations. This may be different to the language used to run the Wizard. Every time the Wizard is started, the language code is defaulted to the current language.

- Change or Delete LANSAs graphic variables.

For example, if you wanted to change the value of the graphic variable used to define the company logo, you would have to change the *LW3CPYLOGO variable using the PC-based Web Function Editor. In the Web Functions Wizard, the variable is described using a sentence-like and easy-to-understand description. You simply set the Company Logo option and then preview the image.

1.2 How does it work?

A LANSA Web Function or WEBEVENT applications can use LANSA's Web component technology and graphic variables to define the application interface or presentation. A LANSA developer can manually define these Web components and variables using the PC-based LANSA Web utilities. To do this, the developer must know the names of the specialized variables and must understand the details of the component architecture.

The Web Functions Wizard provides an alternative development interface for customizing your LANSA WEBEVENT applications. The Wizard is designed to be a much simpler and more productive developer interface. It removes the dependence on the developer's knowledge of HTML, JavaScript and the lower level technical architecture.

The Web Functions Wizard uses Web-based technology instead of the PC-based Web utilities. The Wizard is essentially a set of LANSA functions which generate the same Web components and variables a Web developer using the PC-based Web utilities might generate. The difference is the interface. The Wizard uses an intuitive interface to collect the information required to build the Web components and variables instead of requiring the developer to manually code HTML or JavaScript and have intimate knowledge of the LANSA WEBEVENT architecture.

The Wizard is used to customize the presentation and navigation used in your LANSA WEBEVENT application. Once you have built your LANSA processes and WEBEVENT functions, the Wizard can quickly transform the presentation layer. Again, there is no manual coding of HTML and your functions do not require recompilation.

Developers now have a choice of development interfaces.

Any work developed using the Wizard can also be customized manually using the PC-based utilities.

1.3 Running the Web Functions Wizard

The Web Functions Wizard uses a set of Web functions, which must be imported into your development partition. It can be installed in a multilingual or non-multilingual partition on an IBM i.

In Windows it can only be executed in a multilingual environment. For more details, refer to *Installing the Wizard* in the *Installing LANSA on Windows* guide.

You execute the Web Functions Wizard functions from your Internet browser using the same method used to execute any LANSA Web function.

To run Web Functions Wizard, use this URL:

```
http://<server domain name>/cgi-bin/lansaweb?procfun+wizard2+wiz02+<partition>+<language>
```

where

<server domain name> is the domain name or IP address of your Web server

<partition> is the LANSA partition where the Wizard is installed

<language> is the language you want to use to run the Wizard. By default, the Wizard runs in English. For example, you might enter this:

```
http://101.102.103.104/cgi-bin/lansaweb?  
procfun+wizard2+wiz02+DEM+ENG
```

If the Web Functions Wizard has been successfully installed, the Wizard Home page will be displayed.

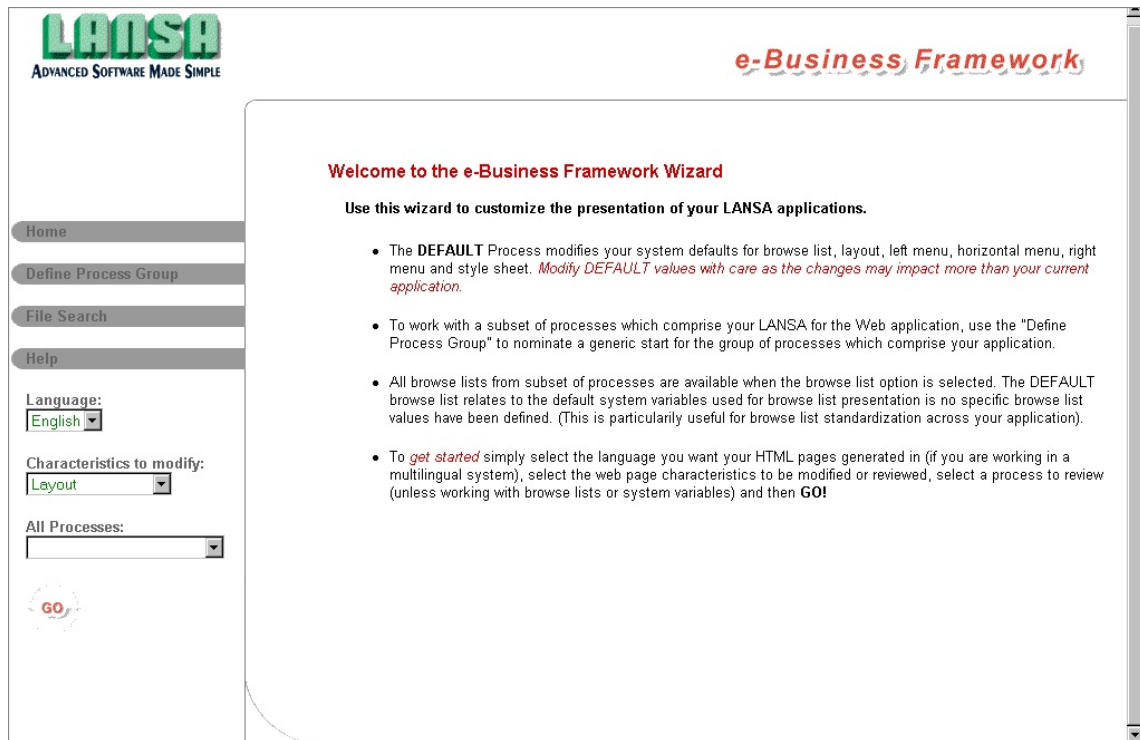
Internet Explorer, version 5.5 or greater is recommended.



[WIZ001 - Executing the Wizard](#)

1.3.1 Wizard Home Page

When you execute the Web Functions Wizard, the Home Page Menu will appear as follows:



From the Web Functions Wizard Home Page Menu, you can select:

- | | |
|----------------------------------|--|
| Home | Return to the Wizard Home Page. |
| Define Process Group | An optional step to nominate a generic subset of processes from which to work in your Wizard session. Refer to <i>Process Selection</i> below. |
| File Search | Search for a file (typically an image) to be used as a value for a variable. |
| Help | On-line access to the Web Functions Wizard documentation. |
| Language | Nominate the language components are to be generated in. (This is only displayed in multilingual partitions). |
| Characteristics to modify | Select the specific characteristic of the Web interface you want to modify. The characteristics available are:
Layout |

[Standard Menus](#)
[DHTML Menus](#)
[Style Sheets](#)
[System Graphic Variables](#)
[Browse Lists](#)

**Process
Selection**

The Web interface characteristics, with the exception of system variables and browse lists, require a specific process to be applied against. Refer to [Define Process Group](#).

1.3.2 Task Tracking

If your partition has task tracking enabled, you must specify a task identifier in order to use the Wizard. The task identifier will be associated with any objects you create or change. To specify a task to run your LANSAs function, you extend the URL syntax to include:

+TASK_ID+<task identifier>

For example, you might enter this:

**http://101.102.103.104/cgi-bin/lansaweb?
procfun+wizard2+wiz02_DEM+ENG+TASK_ID+MYTSK01**

1.3.3 Security

The Web Functions Wizard uses LANSAs Web functions. You do not require user profiles or passwords to execute these functions unless you have set up a higher level of security for your Web system. You may wish to add process level authentication to limit user access to the Wizard.

The security rules as described in the *Installing LANSAs on Windows* and the *Installing LANSAs on iSeries* guides apply to the Wizard processes and functions. If you do not have security defined for your LANSAs for the Web system, you may wish to set up process level authentication so that you have controlled access to the Wizard

To set up process level authentication, refer to the *Installing LANSAs on Windows Guide*.

You will want to secure this process:

WIZARD2

You can also use task tracking with the Wizard. If you have task tracking enabled, refer to [1.3.2 Task Tracking](#).

When you execute the Wizard, you will be creating LANSAs for the Web graphic variables which use LANSAs system variables. The owner of these system variables is based on the user profile used when executing the Wizard. If you wish to edit these system variables using the Web Function Editor, you must have authority to these variables. The owner may grant authority to the system variables. If you execute the Web Function Editor using the owner's profile, you may change these variables. The LANSAs for the Web PC-based utilities are secured by user profile and password.

If Web Server authentication has been set up, then you will enter a profile and password, and you should be able to determine the user profile being used to create the system variables. If no authentication has been configured, then anonymous user access is being used. You must determine the profile associated with the DFTUSR. For more details about user profile security, refer to the *Installing LANSAs on Windows Guide*.

1.3.4 Browser configuration

It is very important that you have your browser properly configured when using the Web Functions Wizard. You should not be using cached images when you customize your application as your browser will display old settings. You should set your browser to retrieve the most current page each time a page is accessed using the Wizard.

The Internet Explorer browser, version 5.5 or greater, is recommended for use.

To configure your Internet Explorer browser

1. Select the *Tools* menu and choose *Internet Options*.
2. Click on the *Settings* button on the General tab.
3. For the option *Check for newer versions of stored pages*, select the *Every visit to the page* radio button.
4. Press *OK* to save your changes.



[WIZ001 - Executing the Wizard](#)

1.3.5 Help Configuration

The Web Functions Wizard's *Help* button gives you access to the online Help. The directory nominated as the Help file location must be mapped to your Web server.

To simplify maintenance of your Web server configuration, include the help files in a sub-directory of the Images directory. For example, /images/lwizhelp.

IBM i Help

When you install or upgrade your LANSA system, the Wizard Help will be automatically installed in your LANSA system directory /WIZHELP when you load the Wizard.

If you modify the help location after installation you will need to reload the help files from the LANSA installation CD-ROM (save file LWIZHLP) to the appropriate directory.

Tip for IBM i Users

Load help files under the images directory to simplify your HTTP configuration.

Windows Help

On Windows, the help files will be automatically installed in your LANSA system directory /LWIZHELP.

Changing the Help file location

To change where your Help is accessed from, simply change the address by following this path in the Wizard:

1. From the Home page, select the *System Variables* characteristics then press Go.
2. When the Maintain System Variables page is presented modify the Default Help File Location and press the Apply button.

You can also set this address by changing graphic variable *LW3WIZHELP using the Web Function Editor. For the Wizard help to open correctly:

- Enter the appropriate directory path (e.g. \mywizardhelp)
- On IBM i, ensure the directory path is mapped in HTTP configuration
- On Windows, ensure the directory path is defined to your Web server.

The Wizard then selects the appropriate file to display from the nominated location.

For more details, refer to [System Wide Graphic Variables](#).

Reminder: Graphic variables are system wide in LANSa. If you change the help location, all developers using the Web Functions Wizard must have access to this location. If you change the help file location it is your responsibility to load the Help files to the nominated location.



[WIZ001 - Executing the Wizard](#)

1.3.6 Wizard navigation

The Wizard is a set of LANSAs for the Web functions designed to execute in an Internet Explorer 5.5 or higher browser.

Navigating around the Wizard is quite simple. It uses these controls:



Request Home at any time to return to the Wizard Home Page and modify your selection options.



Use the *Help* button to access the online help. You can change the Help files' location to access it from any nominated file directory. (Refer to [1.3.5 Help Configuration](#).)



Select the language, characteristics to modify and (optionally) process from the Home page before using the Go button to proceed with your selection.



Use the Go button to proceed to with your selection from the Home Page.



Use the appropriate TAB to review the associated information.



Different buttons are provided throughout the Wizard to execute the associated action. For example after selecting the Layout Style the Build button must be pressed to generate the Web Component.

When using the Wizard, note the following:

- The ENTER key is not supported in the Wizard.
- If you plan to use the Back key, be sure that your browser is properly configured as described in [1.3.4 Browser configuration](#).

- Process and System wide values use exactly the same set of processing screens in the Wizard. System level values are identified by process or browse list name DEFAULT. DEFAULT is a reserved value for both process and browse list names and should not be used in your RDML applications.

For more details, refer to [Wizard Home Page](#).

1.4 Wizard Terminology

When executing the Wizard, you will need to become familiar with these terms:

Process Group	A Process Group allows you to nominate a subset of processes to use when working with the Wizard. This effectively defines the boundaries of the Web application.
Current Process	When a process has been selected in the Wizard Left Menu, this is referred to as the current process. The current process is the process which will be modified by any subsequent actions within the Wizard.
Style sheet	LANSA Web functions allows you to use cascading style sheets (CSS) with your applications. Cascading style sheets allow you to describe a style that applies across one or more of your HTML pages. You can use CSS to define the presentation of an HTML page, including the font (face, size and color), background color, the positioning of elements within your HTML page as well as text decoration such as italics or underline.
Execution Language	The language the Wizard is running in. This is indicated as part of the URL used to execute the Wizard. The default language is used if no language is indicated. When running in a non-multilingual partition, this value is not required.
Development Language	The language components will be generated in. This is indicated as the Language in the Left Menu of the Wizard. When running in a non-multi-lingual partition this value is not displayed and all components will be generated as NAT (National) language.
Menu Component	A Web component used to display a menu in your application. There are two styles of menu – Standard or DHTML and three positional settings – left, right and horizontal.
DHTML Menu	DHTML Menus can include multi-level sub-menus. Only a single DHTML menu can be included in the process layout. DHTML menus can only be defined with a Left or Horizontal orientation.
Standard Menu	Standard menus can include a combination of images, buttons/tabs and hyperlinks. Left, right and horizontal Standard

Menus can be defined and used concurrently on a layout.

Left Menu Component	Either a Standard or DHTML menu which appears on the left side of the function. Items are listed down the menu.
Right Menu Component	A Standard menu which appears on the right side of the function. Items are listed down the menu.
Horizontal Menu Component	Either a Standard or DHTML menu which appears at the top of the function. On the first level of the menu items are listed across the menu.
Menu Item	A menu item is an option which appears on one of the left, right or horizontal menus.
Layout	This describes the overall presentation of the function including the background colors used, the style of the window and the menu components included in the presentation.
Variable	This refers to graphic variables used by browse lists, processes or as system wide defaults.

1.5 WEBEVENT versus Procedural Functions

The Web Functions Wizard has been specifically designed for use with Web applications written as WEBEVENT functions. WEBEVENT functions support the non-persistent connection state used by HTTP and the Internet, i.e. it supports the use of the Back key in the browser. For more details refer to [WEBEVENT Functions](#).

The presentation layouts are designed for easy user navigation. Using menu components in your layouts, you can quickly and easily link to any number of functions or Web sites. WEBEVENT functions are required to support this type of rapid and random user navigation.

If you choose to use the Web Functions Wizard with procedural Web functions, you must be very careful with menu components. Before using any menu components, you must fully understand the LANSAs for the Web transaction server and the execution architecture.

For example, procedural functions do not support the browser's Back key. With procedural functions, your functions should return to a process level menu before executing a new function or else your procedural functions can end abnormally.

The Web Functions Wizard cannot be used with Web Application Modules or WAMs.

1.6 Partition Wide Versus System Wide

It is important to understand how the changes made by the Wizard will impact your LANSAs for the Web application. Some changes impact the complete LANSAs system (i.e. all partitions) while other changes will only impact a single partition.

Graphic Variables are System Wide

Any change to a graphic variable will impact the entire LANSAs system.

Changes to browse list variables will impact all references to browse lists with the specific name in all processes and partitions. For example, many of the LANSAs templates use browse lists named LIST. If you make presentation changes to the specific browse list named LIST, then every function in every partition in your system that uses LIST will be impacted by the change.

If you are working with a subset of processes, only browse lists referenced by these processes will be listed. However, be aware that these browse list names may be used in other processes beyond the scope of your Web application.

Specific process variables are automatically created to store your layout and menu selections. These values will be used by all processes with the same name in your LANSAs system. For example, if you create a process named TEST01 in the DEM partition and you have a process named TEST01 in the PRD partition, both processes are impacted by your changes to the layout color scheme and features.

A pre-determined set of System Variables are used as the default value for all processes in your LANSAs system which do not have specific process variables associated with them.

The impact of changes is a very important consideration if you are exporting processes between partitions in the same or different LANSAs systems. For more details, refer to [Application Import/Export](#).

Layouts and Menu Components are Partition Wide

All presentation layouts and menu components (left, right or horizontal) are partition specific.

The layout and menus associated with process DEFAULT is used as the default layout for processes in your current partition. It is not used for the complete LANSAs system.

LANSAs process names must be unique in a partition. Consequently, any

specific process layout changes will only affect the selected process in the partition where you are working.

If you wish to use your layouts or menu components in another partition, they must be exported and imported into the target partition. Process specific layouts and components are automatically exported with the process definition. For more details, refer to [Application Import/Export](#).

1.7 Multilingual Versus Non-multilingual Objects

It is important to understand how the changes made by the Wizard will impact your LANSAs for the Web application. If you are implementing your application in multiple languages, some changes impact all languages while other changes will only impact a single language.

1.7.1 Graphic Variables are non-multilingual

Any change to a graphic variable will impact all languages in the entire LANSAsystem.

Changes to browse list variables will impact all references to browse lists with the specific name in all languages, processes and partitions.

Specific process variables are automatically created to store your layout and menu selections. All languages will use these values in your LANSAsystem. For example, if you create a standard left menu in language ENG and create a separate standard left menu in FRA, both menus are affected by changes to the menu color scheme when you change the layout. This is further complicated if you have the same process defined in multiple partitions. For more details refer to [1.6 Partition Wide Versus System Wide](#).

System Variables are used as the default value for all processes in your LANSAsystem which do not have specific process variables associated with them. They are not language specific.

For more details, refer to [Graphic Variables](#).

1.7.2 Layouts, Menu and Style Sheet Components are language specific

All presentation layouts, menus (standard or DHTML) and style sheet components are partition and language specific.

The Layout, Menus and Style Sheet associated with process DEFAULT are used as the default components for processes in your current partition. It is not used for the complete LANSAs system.

On IBM i, if you create a Layout, Menu or Style Sheet for a specific process in the LANSAs system default language, these components will be used as the default for the specific process regardless of language. Creating a language specific component will override the default language component.

Consequently, any specific process component changes made in the default language will affect the selected process in all languages in the partition where you are working.

On Windows, separate process level components must be created Layout, Menu or Style Sheets in each language used in your LANSAs application.

Consequently, any changes to process component must be applied to the corresponding component in each language (unless the change is language specific).

Changes to a process component in a language other than the partition default language will only affect the selected process for the selected language in the partition where you are working. For example, if your LANSAs system default language is ENG and you have a new process TEST01 which you want to define a French standard right menu for, LANSAs will look for the following menu components in hierarchical order:

TEST01_RMENU (FRA)

TEST01_RMENU (ENG)

DEFAULT_RMENU (FRA)

DEFAULT_RMENU (ENG)

1.7.3 Language Recommendations

If you are developing a Web Application interface in multiple languages, it is recommended to use a common "look-and feel" in all languages. This approach will reduce your maintenance effort significantly.

Start by using the Wizard to define your presentation interface in the default language. When you are satisfied with the presentation interface in the default language (including layout, menus, colors scheme, menu features) run your application in an alternate language. Take note of the text and images requiring translation. The only Wizard generated objects which include text are the Standard and DHTML menus. Next, generate the menus in your alternate language based on the default language. No other components associated with the presentation interface need to be language specific.

For more details, refer to [Customizing a Standard Menu - Multilingual Considerations](#) and [Customizing a DHTML Menu - Multilingual Considerations](#).

1.8 Considerations for Using the Wizard

Following are some important considerations when using the Web Functions Wizard:

- The Web Functions Wizard V9.1 and V10.0 can be run concurrently in a LANSAs partition.
- The Wizard must be installed into each partition in which you plan to develop with the Wizard. It does not have to be installed into a partition used only for executing finished LANSAs for the Web applications.
- Graphic variable changes made using the Web Functions Wizard will immediately impact your applications. Menu component and presentation layout changes made using the Web Functions Wizard will only impact functions which were compiled after the LANSAs for the Web system was upgraded for the Web Functions Wizard. For more details, refer to [Existing LANSAs for the Web System earlier than V8.0 -EPC538](#).
- If task tracking is enabled in your partition, you will need to use the TASK_ID parameter when running the Wizard.
- The Wizard uses WEBEVENT functions. If you are using the Back key, be sure to check your Browser Configuration.
- You may wish to add process level authentication to provide secured access to the Wizard.
- The Wizard has been designed for use with WEBEVENT functions. The Wizard can also be used with procedural functions but you must be very careful with menu components.
- It has been designed as a multilingual application with English as the default language.
- All images shipped with the Wizard exclude text and, as such, can be used by all languages in multilingual applications.
- Some HTML components shipped with LANSAs include a meta tag reference to charset 8859_1. This may need to be modified, depending on your installation. For details, refer to [Modifying charset for non-English Systems](#) in the *LANSAs for the Web Guide*.
- Before changing the System Wide Default Layouts, be sure that you have read the [1.9 Impact on Existing Applications](#).
- It is very important to understand the impact of the changes made by the

Wizard. For more details, refer to [1.9 Impact on Existing Applications](#).

- If you wish to move your application to a new partition or a new LANSA system, you must understand how the Wizard impacts your system. For more details, refer to [Application Import/Export](#).
- Extended browse list support must be turned on in the partition where you are running the Web Functions Wizard.
- The Web Functions Wizard cannot be used with Web Application Modules or WAMs.

1.9 Impact on Existing Applications

The impact of the Web Functions Wizard on your LANSAs Web applications will depend upon your system installation.

As appropriate, refer to:

[1.9.1 New LANSAs for Web Installation](#)

[1.9.2 Existing LANSAs for Web Installation](#)

1.9.1 New LANSAs for Web Installation

If this is your first LANSAs for the Web system, the Web Functions Wizard can be applied to all your new processes and functions provided you have installed the Wizard in the partition.

To install the Web Functions Wizard with a new LANSAs installation, Refer to the Installing LANSAs on iSeries Guide or the Installing LANSAs on Windows Guide for how to install the you select the Web Functions Wizard either into a new LANSAs installation or a new partition option in *Partition Initialization* during the *Custom Install* process. Refer to Installing LANSAs on iSeries Guide for details.

If you have created a new Partition, you will need to import the Web Functions Wizard from a save file as described in the *Installing LANSAs on iSeries Guide*.

If you import an application created from a LANSAs system that did not have the Web Functions Wizard installed, refer to [1.9.2 Existing LANSAs for Web Installation](#).

1.9.2 Existing LANSA for Web Installation

Existing LANSA for the Web System earlier than V10.0

If you have LANSA Web Applications with an interface generated by the Web Functions Wizard before Version 10.0, changes made with the Web Functions Wizard will have some impact on your existing Web functions.

The Web Functions Wizard V10.0 includes new layout schemas and has modified the menu build process significantly.

If you have a pre V10.0 Web Application built using the Wizard and want to make minor changes to the presentation interface, continue to use the Web Functions Wizard V9.0 or V9.1.

If you have a pre V10.0 Web Application built using the Wizard and want to overhaul the interface, consider using the Web Functions Wizard V10.0 for a new "look-and-feel" for your Web Application. In this case, existing menus will be automatically converted to the Wizard V10.0 format but may need to be reviewed to select new button styles and so on. Menus must then be rebuilt. If you rebuild your menus using the Web Functions Wizard, you must also rebuild your Layout, and vice versa. For more details, refer to [1.9.3 Converting a Web Application from Wizard V9.1 to V10.0](#).

Once you run the Wizard V10.0 over your application, you will no longer be able to view all the menu definitions in Wizard V9.0 or V9.1, if you choose to revert to that version of the software.

Most of the graphic variables used in the Wizard V9.0 and V9.1 are also used in the Wizard V10.0.

Existing LANSA for the Web System earlier than V8.0 -EPC538

If you have LANSA Web applications created before Release 8.0 and EPC538 were installed, changes made with the Web Functions Wizard will have some impact on your existing functions.

Graphic variable changes made using the Web Functions Wizard will immediately impact your existing applications without any recompilation. The Wizard simply provides an alternate interface for changing the graphic variables you have used in your functions. Any changes made to specific browse list variables, process specific variables or system wide variables will immediately impact your Web functions.

Menu component and presentation layout changes made using the Web Functions Wizard will only impact functions which are compiled or recompiled

after the LANSAs for the Web system was upgraded for the Web Functions Wizard. The HTML pages for your functions must include a new set of RDML tags such as <RDML LAYOUT> in order to use the Wizard changes. These tags will not exist in your older functions until they have been recompiled.

When you recompile an older function, it may be updated to use the new layouts and components if you have specified process specific or system wide default layout changes.

If you are recompiling existing functions in a partition using the Wizard and you do not wish to have the Wizard change the presentation layout (colors, menu components, etc.) of your existing function, then you have several options:

1. Do not create any system wide default layouts. The RDML LAYOUT tags are still used in your HTML pages when recompiled, but no components will have been defined. Your existing functions will continue to execute without change.
2. Do not create any process specific layouts for a process which already exists, i.e., do not create a layout for the process unless you want it to adopt the new layouts.
3. If you have created system wide default layouts but do not want them to impact specific processes, then you can create a process specific Web skeleton which does not include the RDML LAYOUT and other new RDML tags. For more technical details, refer to [Customizing a Presentation Layout – Technically Speaking](#).

When you import any functions into your system, you should be aware of these points so that you can properly handle the impact of the Wizard on your application.

On IBM i, If you have created your own skeletons in file DC@W22 and you wish to use the Web Functions Wizard for these existing functions, you may need to modify your skeletons according to the changes made in the default shipped skeleton (WEBSKEL in file DC@F28 in the LANSAs data library).

1.9.3 Converting a Web Application from Wizard V9.1 to V10.0

If you have used an earlier version of the Web Functions Wizard to customize a LANSa Web Function Application you can:

- Continue to use and maintain your Web Function Application with its current interface
This is the recommended approach if do not want to change the "look and feel" of your Web Function Application. In this case, continue to use the Wizard V9.1 to modify menus.
- Modify your existing interface using the Web Functions Wizard V10.0
To rejuvenate your Web Function Application by adding a new "look and feel" consider using the Wizard V10.0 to upgrade the interface. How to do this is described in the simple steps following.

Warning: You cannot mix and match Wizard V10.0 features with features from an earlier release of the Wizard. For example, you cannot elect to use a Wizard V10.0 layout with Wizard V9.1 menu styles.

To convert an interface created using a previous version of the Web Functions Wizard to Wizard V10.0:

1. Use the Define Process Group to nominate the set of processes to be modified (if your Web Function Application uses generic naming for processes).
2. Select the primary process used in the Web Function Application and build a new layout using the Wizard V10.0 (including Color Scheme and Layout Features). Ensure your menu selection matches your new design requirements. Any pre-existing menu definitions will be converted as Standard Menus.
3. Select Menus from the Characteristics to Modify drop down list, and review the menu definition which has been automatically converted from the Wizard 9.1 definitions. If you intend to use the *Dynamic Buttons* option to display menu items, review the Menu Features and select a default dynamic button style and color to be used for your process.

Technical Note: In previous versions of the LANSa Web Functions Wizard, the menu definition were included as a comment at the beginning of the associated menu HTML page. In Version 10.0, these definitions are removed

from the HTML page and are stored in files DC@W31 and DC@W32. Once the menu is re-built using the V10.0 of the Wizard, the menu definitions will no longer be available for modification.

4. DHTML menus were not supported prior to the Wizard V10.0 so there is no earlier information available to be converted.
5. If you manually created a process level style sheet, the definition will be automatically interpreted by the Wizard V10.0. You can continue to manually maintain your style sheet or use the Wizard V10.0 to maintain the style sheet definition.
6. Browse List information is handled the same way in the Wizard V10.0 as in earlier versions of the Wizard. No conversion is required to use the Wizard V10.0.
7. Adopted processes do not need to be modified UNLESS you want to remove the process level variables associated with an adopting process (This feature was not available in earlier versions of the Wizard).

Warning: Once you have converted any pre-existing menus to use the LANSA Web Functions Wizard V10.0, you will no longer have access to the menu definitions from earlier versions of the Wizard. This will only take effect after you have re-built the menus using the Wizard V10.0.

2. Wizard Home Page

When you execute the Web Functions Wizard (refer to [Running the Web Functions Wizard](#)) you will be presented with the Wizard Home Page as your starting point. The Wizard Home Page is the main menu for the Wizard where you return to in order to modify your selections.

The following actions can be initiated from the Wizard Home Page:

[2.1 Home](#) [2.2 Define Process Group](#) [2.3 File Search](#)

[2.4 Language](#) [2.5 Characteristics to Modify](#) [2.6 Process Selection](#)

[2.7 Go](#)

LANSa
ADVANCED SOFTWARE MADE SIMPLE

e-Business Framework

Welcome to the e-Business Framework Wizard

Use this wizard to customize the presentation of your LANSa applications.

- The **DEFAULT** Process modifies your system defaults for browse list, layout, left menu, horizontal menu, right menu and style sheet. *Modify DEFAULT values with care as the changes may impact more than your current application.*
- To work with a subset of processes which comprise your LANSa for the Web application, use the "Define Process Group" to nominate a generic start for the group of processes which comprise your application.
- All browse lists from subset of processes are available when the browse list option is selected. The **DEFAULT** browse list relates to the default system variables used for browse list presentation is no specific browse list values have been defined. (This is particularly useful for browse list standardization across your application).
- To *get started* simply select the language you want your HTML pages generated in (if you are working in a multilingual system), select the web page characteristics to be modified or reviewed, select a process to review (unless working with browse lists or system variables) and then **GO!**

2.1 Home

A rectangular button with a light gray background and a darker gray border. The word "Home" is written in a dark gray, sans-serif font in the center of the button.

Selecting Home will return you to the Wizard Home Page from wherever you are currently working in the Wizard.

- If you do not select *Apply*, when relevant, before selecting *Home* all your changes will be lost.
- On return to the Home page on an IBM i, your current language and process settings remain the same but your Characteristics to Modify selection will be reset.

2.2 Define Process Group

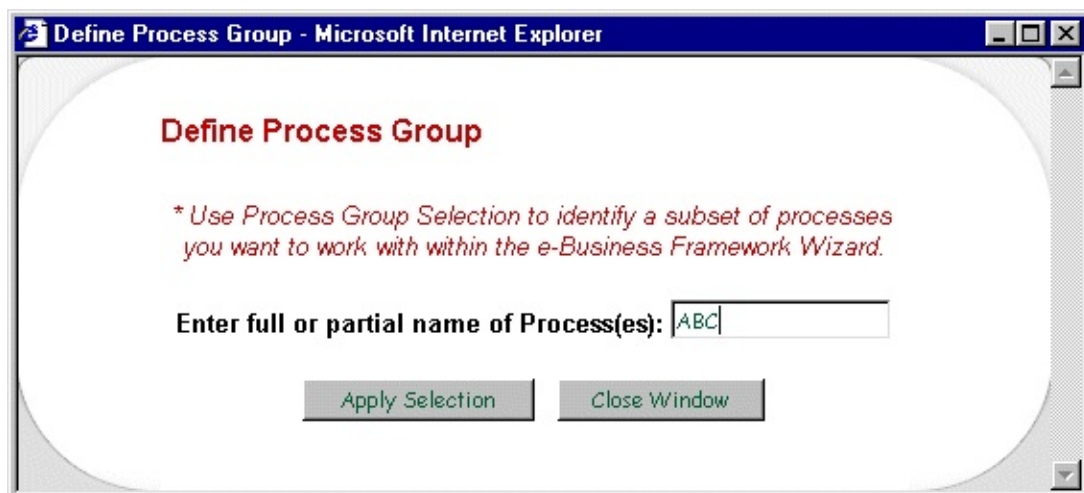
The Define Process Group option allows you to nominate a subset of processes to use when working with the Wizard. It effectively defines the boundaries of the Web function application you are customizing.

- Selecting the Define Process Group option will present a window prompting you to enter a generic prefix to identify the group of processes you wish to work with within the Wizard. For example, if your web application consisted of the web-enabled processes ABCMNT01, ABCINQ01 and ABCWND01, you would enter a partial process name of "ABC".
- After pressing the Apply Selection button, you return to the Wizard Home Page where a modified list of processes beginning with "ABC" would appear in the drop down. All process-based actions within the Wizard will then use this subset of processes.

Only processes defined in the development [2.4 Language](#) will be available for selection.

- The Process Group will always include the entry DEFAULT as the last entry. DEFAULT is associated with the default components for your LANSA partition.
- Your Process Group selection can be modified at any time, initiating the Define Process Group window.
- To reset the Process Group to display all processes in the partition, enter a blank process name.

For more details, refer to [2.6 Process Selection](#).



Define Process Group - Microsoft Internet Explorer

Define Process Group

** Use Process Group Selection to identify a subset of processes you want to work with within the e-Business Framework Wizard.*

Enter full or partial name of Process(es):

2.3 File Search

The File Search is a generic facility to browse a directory and select a file.

A file selection will return a fully qualified file name. To use this feature effectively, the Image Location, must be defined as a mapped drive. (Image Location is configured in LANSA for the Web using the Web Administrator.) The desired file name will be selected from the directory corresponding to the virtual directory "\images" and then the required extension beyond the "\images" directory can be cut and pasted into the variables to be modified.

For example:

For an installation you might map the image directory used by the Web Server to *c:\InetPub\wwwroot\images*. You could then use the file search to locate the file *bullet.gif* in a sub-directory under the "\images" directory. To select this file which would return the file name:

```
c:\InetPub\wwwroot\images\
```

To use this image as the browse list selection image, you would then need to copy the \

The Web Functions Wizard typically uses files references for images. Files can be nominated for:

- Browse list selection image
- A menu item separator image
- Background images
- Company logo
- LANSA messages image
- Help files location (this is an .htm file).

For more details, refer to [Images for Browse List Selection](#).

File Search

** Use File Search to browse the IFS and return a fully qualified file name.
Then, cut and paste the appropriate sub-directory and file name into the Wizard as required.
To use this feature effectively it is important to know where your /IMAGES/ directory is mapped on your IFS.*

Qualified File

Name:

2.4 Language

If you are running the Web Functions Wizard in a multilingual partition, you can nominate the language in which Web components are to be generated and saved. This development language may be different to the execution language specified on the Web Functions Wizard URL.

On an IBM i, the language nominated on the URL will be used as the development language by default. If no language is included on the URL, then the system default language is used.

On Windows, the first entry on the language drop down will be used as the development language by default.

The language nominated may impact your Process's drop down. Only processes defined in the development language will be available for selection. If you change your development language, the current process will still be available to use. When switching between languages, it is advisable to restart the Wizard from the URL to ensure the correct Process drop down is presented. For more details, refer to [2.6 Process Selection](#).

For more information on generating components for multiple languages, refer to [Customizing a Standard Menu](#) and [Customizing a DHTML Menu](#). The Language selected can only be modified on the Wizard Home page.



2.5 Characteristics to Modify

To modify your Web Function Application using the Wizard, you must select the characteristics to be changed. These are the characteristics that can be selected from the drop down:

- Layout
- Menus
- DHTML Menus
- Style sheet
- System Variables
- Browse lists

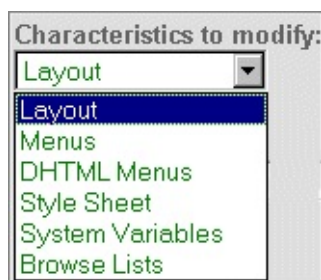
The Characteristics to Modify is used in conjunction with the Process drop down to determine the components to be modified by the Wizard.

The characteristics Layout, Menus, DHTML Menus and Style sheet all modify components associated with a specific process (or DEFAULT for system level values). For more details, refer to [2.6 Process Selection](#) and [Partition Wide versus System Wide](#).

The Browse Lists characteristic is based on the browse lists defined in all the Processes currently listed in the Process drop down. Any selection of a process will be ignored. Refer to [Customizing Browse Lists](#) for additional information.

The System Variables characteristic does not require a process to be nominated as all associated changes impact the current LANSa system. For more details, refer to [Partition Wide versus System Wide](#).

The Characteristic to modify can only be selected on the Wizard Home page.



2.6 Process Selection

The Process drop down allows you to select a single process as your current process.

When you execute the Web Functions Wizard, the Process drop down automatically includes all web-enabled processes in the partition nominated in your URL, defined for the current development language. This selection is indicated on the Wizard Home page by a heading "All Processes" associated with the process drop down.

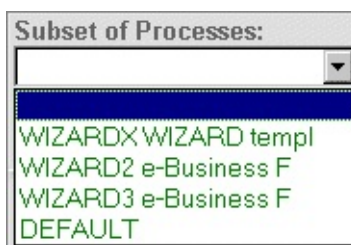
The available processes can be modified using the [2.2 Define Process Group](#) option. If you nominate a generic selection of processes using Define Process Group the text associated with the process dropdown list on the Home page will be changed to Subset of Processes to reflect this new grouping.

Process DEFAULT is always included as the last entry in the process drop down. The components associated with the DEFAULT process are reserved as the partition default components. Refer to [Partition Wide versus System Wide](#) for additional information.

The Process drop down is used in conjunction with the [2.5 Characteristics to Modify](#) to determine the components to be modified by the Wizard.

The Process selected can only be modified on the Wizard Home page.

If you change your development [2.4 Language](#) the current process will still be available to use. When switching between languages, it is advisable to restart the Wizard from the URL to ensure the correct process drop down is presented.



2.7 Go

After selecting your Language, Characteristics to Modify and Process (if required), use Go to execute your Wizard selection.

The ENTER key is not supported in the Wizard.

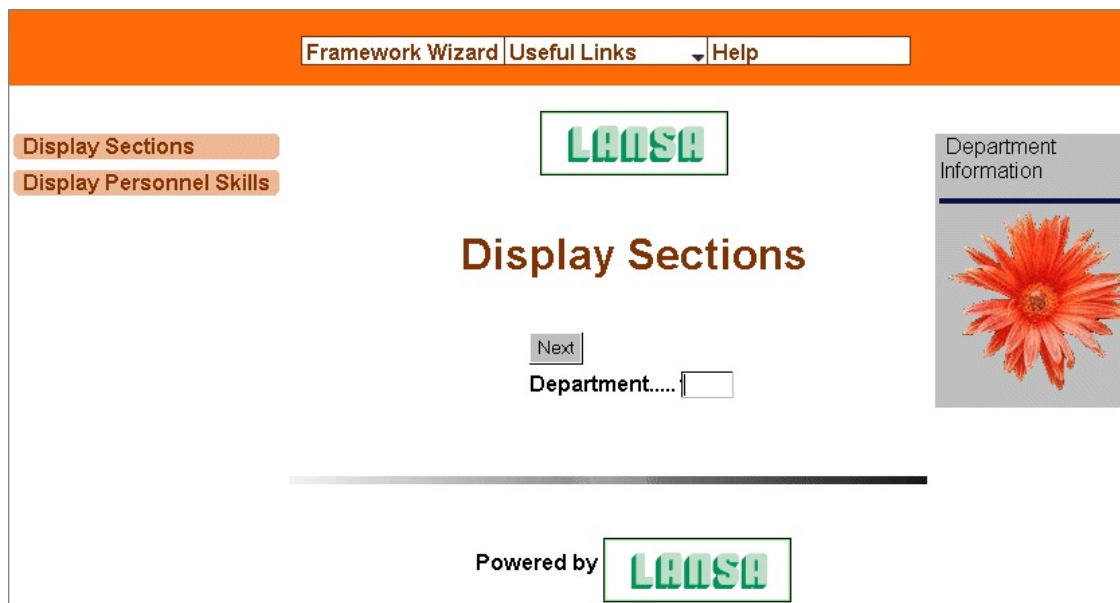
For more details, refer to [2.4 Language](#), [2.5 Characteristics to Modify](#) and [2.6 Process Selection](#).



3. Customizing a Presentation Layout

The Web Functions Wizard allows you to quickly create a professional presentation layout for your Web Function Application. This is achieved in three steps by:

- selecting a layout schema,
- customizing the color scheme, and
- selecting your layout features.



Any modifications to the layout presentation apply to the process nominated as the current process on the left hand menu bar. All presentation layouts are partition specific.

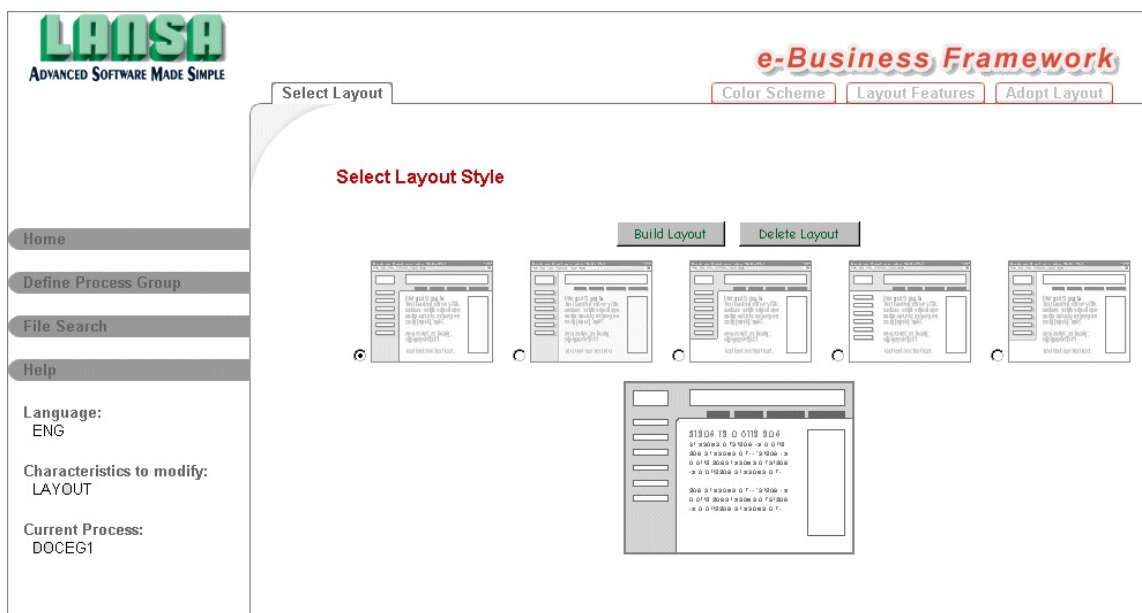
The layout customization applies to all the LANSAs functions defined in the current process. In most cases, these functions should be WEBEVENT functions. For more details, refer to [WEBEVENT versus Procedural Functions](#).

The layout page associated with the current process can be modified using either the Select Layout or Adopt Layout tabs. The [3.2 Color Scheme](#) and [3.3 Layout Features](#) are interpretive settings used in conjunction with the [3.1 Select Layout](#) and [3.4 Adopt Layout](#) options.

Before changing the System Wide Default Layout (current process DEFAULT) be sure that you have read [Impact on Existing Applications](#).

3.1 Select Layout

After you select the Layout characteristic and press Go from the [Wizard Home Page](#) you will be presented with the following.



- The Wizard allows you to choose from five presentation layout schemas.
- If you move your mouse cursor over the presentation layouts, you will see an enlargement of the schema.
- Select one layout to use as the schema for your process, and then press the *Build* button.

The Build process will incorporate your current Layout Feature settings. Any modifications to the [3.3 Layout Features](#) for this process will require the Layout to be re-built. However, [3.2 Color Scheme](#) details can be modified at any time without rebuilding your layout.

The layout selection will only impact the functions defined in the current process. (Unless the current process has been adopted by other processes. For more details, refer to [3.4 Adopt Layout](#).)

Any previous layout selection for the current process will be pre-selected. If there is no pre-selected layout, this is the first time the Wizard has generated a layout for the current process.

- It is recommended to open a second browser session to execute one function

in your current process. Use the Wizard in your first browser session to make iterative modifications to your layout, and then refresh your function URL in the second browse window to preview the results.

When you are satisfied with the layout presentation, you may return to the Wizard Home page to make additional selections to customize the process interface.

- The Delete Layout button will remove the layout component for the current process.

For additional information, refer to [3.2.2 Technically Speaking](#) .



[WIZ03 - Customizing a Presentation Layout](#)

3.1.1 Tips & Techniques

Review your current layout features before building your layout selection to ensure they suit your current requirements.

3.2 Color Scheme

Once you have selected the *Color Scheme* tab the color specific variables defined for the current process are shown.

The screenshot shows the 'Select Color Scheme' configuration page within the LAOSA e-Business Framework. The page has a left sidebar with navigation links: Home, Define Process Group, File Search, and Help. Below these are settings for Language (ENG), Characteristics to modify (LAYOUT), and Current Process (DOCEG1). The main content area is titled 'Select Color Scheme' and includes 'Apply' and 'Delete' buttons. A note states: '* Layout does NOT need to be rebuilt for any changes to take effect'. The configuration is organized into three sections: 1. 'Background Color' with a text input for 'Background color', a 'Color' button, and a 'Color Picker' button. 2. 'Background Image /IMAGES/' with a text input and a 'Preview' button. 3. 'Left / Horizontal Menu Background Color' with a 'Wizard supported colors' dropdown menu (currently showing 'Orange'), a 'Select your own color' text input, and 'Color' and 'Color Picker' buttons. 4. 'Right Menu Background Color' with a 'Right Menu Background' text input, a 'Color' button, and a 'Color Picker' button.

Colors can be selected using the [Color Picker](#) button to activate a "point-and-click" color palette of 216 browser-safe colors and a selected palette of grays. Alternatively, you can manually enter color values by name (e.g. red) or by using their hex code values (e.g. #FF0000).

The [Color button](#) will display the color you have specified in the related field. The *Background Color* is used for the background of the function area. *Any Background Image specified will be repeated in the background of the function area.*

You can select a Left/Horizontal Menu Color from the Wizard Supported Colors drop down or you can manually enter a color value. Wizard Supported Colors are particularly relevant in relation to the first, third and fifth layout schemes. When selected in combination with one of these schemas, the resulting layout will incorporate curved corners to delimit the left/horizontal menu area. If you manually select a color, these schemas will have squared corners. In addition, the Wizard Supported Colors were used to develop the menu buttons supplied with the Web Functions Wizard.

The Right Menu *Background Color* is used for the background of the right menu.

You do not have to configure all the variables. If the color variables are left

blank, or if they are changed to a blank value, they will adopt the system wide definitions or the default values assumed in LANSAs for the Web.

The [Image](#) button will check if the specified image exists in the image location configured for LANSAs for the Web and it will display the image.

Once you have specified an image for the *Background Image*, you should always enter an image name. If this value is changed to blanks after specifying a valid image, you will see an image not found error in your Web page. If you do not want the process specific image used, you must manually delete the *LW3PBGI_<process name> graphic variable.

Remember, the Wizard does not validate your entries. Use the *Color and Image* buttons to verify your selection.

After you have finished specifying your color scheme, press the *Apply* button to update the variables. A completion message "Layout Color Scheme has been successfully updated" will be issued.

Changes to the color scheme are applied dynamically (i.e. you do not have to rebuild the layout to view changes to the color scheme).

For more details, refer to [3.2.1 Tips & Techniques](#) or [3.2.2 Technically Speaking](#)



WIZ03 - Customizing a Presentation Layout

3.2.1 Tips & Techniques

- When a color graphic variable is set to blank, LANSA for the Web will ignore this value and look for another default. A blank color is treated as if the variable does not exist.
- To set the background image back to the system default, do not blank out the image file name. You must manually delete the *LW3PBGI_<process name> graphic variable using the Web Function Editor.
- If you use a Wizard Supported Color for the Left/Horizontal Menu Color in combination with selecting layout schema one, three or five it is highly recommended that the background color be set to white (#FFFFFF) for the best results.
- If you change the Left/Horizontal Menu Color from a Wizard Supported Color to a manually entered color, or vice-versa, the layout should be re-built to remove any references to the graphics used as the corner images. This only applies if you have selected layout schema one, three or five.

3.2.2 Technically Speaking

You are specifying these values for graphic variables:

- Background Color *LW3PBGC_<process name> is used over *LW3CLNTCOLOR
- Left/Horizontal Menu background Color *LW3PCLR1_<process name> is used over *LW3COLOR1
- Right Menu Background Color *LW3RMBG_<process name> is used over *LW3RMENUBKGND
- Background Image *LW3PBG_<process name> is used over *LW3CLNTBKGND

where <process name> is the name you specified for specific process.

For more details, refer to [Process Level Graphic Variables](#).

3.3 Layout Features

You can customize your selected presentation layout by including or excluding different layout features. These features, with the exception of Click Tracking, will impact the look-and-feel of your resulting layout.

The screenshot shows the 'Select Layout Features' configuration page within the LAOSA e-Business Framework. The page has a header with the LAOSA logo (Advanced Software Made Simple) and the e-Business Framework title. Navigation tabs include 'Select Layout', 'Color Scheme', 'Layout Features' (which is active), and 'Adopt Layout'. On the left, there is a sidebar with navigation links: Home, Define Process Group, File Search, and Help. Below these links, it shows 'Language: ENG', 'Characteristics to modify: LAYOUT', and 'Current Process: DOCEG1'. The main content area is titled 'Select Layout Features' and contains an 'Apply' button. A note states: '* Layout must be rebuilt for any changes to take effect'. The configuration options are as follows:

- Include Left Menu
 - Standard Menu
 - DHTML Menu
- Include Horizontal Menu
 - Standard Menu
 - DHTML Menu
- Include Right Menu
- Include Company Logo
- Include Banner
- Include Click Tracking

At the bottom, there is a link: Check / Uncheck All

Choose the features to include in your layout by selecting the appropriate checkbox. Then press the *Apply* button.

The features available for your presentation layout include:

- **Left Menu** The Left Menu component can be either a standard or DHTML menu and is shown at the left-hand side of the selected presentation schema. Typically, the Left Menu component contains menu items that are defined to link the various function modules in your application.
- **Horizontal Menu**
The Horizontal Menu component can be either a standard or DHTML menu and is shown at the top of the presentation aligned horizontally. Typically, the Horizontal Menu component acts as the toolbar for the application.
- **Right Menu**
The Right Menu component can only be defined as a standard menu and is shown at the right-hand side. Typically, it is used for promotional materials.
- **Banner**
For more details, refer to [3.6.4 Banner Components](#).
- **Company Logo**

If you want to change the value of the graphic variable used to define the company logo, you would have to change the *LW3CPYLOGO variable. This variable can be modified using the System Variables characteristic in the Web Functions Wizard or by modifying the graphic variable *LW3CPYLOGO using the PC-based Web Function Editor. For details, refer to [3.6.3 System Variables](#) .

- **Click Tracking**

Including Click Tracking will embed the RDML command <RDML TRACE> in your presentation layout. You must configure your LANSa system to use Event Logging for this feature to be used. For more details, refer to [Enabling Event Logging](#).

When a menu is included in a presentation layout, LANSa will search for a process specific Menu component. If a process specific component cannot be found, the default Menu component will be used if it exists.

Defining the menus is a separate step. For more details, refer to [Customizing a Standard Menu](#) and [Customizing a DHTML Menu](#).



[WIZ04 - Defining a Standard Menu Component](#)

[WIZ05 - Defining a DHTML Menu Component](#)

[WIZ03 - Customizing a Presentation Layout](#)

3.3.1 Tips & Techniques

If you are changing an existing layout and decide not to include an existing menu, the associated menu component is not deleted. The component simply does not appear.

- For example, you may define both a DHTML Left Menu and a Standard Left Menu to prototype the two different options. Only one of these menus can be included in the presentation layout at any given time, but the alternate menu component remains defined and is available to be included by updating the Layout Features and then re-building your presentation layout.

3.3.2 Technically Speaking

You are specifying a single graphic variable:

`*LW3PLIN_<process name>` is used over `*LW3LINCLUDE`

where `<process name>` is the name you specified for specific process.

Each character of this graphic variable is a flag used to indicate if a specific feature should be included or excluded when building the layout page. A value of 'Y' indicates the feature should be included.

***LW3PLIN__<process name> Character Set to Y to include:**

1	Left Menu
2	Horizontal Menu
3	Right Menu
4	Banner
5	Trace
6	Company Logo
7	Standard Left Menu
8	DHTML Left Menu
9	Standard Horizontal Menu
10	DHTML Horizontal Menu

3.4 Adopt Layout

You can share the presentation layout defined for another process by adopting the process layout. This feature is extremely useful if your Web application has multiple LANSAs processes. It allows you to customize the presentation layout for a single process and let the other processes adopt the presentation from this process to provide a consistent presentation for your complete application. Most changes performed against the original process layout will be automatically available in other processes that have adopted this layout.

The screenshot shows the LANSAs e-Business Framework interface. The top left features the LANSAs logo with the tagline 'ADVANCED SOFTWARE MADE SIMPLE'. The top right displays 'e-Business Framework' and navigation buttons for 'Color Scheme', 'Layout Features', and 'Adopt Layout'. A 'Select Layout' button is also visible. The main content area is titled 'Adopt a Layout' and includes an 'Adopt Process' button. A note states: '* Layout, Layout Features and Left/Horizontal Menu Color are always adopted'. Below this is a dropdown menu labeled 'Select Process to Adopt from:' with 'AAPROC01 Multilingual testing' selected. Three checkboxes are present: 'Inherit Background Image', 'Inherit Background Color', and 'Inherit Right Menu Background Color'. At the bottom, there is a 'Check / Uncheck All' checkbox. The left sidebar contains navigation links: Home, Define Process Group, File Search, and Help. It also shows 'Language: ENG', 'Characteristics to modify: LAYOUT', and 'Current Process: DOCEG1'.

To adopt a layout, use the dropdown list to select the name of the process to adopt from.

In addition, there are three checkboxes displayed on the page to assist in the customization of your Web function application. Use these checkboxes to delete the associated graphic variables from the current process. This feature ensures the values used for Background Image, Background Color and Right Menu Background Color are consistent with the values used in the adopted process (if desired). The Left/Horizontal Menu Color is handled as an exception. (For more details, refer to [3.4.2 Technically Speaking](#).)

After you have specified the process in the *Select process to Adopt from* box and you have selected any appropriate checkboxes, press the Adopt Process button to continue.

Following is the hierarchy of what is adopted:

- **Layout Schema** The layout schema is always the same as the adopted process. If the adopted process does not have a specific layout defined, then

DEFAULT_LAYOUT page will be used by both processes.

- **Color Scheme**

Your process will adopt the color scheme associated with the parent process. You can override these values by defining process specific variables. The Left/Horizontal Menu Color is handled as an exception. (For more details, refer to [3.4.2 Technically Speaking](#).)

- **Layout Features**

As the layout features are embedded in the <process name>_LAYOUT page, they are always inherited from the adopted process. For more details, refer to [3.3 Layout Features](#).

- **Standard Menus**

Standard Menus will be inherited from the adopted process. You can override these menus by defining a corresponding menu for the current process. For more details, refer to [Customizing a Standard Menu](#).

- **DHTML Menus**

If your adopted process includes a DHTML menu, this DHTML menu will always be used by the adopting menu. If you define a process specific DHTML menu for the adopting process this will be ignored.

- **Style Sheet**

Your process will adopt the style sheet associated with the parent process.

You can override this style sheet by defining a process specific style sheet.

Before using the adopted layout, be sure to review the [3.4.1 Tips & Techniques](#).

For additional information, refer to [3.4.2 Technically Speaking](#).



[WIZ06 - Defining a Style Sheet](#)

3.4.1 Tips & Techniques

- In adopting the presentation layout of another LANSAs process, LANSAs for the Web allows you to override any of the components embedded in the presentation. You can create process specific left, right and horizontal menu components for the process even though you have chosen to adopt the presentation from another LANSAs process.
- If you change a standard menu definition in the parent process, the changes will be automatically used in the adopting process provided process specific components do not already exist.
- The Menu Separator Images variable (*LW3MENSEP for the default or *LW3PMSP_<process name> for a specific process) is embedded in the associated menu component. This value will always be evaluated based on the current process. If a value does not exist for the current process, the default system value will be used (i.e. the adopted from process value is not referenced in this case).
- If you no longer want to use the adopted layout, you can create a process specific layout.
- To reset to the DEFAULT layout, use the *Select Layout* option to delete the layout. For more details, refer to [3.1 Select Layout](#).
- The adopt layout feature is extremely useful if your Web application has many LANSAs processes. It allows you to customize the presentation layout for a single process and let the other processes adopt the presentation for this process to provide a consistent presentation for your complete application. For an example, refer to [WIZ04 - Defining a Standard Menu Component](#) and [WIZ06 - Defining a Style Sheet](#).
- If your Web application consists of more than one process, and you want all processes to have the same presentation layout, it is recommended to customize one layout first. When you are satisfied with the first process layout, use the Adopt Layout to inherit this "look-and-feel" for the other processes in your Web application.

3.4.2 Technically Speaking

When you adopt a layout, the Wizard creates a link between the process specific layout page and the adopted from page. **It does not create a copy of the adopted layout page.**

If you review the <process name>_LAYOUT page, you will see a single line:

```
<RDML PARENT="procname">
```

where `procname` is adopted from the process.

If you review the adopted from `procname_LAYOUT` page, you will see the layout which will be adopted by your process.

When a change is made to the adopted from layout, it is automatically used in the adopting process because they share the same page as specified by <RDML PARENT="procname">.

Once you define process specific variables or components for the adopting process, these variables and components will be used. The adopted from process variables and components can no longer be used.

The process specific pages are defined at the partition level.

Left/Horizontal Menu background color will always be the same as the adopted from menu.

A graphic variable `*LW3PCLR1_<process name>` has been defined to store the Left/Horizontal Menu Background Color. The default system value is stored in the graphic variable `*LW3COLOR1`.

These new variables are used exclusively by the Web Functions Wizard. LANSAs for the Web does not include a hierarchical interpretation of the variables. References to these variables in the layout page use conditional logic to determine if the system variable or process variable should be used. As a consequence of this architecture, the Left/Horizontal Menu Background Color will always be the same as the adopted process. Any process specific values in the adopting process will be ignored in this instance.

3.5 Tips & Techniques

- It is recommended to set up your process Layout Features before building your Layout to ensure all variables are included or excluded as expected.
- Customize one process (including menus and style sheet) then use adopt to apply this same look and feel to the other processes in your Web application.
- You can create process specific banner components using `<process name>_STDBANNER`. Remember that any changes to a specific process layout may impact other processes which have adopted this layout. Think carefully before creating default layout and default menu components. Be sure to review the section on [Impact on Existing Applications](#).
- If you want to adopt the layout from another process but have a different standard menu to the adopted process, first:
 - a) create your standard menu and a "dummy" layout for the adopting process (you must create the layout to generate the `<processname_LMENU>` component) THEN
 - b) modify the layout to adopt from the desired process.

This will guarantee that the required `<processname_LMENU>` and `<processname_LMENU1>` components both exist.

Conversely if you have a different standard menu on your adopting process and want to revert back to the menu of the adopted process:

- a) use the wizard to delete the standard menu component and then
- b) MANUALLY remove the associated `<processname_LMENU>` component using the Web Function Editor. The menu will then be automatically inherited from the adopted process.

3.6 Technically Speaking

For more technical information on the Wizard implementation refer to:

- [3.6.1 Layout Schemas](#)
- [3.6.2 Layout Pages](#)
- [3.6.3 System Variables](#)
- [3.6.4 Banner Components](#)
- [3.6.5 Menu Components](#)
- [3.6.6 Style Sheets](#)

3.6.1 Layout Schemas

The Web Functions Wizard is shipped with five layout schemas, each defined by a component (WIZLAYO01 through WIZLAYO05). They are installed into your partition when you import the Web Functions Wizard (save file LWEBWIZ2).

Do NOT modify the pages associated with these components or use this naming convention to create your own layouts as future releases of the Web Functions Wizard may update these pages.

When you have completed the customization of the presentation layout, the Web Functions Wizard creates a new layout page:

<process name>_LAYOUT

where <process name> is the name of your current LANSAP process (this will be DEFAULT for the system default component)

The layout pages are built from the layout schema components shipped with the Web Functions Wizard. The resulting layout page is then used when you run your LANSAP application. A comment tag in your HTML page will indicate which of the five layouts was selected.

3.6.2 Layout Pages

When you Build or Adopt a layout, the Web Functions Wizard creates (or modifies) a process specific HTML page:

<process name>_LAYOUT

where <process name> is the name of your current LANSAs process (this will be DEFAULT for the system default component). Using the Delete Layout button will delete this component.

The <process name>_LAYOUT page contains the details of the presentation. You can view the layout page using the Web Function Editor. The layout page is a partition specific Web component.

In the HTML generated for your LANSAs function, there is an <RDML LAYOUT> tag. When you execute the LANSAs function, this tag instructs LANSAs for the Web to use a presentation layout. LANSAs for the Web will attempt to locate a process specific layout page (<process name>_LAYOUT). If this page is not found, the DEFAULT_LAYOUT page is used.

LAYOUT pages generated by the Wizard incorporate the graphic variables used to store the Color Scheme. Another graphic variable is used to store the Layout Features (*LW3PLIN_<process name>) and is interpreted when the LAYOUT page is generated to ensure the appropriate details are included or excluded in the generated HTML page. Changes to the Color Scheme are automatically applied as they are interpreted at run time but changes to the Layout Features require a re-build of the Layout to modify the generated HTML.

To remove a process specific layout component so the DEFAULT_LAYOUT will be used, select the Delete Layout button from Select Layout or manually delete the <process name>_LAYOUT component using the Web Function Editor. The graphic variables associated with the Color Scheme can be deleted separately using the Delete button from Color Scheme.

The function is embedded in the presentation layout by the <RDML FUNCTION> tag. The location of this tag determines where the generated HTML for your LANSAs function is processed.

3.6.3 System Variables

The Web Functions Wizard uses graphic variables to control the presentation of the layout. The Wizard will attempt to locate process specific variables. If these variables are not found, the system wide default variables will be used.

The following graphic variables are used in combination with the Layout Pages generated by the Wizard:

System Variable Description	System level variable	Process level variable
Layout Background Color	*LW3CLNTCOLOR	*LW3PBGC_<process name>
Layout Left/Horizontal Menu Background Color	*LW3COLOR1	*LW3PCLR1_<process name>
Layout Right menu Background Color	*LW3RMENUBKGND	*LW3RMBG_<process name>
Layout Inclusion/Exclusion flags	*LW3LINCLUDE	*LW3PLIN_<process name>
Background Image	*LW3CLNTBKGND	*LW3PBGI_<process name>

The graphic variables associated with the Color Scheme can be deleted using the Delete button from [3.2 Color Scheme](#). The company logo is stored in the *LW3CPYLOGO graphic variable. (For more details, refer to [Process Level Graphic Variables](#)).

The Web Functions Wizard Help is stored in *LW3WIZHLP. (For details, refer to [Help Configuration](#)).

Several system and process graphic variables are used only for the Wizard and its associated layouts. For example, the selections associated with each Wizard process are saved and re-used when you next use that process. These Wizard usage selections are stored in graphic variables.

3.6.4 Banner Components

The default banner Web component used in the layouts is stored in STDBANNER.

You can create a process specific banner by using the Web Function Editor to create a banner Web component named <process name>_STDBANNER.

3.6.5 Menu Components

If you look at the generated layout page, you will notice that the first five to seven executable lines are <RDML> tags. For more details, refer to [RDML Tags](#).

```
<RDML SETTMPFLD="10" VALUE="1">  
<RDML SETTMPFLD="11" VALUE="1">  
<RDML SETTMPFLD="12" VALUE="1">  
<RDML SETTMPFLD="13" VALUE="1">  
<RDML SETTMPFLD="14" VALUE="1">  
<RDML SETTMPFLD="09" VALUE="1">  
<RDML TRACE>
```

The <RDML SETTMPFLD> tags are used to set user-updateable fields in LANSAs for the Web. These fields are used to determine whether or not a particular Layout Feature is shown. The values associated with these tags are a result of your choice of the Layout Features to include in your presentation layout.

- RDML SETTMPFLD="9" indicates a Wizard Supported Color has been selected
- RDML SETTMPFLD="10" is used for the Left Menu component
- RDML SETTMPFLD="11" is used for the Horizontal Menu component
- RDML SETTMPFLD="12" is used for the Right Menu component.
- RDML SETTMPFLD="13" is used for Banner Inclusion
- RDML SETTMPFLD="14" is used for Company Logo inclusion.

A value of "1" indicates the associated feature should be included when displaying the presentation layout.

If "Click Tracking" has been included in the Layout Feature, the command <RDML TRACE> is automatically embedded in the layout. For more details on configuring your Web server to use <RDML TRACE>, refer to [Enabling Event Logging](#).

The example shown has included all possible layout features.

If you have included a left or horizontal menu in your layout component, the HTML code is generated to include a DHTML Menu or Standard Menu as indicated in your Layout Features selection.

If your current process included a standard Left Menu and you review the

<process name>_LMENU page, you will see a single line:

```
<process name>_LMENU1
```

The <process name>_LMENU component will be embedded in the <process name>_LAYOUT of the <RDML MERGE=&LMENU> command.

If your current process included a DHTML Left Menu and you review the <process name>_LMENU page, you will see the following:

```
<table>
  <tr><td>
    <div id='MenuPos' style='position:relative; width:138; height:138;'>
      <img src='<RDML MERGE="&IMAGE"'/>/clearpixel.gif' width='138' height=
    </div>
  </td></tr>
</table>
```

As for a standard menu, the <process name>_LMENU component will be embedded in the <process name>_LAYOUT at the <RDML MERGE=&LMENU> command. In addition, the <process_name>_LAYOUT will include a reference to the component <process_name>_LMENU2 (which contains your DHTML menu definition) and the component WIZMNUSCR (which contains the javascript to support the DHTML menu).

Only a single DHTML menu can be used by a presentation layout.
--

3.6.6 Style Sheets

When the presentation layout is built, if no style sheet exists for the current process, a default style sheet will be created which includes all classes used by objects generated from the Wizard.

The default style sheet is defined by copying the WIZARDX_STYLE component into the current processes style component. The process WIZARDX is shipped with the Web Functions Wizard and is defined as follows:

```
<!-- WIZARDX_STYLE Style sheet -->
<style type="text/css">
<!--
.rightmenu{font-weight: bold; color:#FFFFFF; text-decoration: none;}
.selrightmenu{font-weight: bold; color:#000000; text-decoration: none;}
.leftmenu{font-weight: bold; color:#FFFFFF; text-decoration: none;}
.selleftmenu{font-weight: bold; color:#000000; text-decoration: none;}
.hrzmenu{font-weight: bold; color:#FFFFFF; text-decoration: none;}
.selhrzmenu{font-weight: bold; color:#000000; text-decoration: none;}
-->
</style>
```

4. Customizing a Standard Menu

The Web Functions Wizard supports two types of menu:

- Standard Menus
- DHTML Menus.

Standard Menus can be defined for the Left Menu component, the Right Menu component and the Horizontal Menu component.

For details on DHTML Menus, refer to [Customizing a DHTML Menu](#).

Modifications to standard menus apply to the process nominated as the current process on the left hand menu bar. The menu customization applies to all the LANSA functions defined in the current process. Menus are partition specific.

After defining the required menus, ensure that you include these menus in your presentation layout using the [Layout Features](#).

This picture shows samples of a Left, Right and Horizontal Standard Menu.

The screenshot displays a web application interface with a purple header and sidebar. The header contains a horizontal menu with items: Options, Tools, Calendar, and Help. The left sidebar contains a vertical menu with items: Home, Sections, and Personnel Skills. The main content area features a table with columns: Dept Code, Section Code, and Section Description. The table lists five rows of data. Above the table are two buttons: Next Page and New Search. To the right of the table is a right sidebar menu with items: Department Information, Administration Information, and Contact Us. Below the Contact Us item is a dropdown menu labeled State:.

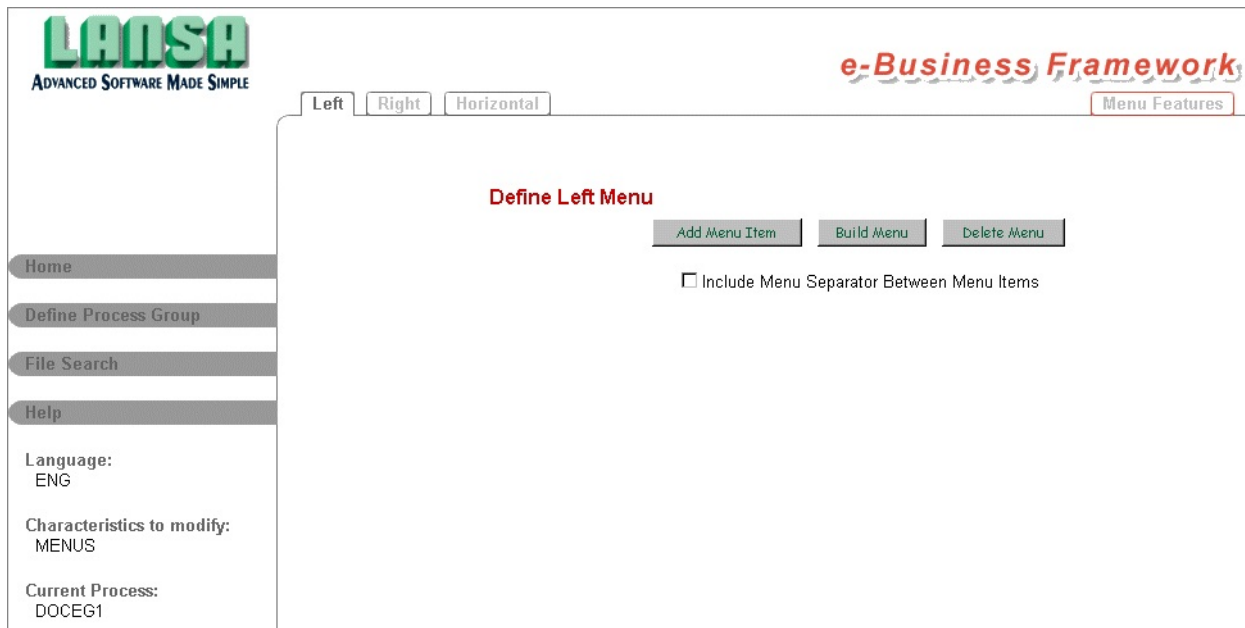
Dept Code	Section Code	Section Description
ADM 01		INTERNAL ADMIN SRV
ADM 02		PURCHASING SECTION
ADM 03		ACCOUNTING SECTION
ADM 04		SALES AND MARKETING
ADM 05		MAINTENANCE

Before changing any of the System Wide Default Menu components (current process DEFAULT), be sure that you have read the [Impact on Existing Applications](#).

Warning: If you generate a standard menu component using the Web Functions Wizard, you must generate the associated layout using the Wizard to ensure the menu is correctly applied.

4.1 Select Menu Component

Once you select the Menu characteristic and press Go on the [Wizard Home Page](#) you will be presented with the *Define Left Menu* screen.



The screenshot displays the 'Define Left Menu' interface. At the top left is the LAOSA logo with the tagline 'ADVANCED SOFTWARE MADE SIMPLE'. At the top right is the 'e-Business Framework' logo. Below the LAOSA logo is a vertical menu with items: Home, Define Process Group, File Search, and Help. Below the menu are fields for 'Language: ENG', 'Characteristics to modify: MENUS', and 'Current Process: DOCEG1'. The main content area has three tabs: 'Left' (selected), 'Right', and 'Horizontal'. Below the tabs is the title 'Define Left Menu' and three buttons: 'Add Menu Item', 'Build Menu', and 'Delete Menu'. A checkbox labeled 'Include Menu Separator Between Menu Items' is also present.

The Left Menu tab is automatically pre-selected. To define a Right or Horizontal Menu definition click on the appropriate tab. The steps to customize a Left, Right or Horizontal Menu are identical.

The Menu Features tab, on the right side of the screen, allows you to set up default values to be used when defining and building your Left, Right and Horizontal Menus. For more details, refer to [4.4 Menu Features](#)

To customize the system default menus, select the process DEFAULT on the left hand menu of the Wizard. The steps in customizing the system default menus or the menus for a specific process are the same.



[WIZ004 - Defining a Standard Menu Component](#)

4.2 Modifying a Menu Component

Once you have selected the Left, Right or Horizontal menu tab, you will be allowed to modify the requested menu component. If menu items have been previously defined, you will see a screen as shown below.

If this is a new menu component, there will be no menu items configured for the menu.

The screenshot shows the 'Define Horizontal Menu' screen in the LANSAS e-Business Framework. The interface includes a navigation menu on the left and a main content area with tabs for 'Left', 'Right', and 'Horizontal'. Below the tabs are buttons for 'Add Menu Item', 'Build Menu', and 'Delete Menu'. A table lists three menu items with columns for Seq., Menu Type, Menu State, Link Type, and Select an Action.

Seq.	Menu Type	Menu State	Link Type	Select an Action
10	Button Description Horizontal 1	Always enabled	LANSAS Function DOCEG2 / DOCSC1 Reuse LANSAS Job	Modify Delete
20	Button Description Horizontal 2	Always enabled	LANSAS Function DOCEG2 / DOCSC1 Launch new LANSAS job	Modify Delete
30	Button Description Horizontal 3	Always enabled	LANSAS Function DOCEGE2 / DOCSC1 Reuse LANSAS Job	Modify Delete

The option to include a menu separator image between the menu items is only supported for left or right menu components.

- To include a menu separator image, check the associated box before building the menu. The process specific Menu Separator image is defined using the Menu Features. The default system wide Menu Separator image is defined using either the Menu Features with current process DEFAULT, or using the System Variables characteristics from the Wizard Home Page. The process specific Menu Separator image (*LW3PMSP_<process name>) is used over the default system wide Menu Separator image (*LW3MENSEP).
- Check the box just before using the *Build Menu* button. If you check the box and use the *Add Menu Item* button, the box will be unchecked when you return to this screen after adding the menu item.
- Press the *Add Menu Item* button to enter the parameters for a new menu item.
- Press the *Delete Menu* button to delete the menu definition and associated

components.

- To modify or delete a specific menu item, select the appropriate action against the item from the menu item list.
- Press the Build Menu button once you are satisfied with the definition of your menu component. The menu cannot be used until it has been built. The message "Menu created successfully" will be issued when the menu has been created. Ensure the new menu is included in your presentation layout using the [Layout Features](#).

Note: If you have menu components created by earlier versions (before V10.0) of the Web Functions Wizard, when you display these menus using the current Web Functions Wizard, the menu definitions will be automatically converted to support the new format. Once you have viewed the menu using the current Web Functions Wizard, your menu definition will no longer be available for modification or review using an earlier version of the Web Functions Wizard. The associated menu component will not be modified until you select to re-build the menu using the new Web Functions Wizard.



[WIZ004 - Defining a Standard Menu Component](#)

4.3 Menu Item Details

Once you have pressed the *Add Menu Item* button or selected to modify an existing menu item, you can enter the parameters for the menu item.

Define Horizontal Menu

Save Menu Item Delete Menu Item Cancel

Menu Sequence

Sequence in Menu: 10

Menu Type Top Bottom

Use Component Merge Page Description Dynamic Button Single image Rollover images

Description, component, page name or tooltip: Horizontal 1

Dynamic Button Style: Button Curved Light Green

Static image (mouseOut image) /IMAGES/ Preview

MouseOver image /IMAGES/ Preview

Menu State Top Bottom

Always shown in the Menu Component

Only shown if the field condition is TRUE

Only shown if the field condition is FALSE

Field to check:

Value to check:

Link Type Top Bottom

No link

Component

URL

Start in New Window: *

LANSAS Function

Process: DOCEG2

Function: DOCS1

Reuse Job: *

Start in New Window: *

JavaScript: *

Details indicated with * are optional

Save Menu Item Delete Menu Item Cancel

When you are satisfied with your menu item definition, press the *Save Menu Item* button.

To delete the menu item, press the *Delete Menu Item* button. When defining a new menu item, you will not see the *Delete Menu Item* button.

Press *Cancel* to return without saving your changes.

The Wizard completes a simple validation of your menu items. For example, if

you specify a Menu Type of Rollover Images, the Wizard checks if you have entered a Static Image and a MouseOver Image and will clear any values from the Dynamic Button Style.

The configuration of a menu item is divided into parts including:

- [4.3.1 Menu Sequence Number](#)
- [4.3.2 Menu Type](#)
- [4.3.3 Menu State](#)
- [4.3.4 Link Type](#)



[WIZ004 - Defining a Standard Menu Component](#)

4.3.1 Menu Sequence Number

The *Sequence in Menu* item allows you to sequence the menu items in your Menu component. The Web Functions Wizard automatically assigns any new menu item to the end of the Menu component.

You can control the sequence by assigning a different sequence number to the item. For example, to make the item appear at the top of the menu, use a Sequence Number of 1.

The Sequence Numbers are automatically assigned using intervals of 10.

4.3.2 Menu Type

The *Menu Type* is used by the Web Functions Wizard to determine the HTML to generate for the menu item. A menu item can be:

- **A LANS A Web component** Select the *Use Component* option to embed an existing Web component into your Menu component. The name of the component is specified in the *Description* field. Enter the component name in upper case. The Dynamic Button Style, *Static Image* and *MouseOver Image* fields are ignored in this case.

The first item of the Right Menu component of the sample application is an example of a Web component used as a menu item. In this case, the Web component is a Banner component that cycles through a series of other Web components – set to cycle on every visit to the site. Using the menu item in this way allows for a number of promotional materials to be defined (as Web components) and LANS A for the Web then takes care of the cycling of these components via the use of the Web Banner component.

The code within the component is the responsibility of the developer.

- **A LANS A for the Web page** Select the *Merge Page* option to embed another HTML page into your Menu component. The name of the page is specified in the *Description* field. Enter the page name in upper case. The Dynamic Button Style, *Static Image* and *MouseOver Image* fields are ignored in this case.

The first item of the Left Menu component of the sample application is an example of a merged page. In this case, it is a page that displays a search capability.

The code within the Web page is the responsibility of the developer.

- **Static descriptive text** Select the *Description* option to specify text as the menu item. The text you specify in the *Description* field will be displayed as the menu item. The Dynamic Button Style, *Static Image* and *MouseOver Image* fields are ignored in this case.
- **A Dynamic Button** Select the *Dynamic Button* option to generate a button image. You must

select both a style and color for the buttons. The text you specify in the *Description* field will be displayed as the button text. The *Static Image* and *MouseOver Image* fields are ignored in this case.

This is the default menu type. The default value used for the button style and color can be set up using the [4.4 Menu Features](#)

The Web Functions Wizard is shipped with four button styles (only two are available for left and right menus) and 19 color options.

- **A Static Image**

Select the *Single Image* option if you do not want to have the MouseOver image effect. In this case, a single image is used for the menu item, i.e. the image does not change if the user moves his mouse over the menu item.

You can associate a user hint with the menu item in the *Description* field. This tip is shown when the user positions their mouse over the menu item.

Specify the name of the image in the *Static Image* field. Use the *Image* button to check if you have entered the name correctly or if the image exists in your image location.

To locate an image use the [File Search](#) facility on the Left Menu of the Wizard.

- **A Rollover Image**

Select the *Rollover Images* option to specify a different image to display when the user positions his mouse over the menu item.

You can associate a user hint with the menu item in the *Description* field. This tip is shown when the user positions their mouse over the menu item.

You specify the name of the image used, when the mouse is not positioned over the item, in the *Static Image* field. The image for the mouse over effect is specified in the *MouseOver Image* field.

Use the *Image* button to check if you have entered the name correctly or if the image exists in your image location.

To locate an image use the [File Search](#) facility on the Left Menu of the Wizard.

4.3.3 Menu State

The *Menu State* allows you to control when you want to show the menu item in the Menu component. For each item in the Menu component, you can choose to:

- Always show the item.
Select the *Always shown in the Menu component* option.
- Only show the item if a particular field exists in your application, and the value of the field is not blank or not zero.
Select the *Only shown if the field condition is TRUE* option. The name of the field used to determine the state of the item is specified in the *Field to check* field. For example, if you enter a field name and the field exists in your function (it must also be non blank or not zero), then the condition is considered TRUE and the menu item is displayed.

In addition, you can choose to only show the item if the specified field contains a particular value. This means that even though the field exists in the LANSAs function and it is non blank or not zero, the menu item will only be shown if the current field value matches the value you had specified in the *Value to check* field.

- Only show the item if a particular field does not exist in your application, or if it exists, the item is only shown if the current field value is blank or zero.
Select the *Only shown if the field condition is FALSE* option. This option is the reverse of the option previously described. For example, if you enter a field name but the field does not exist in your function, then the condition is considered FALSE and the menu item is displayed.

Fields that are part of a list cannot be used to determine the menu state.

- System variables can be used as both the "value to check" and the "field to check". For example, *FUNCTION (field to check) equal WIZ02 (value to check).
- Any field referenced in the "field to check" must be available to evaluate. For example, if #FUNCTION is not referenced directly in your RDML use *FUNCTION to evaluate the current function name.

4.3.4 Link Type

The *Link Type* defines what will happen when a menu item is selected. The Link Type options are identical to those available in DHTML menus. A menu item can be used to:

- Act as a group heading or separator
Select the *No link* option. This means that the menu item is not a hyperlink. For example, you can use this option to embed some text or an image into the menu.
- Use a component
Select the Component option. This option allows a LANSAs component to be executed when the link is selected. The code within the component is the responsibility of the developer.
- Link to a static page or another Web site
Select the *Static page* option. The URL to access the page or the Web site is specified in the field. For example, you might enter <http://www.lansa.com> to access the LANSAs Web site.

You can choose to launch the link in a new window by selecting the *Start in New Window* option.

- Link to another LANSAs function
Select the *LANSAs function* option. The name of the LANSAs process and function are specified in the *Process* and *Function* fields respectively. The current partition is assumed when invoking the function.

You can choose to use the HandleEvent JavaScript function to reuse the same LANSAs job at the Application Server by selecting the *Reuse Job* option. If this option is not selected, the menu item uses an anchor block to link to the LANSAs function. In other words, it always runs the LANSAs function as a new LANSAs job at the Application Server.

You can choose to launch the link in a new window by selecting the *Start in New Window* option.

The Reuse Job and Start in New Window options are mutually exclusive.

The Optional JavaScript option allows extra parameters to be passed before a

LANSA function is called. If it is left blank, it is assumed that no parameters are passed. An example of an optional JavaScript is:

```
document.LANSA.ANEXTACTION.value='ADD'
```

This will generate the following syntax for the JavaScript:

```
"JavaScript:document.LANSA.ANEXTACTION.value='ADD' ;HandleEvent('PROCESSNAME', 'FUNCTIONNAME');"
```

When adding optional JavaScript, it is your responsibility to make sure that the syntax is correct. The syntax is not validated.

If you choose to link to a non-*WEBEVENT function (i.e. procedural function) from a *WEBEVENT function, you should not select the *Reuse Job* option. You should launch the link in a new window by selecting the *Start in New Window* option.

4.4 Menu Features

Once you have selected the Menu Features tab you will see a screen as shown in this example.

The screenshot shows the 'Define Menu Features' interface within the LAOSA e-Business Framework. The top left features the LAOSA logo with the tagline 'ADVANCED SOFTWARE MADE SIMPLE'. The top right displays 'e-Business Framework' and a 'Menu Features' tab. Below the logo is a left-hand navigation menu with options: Home, Define Process Group, File Search, and Help. The main content area is titled 'Define Menu Features' and includes an 'Apply' button and a note: '* Menu must be rebuilt for any changes to take effect'. A 'Miscellaneous Process Settings' section contains a 'Menu Separator Image' field with the value '//IMAGES/' and a 'Preview' button, and a 'Default Button Style and Color' section with dropdown menus set to 'Button Curved' and 'Light Green'. At the bottom left, it shows 'Language: ENG', 'Characteristics to modify: MENUS', and 'Current Process: DOCEG2'.

You can specify the name of the Menu Separator Image for the current process. You can use the *Image* button to check if you have entered the name correctly or if the image exists in your image location. To locate an image use the [File Search](#) facility on the Left Menu of the Wizard. For more details, refer to [4.2 Modifying a Menu Component](#) .

Set the Default Button Style and Color to be used when you select to create a new menu item. For more details, refer to [4.3.2 Menu Type](#) .

After you have finished specifying your menu features, press the *Apply* button to update the variables. A completion message "Menu Features updated successfully" will be issued. The modified menu features will be used next time you modify a left, right or horizontal menu. To use a new menu separator image the appropriate menus must be re-built.

4.5 Multilingual Considerations

If you plan to execute your Web Function Application in multiple languages, start by creating your menus for a single language. When you are satisfied with the menu presentation and functionality, return to the [Wizard Home Page](#) and select a different language from the [Language](#) drop down in the wizard left hand menu.

Repeat the steps used to create the menus in your first language.

If the menu was originally created in English, notice that your new menu definitions will already exist and have been copied from the English version of the menu.

If the menu was originally defined in a language other than English, the menu definitions will still be copied with the exception of the Description, Static Image name, MouseOver Image Name and any Optional JavaScript.

The screenshot displays the 'Define Horizontal Menu' interface within the LANSAS e-Business Framework. The interface includes a sidebar on the left with navigation options: Home, Define Process Group, File Search, and Help. Below the sidebar, it shows the current language as 'ENG', characteristics to modify as 'MENUS', and the current process as 'AAPROC01'. The main area features a table of menu items with the following data:

Seq.	Menu Type	Menu State	Link Type	Select an Action
10	Button Description *****	Always enabled	LANSAS Function AAPROC01 / AA01 Launch new LANSAS job Start in new window	Modify Delete
20	Button Description *****	Always enabled	LANSAS Function AAPROC01 / AA02 Launch new LANSAS job Start in new window	Modify Delete
30	Button Description *****	Always enabled	LANSAS Function AAPROC01 / AA03 Reuse LANSAS Job Start in new window	Modify Delete
40	Button Description *****	Always enabled	Static Link Page	Modify Delete
50	Button Description *****	Always enabled	LANSAS Function AAPROC01 / AA01 Launch new LANSAS job Start in new window	Modify Delete

Notice that while the menu definitions have been copied (i.e. records defined on file DC@W32), the corresponding menu components do not exist for the current language in the Web Function Editor.

Modify each menu item as required for the new language, and then build the menu for the current language.

All menu details other than Description, Static Image name, MouseOver Image Name and any Optional JavaScript, are not language specific. If you change the value of any other fields, this change will modify the definition for all

languages. Changes to the definition will not take effect until the menu is rebuilt in the corresponding language.

When you delete a menu item, the menu item will be deleted from all languages. Be sure to rebuild the menu in all languages to ensure the menus are consistent for each language.

4.6 Tips & Techniques

- Menus created with earlier versions (before V10.0) of the Web Functions Wizard will be automatically converted for the current Web Functions Wizard. Be aware that once these menus have been converted, the menu definitions will no longer be available if you revert to running an earlier version of the Web Functions Wizard. The associated menu component will not be modified until you select to re-build the menu using the new Web Functions Wizard.
- It is recommended that you configure your Menu Features before defining or building any standard menus. This is especially useful if you intend to use dynamic buttons on your menus.
- The menu definitions (Standard and DHTML) are stored on the files DC@W31 and DC@W32. To use the Wizard in a different partition and access these menu definitions, the data from these files must be exported to the partition where the Wizard will be executed.
- When defining a Horizontal Standard Menu including dynamic buttons, you have the option to select a Tab or Button style. Tab buttons will sit on the edge of the horizontal menu area. The conventional button style is designed to sit above the edge of the horizontal menu area. The combined use of Tab and Button styles will provide undesirable results when the menu is built.
- When defining a Horizontal Standard Menu including dynamic buttons, be careful about including other Menu Types in the Menu. Variations in the menu width will provide undesirable results when the menu is built.
- It is not normal to define a dynamic button without an action. If that were the case, if the condition is true the button will appear but when the user clicks on it nothing will happen.
- If you are defining a Left or Right Standard Menu including dynamic buttons, you will only have the option to a conventional Button style. Tabs are not supported in this orientation.
- Defining the menus is the responsibility of the developer. All available combinations of menu type, menu state and link type can be used at the developer's discretion although some variations may not be good design. For example, dynamic buttons are typically used to execute something, so use of dynamic buttons with "no link" may cause user confusion.
- The code within components or Web pages used by the menu is the

responsibility of the developer.

- Any optional JavaScript is the responsibility of the developer.
- If you previously chose to Include Menu Separator between Menu Items, the box will be checked when you review the menu definition. Any subsequent changes to Menu items will cause the box to become unchecked.
- If you are using dynamic buttons on your menu, consider including the field STDTABFLR in your functions to automatically support on and off images to indicate which button is currently selected. The STDTABFLR field should be included as a *HIDDEN field in your REQUEST or DISPLAY statements, or in your GROUP_BY statements used with your screens. You can also add the STDTABFLR field to your DEFAULT_HIDDEN or <process>_HIDDEN page. For more details, refer to [WIZ004 – Defining a Standard Menu Component](#).
- When creating menus in multiple languages, configure a single language and test the menus before defining the menus in any other language. Only the Description, Static Image name, MouseOver Image Name and any Optional JavaScript will vary with language.
- When creating menus in multiple languages, one of which is English, configure the English menus first. The English version will then be used as the basis for menus in other languages.



[WIZ004 - Defining a Standard Menu Component](#)

4.7 Technically Speaking

- [4.7.1 Menu Components](#)
- [4.7.2 Style Sheets](#)

4.7.1 Menu Components

When you press the *Build Menu* button, the Web Functions Wizard generates an HTML page for the component. If you do not press the *Build Menu* button, your menu definition will be saved but any existing menu component will not correspond with the defined menu entries.

If you are defining a Left Menu component, the Web Functions Wizard generates an HTML page named as:

```
<process name>_LMENU1
```

where

<process name> is the name of the LANSAs process.

If it is the Horizontal Menu component, then the page is named <process name>_HMENU1. If you are defining a Right Menu component, the page generated is <process name>_RMENU.

The left and horizontal menu components have a different naming convention to support DHTML menu and standard menus. You can define both a DHTML menu and a Standard menu for the same menu component. This will allow you to prototype and experiment with DHTML and Standard menus by varying your [Layout Features](#).

When you select to include a standard menu, the <process name>_LMENU1 components will be referenced by the <process name>_LMENU component. For more details, refer to [Customizing a Presentation Layout – Technically Speaking](#).

The following HTML code example shows the HTML of the Left Menu component used in the Wizard. The Menu component consists of a single menu item.

```
<!-- LEFT Menu -->
<div align="center">
<table cellspacing="0" cellpadding="0" border="0">
<tr><td>
<RDML COMPONENT="WIZVALUES">
</td></tr>
</table>
</div>
```

4.7.2 Style Sheets

If your menu includes dynamic buttons, the generated menu component will include references to two associated Cascading Style Sheet classes. The classes used are as follows:

Menu	Classes
Left Menu	.leftmenu .selleftmenu
Right Menu	.rightmenu .selrightmenu
Horizontal Menu	.hrzmenu .selhrzmenu

A default style sheet, which includes the above classes, will be generated when you build a layout (if a style sheet does not already exist for the current process). The default style sheet removes the underline typically associated with a hyperlink and sets the default text color and weight. For more details, refer to [Customizing a Presentation Layout – Technically Speaking](#).

5. Customizing a DHTML Menu

The Web Functions Wizard supports two types of menu:

- Standard Menus and DHTML Menus.

DHTML menus can only be defined for the Left Menu component and the Horizontal Menu component. Only a single DHTML menu can be included in a presentation layout.

Note: The DHTML menu script used by the Wizard (component WIZMNUSCR) is not supported in version 6 of the Netscape browser. If you expect to your Web function applications to be used with Netscape V6, it is recommended that you use Standard Menus in your interface design. For details on Standard Menus, refer to [Customizing a Standard Menu](#).

DHTML menu features include:

- Multi-level submenus (i.e. 3 levels drill down)
- A customizable menu interface (e.g. color, alignment)

Modifications to DHTML menus apply to the process nominated as the current process on the left hand menu bar. The menu customization applies to all the LANSAs functions defined in the current process. Menus are partition specific.

After defining the required DHTML menu, illustrated below ensure that you include this menu in your presentation layout using the [Layout Features](#) option.

Warning: If you generate a DHTML menu component using the Web Functions Wizard, you must generate the associated layout using the Wizard to ensure the menu is correctly applied.

Update Details Delete Print

LAOSA

Employee Details

Employee Number A1234

Surname

Given Names

Address 1

Address 2

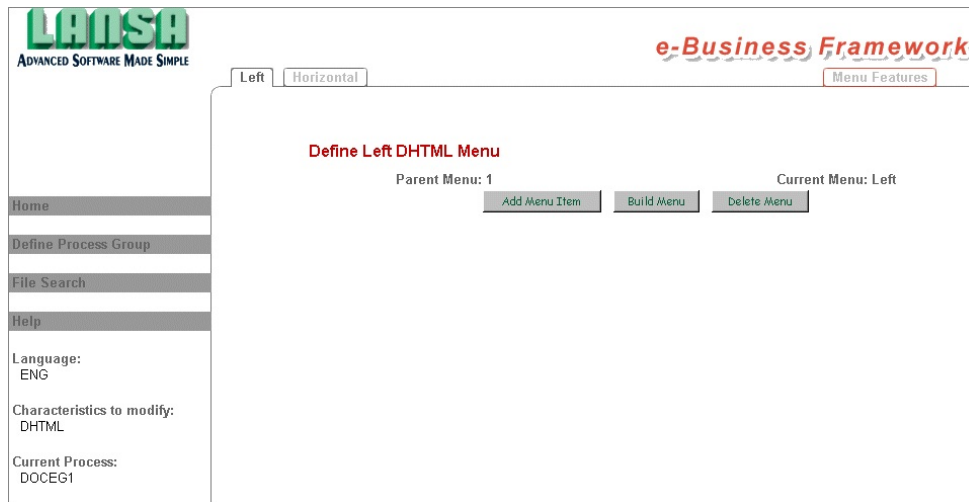
Address 3

Skill Code	Date Skl Acquired	Grade	Comment
ADVPGM	1/11/01	D	
INTPGM	12/06/01	C	
MBA	15/07/00	D	Credit Average
MANAGE1	2/02/02	P	Internal Training

Before changing any of the System Wide Default Menu components (current process DEFAULT), be sure that you have read the [Impact on Existing Applications](#).

5.1 Select Menu Component

Once you select the Menu characteristic and press *Go* on the [Wizard Home Page](#) you will be presented with the *Define Left HTML Menu* screen.



The screenshot shows the 'Define Left DHTML Menu' screen. The top left corner features the LAOSA logo with the tagline 'ADVANCED SOFTWARE MADE SIMPLE'. The top right corner displays 'e-Business Framework' and a 'Menu Features' tab. The main content area has two tabs: 'Left' (selected) and 'Horizontal'. Below the tabs, the title 'Define Left DHTML Menu' is centered. Underneath, it shows 'Parent Menu: 1' and 'Current Menu: Left'. Three buttons are visible: 'Add Menu Item', 'Build Menu', and 'Delete Menu'. On the left side, there is a vertical navigation menu with items: Home, Define Process Group, File Search, and Help. Below the menu, it shows 'Language: ENG', 'Characteristics to modify: DHTML', and 'Current Process: DOCEG1'.

The Left Menu tab is automatically pre-selected. To define a Horizontal Menu definition click on the Horizontal tab. The process of defining a Left or Horizontal Menu is identical.

Right menus cannot be defined as a DHTML menu.

The Menu Features tab, on the right side of the screen, allows you to set up default values to be used when building your Left and Horizontal Menus. For more details, refer to [5.4 Menu Features](#) .

To customize the system default menus, select the process DEFAULT on the left hand menu of the Wizard. The steps in customizing the system default menus or the menus for a specific process are the same.



[WIZ005 - Defining a DHTML Menu Component](#)

5.2 Modifying a Menu Component

Once you have selected the Left or Horizontal menu tab, you will be allowed to modify the requested menu component. If menu items have been previously defined, you will see them on the screen.

If this is a new menu component there will be no menu items configured for the menu.

The screenshot shows the LAOSA e-Business Framework interface. The top left features the LAOSA logo with the tagline "ADVANCED SOFTWARE MADE SIMPLE". The top right displays "e-Business Framework" and a "Menu Features" button. Below the logo, there are navigation tabs for "Left" and "Horizontal". The main content area is titled "Define Left DHTML Menu". It shows "Parent Menu: 1" and "Current Menu: Left". There are three buttons: "Add Menu Item", "Build Menu", and "Delete Menu". Below these buttons is a table with three columns: "Seq.", "Description", and "Select an Action".

Seq.	Description	Select an Action		
1	Employee Contact	Modify	Sub-Menu	Delete
2	Additional Information	Modify	Sub-Menu	Delete
3	Manager	Modify	Sub-Menu	Delete

On the left side of the interface, there is a sidebar with the following elements: "Home", "Define Process Group", "File Search", "Help", "Language: ENG", "Characteristics to modify: DHTML", and "Current Process: DOCEG2".

If you want to add a new menu item, press the *Add Menu Item* button. You will be allowed to enter the parameters for the menu item.

If you want to delete the menu definition and associated components, press the *Delete Menu* button.

To modify or delete a specific menu item, select the appropriate action against the item from the menu item list.

The Sub-Menu link, available against each menu item, allows you to define a menu below this option. The Parent Menu identifier indicates the current menu being modified. The top-level menu is always Parent Menu 1. After defining any Sub-Menus, use the Parent Menu button to return to the top-level menu. If you have drilled down several levels, the Parent Menu button will only return you to the immediate parent menu. Use the Parent Menu repeatedly to return to the top-level menu, Parent Menu 1. The Build Menu button is only available from the top-level menu.

Once you are satisfied with the definition of your Menu component, you can press the Build Menu button. The menu cannot be used until it has been built. When the menu has been created the message "Menu created successfully" will

be issued. Ensure the new menu is included in your presentation layout using the [Layout Features](#).

The menu component will not be modified until you select to Build Menu even though your menu definitions have been updated.



[WIZ005 - Defining a DHTML Menu Component](#)

5.3 Menu Item Details

Once you have pressed the *Add Menu Item* button or selected to modify an existing menu item, you will be allowed to enter the parameters for the menu item.

The screenshot shows the LANSA e-Business Framework interface. The top left features the LANSA logo with the tagline 'ADVANCED SOFTWARE MADE SIMPLE'. The top right displays 'e-Business Framework' and a 'Menu Features' button. The main content area is titled 'Define Left DHTML Menu' and includes a navigation bar with 'Left' and 'Horizontal' tabs. Below the title, there are buttons for 'Save Menu Item', 'Delete Menu Item', and 'Cancel'. The 'Parent Menu' is set to '1' and the 'Current Menu' is 'Left'. The 'Menu Sequence' section contains a 'Description' field with 'Employee Contact' and a 'Sequence in Menu' field with '1'. The 'Link Type' section has radio buttons for 'No link' (selected), 'Component', and 'URL'. Below these are fields for 'Process', 'Function', and 'Reuse Job', each with a checkbox and an asterisk indicating it is optional. A 'Start in New Window' checkbox is also present. A 'javascript:' field is at the bottom. A footer note states 'Details indicated with * are optional'.

When you are satisfied with your menu item definition, press the *Save Menu Item* button.

To delete the menu item, press the *Delete Menu Item* button. If you are defining a new menu item, you will not see the *Delete Menu Item* button.

Press *Cancel* to return without saving your changes. You will be returned to the current menu definition.

The Wizard completes a simple validation of your menu items. For example, if you specify a Link Type of URL, the Wizard checks if you have a URL and will clear any other variables associated with the Link Type.

The configuration of a menu item is divided into two parts:

- [5.3.1 Menu Sequence and Description](#)
- [5.3.2 Link Type](#)



5.3.1 Menu Sequence and Description

The *Description* will be displayed as the menu item.

The *Sequence Number* allows you to sequence the menu items in your Menu component. The Web Functions Wizard automatically assigns any new menu item to the end of the Menu component.

You can control the sequence by assigning a different sequence number to the item. For example, to make the item appear at the top of the menu, use a Sequence Number of 1.

The Sequence Numbers are automatically assigned using intervals of 1.

5.3.2 Link Type

The *Link Type* defines what will happen when a menu item is selected. The Link Type options are identical to those available in Standard menus. A menu item can be used to:

- Act as a menu heading or separator.
Select the *No link* option. This means that the menu item is not a hyperlink. This option is typically used to identify a sub-menu which will expand under this heading.
- Use a component.
Select the Component option. This option allows a LANSAs component to be when the link is selected. The code within the component is the responsibility of the developer.
Components used in DHTML menus must be defined as **script** type components in the Web Function Editor.
- Link to a static page or another Web site.
Select the *Static page* option. The URL to access the page or the Web site is specified in the field. For example, you might enter <http://www.lansa.com> to access the LANSAs Web site.
You can choose to launch the link in a new window by selecting the *Start in New Window* option.
- Link to another LANSAs function.
Select the *LANSAs function* option. The name of the LANSAs process and function are specified in the *Process* and *Function* fields respectively. The current partition is assumed when invoking the function.

The Reuse Job and Start in New Window options

You can choose to use the HandleEvent JavaScript function to reuse the same LANSAs job at the Application Server by selecting the *Reuse Job* option. If this option is not selected, the menu item uses an anchor block to link to the LANSAs function. In other words, it always runs the LANSAs function as a new LANSAs job at the Application Server.

You can choose to launch the link in a new window by selecting the *Start in New Window* option.

The Reuse Job and Start in New Window options are mutually exclusive.

The Optional JavaScript option allows extra parameters to be passed before a LANSAs function is called. If it is left blank, it is assumed that no parameters are

passed. An example of optional JavaScript is:

```
document.LANSA.ANEXTACTION.value='ADD'
```

This will generate the following syntax for the JavaScript:

```
"JavaScript:document.LANSA.ANEXTACTION.value='ADD'  
;HandleEvent('PROCESSNAME', 'FUNCTIONNAME');"
```

When adding optional JavaScript, it is your responsibility to make sure that the syntax is correct. The syntax is not validated.

If you choose to link to a non-*WEBEVENT function (i.e. procedural function) from a *WEBEVENT function, you should not select the *Reuse Job* option. You should launch the link in a new window by selecting the *Start in New Window* option.

5.4 Menu Features

Once you have selected the Menu Features tab, the Define DHTML Menu Features screen will be displayed.

Define DHTML Menu Features

Upper half of this screen.

** DHTML menus must be rebuilt for any changes to the menu features to take effect*

Color Settings

Background Color	<input type="text" value="#709E6D"/>	Color	<input type="button" value="Color Picker"/>
Sub-Menu Background Color	<input type="text" value="#BACAB9"/>	Color	<input type="button" value="Color Picker"/>
Highlighted Background Color	<input type="text" value="#BACAB9"/>	Color	<input type="button" value="Color Picker"/>
Highlighted Sub-Menu Background Color	<input type="text" value="#709E6D"/>	Color	<input type="button" value="Color Picker"/>
Font Color	<input type="text" value="#395138"/>	Color	<input type="button" value="Color Picker"/>
Font Sub-Menu Color	<input type="text" value="#395138"/>	Color	<input type="button" value="Color Picker"/>
Font Highlighted Color	<input type="text"/>	Color	<input type="button" value="Color Picker"/>
Font Sub-Menu Highlighted Color	<input type="text"/>	Color	<input type="button" value="Color Picker"/>
Border Color	<input type="text"/>	Color	<input type="button" value="Color Picker"/>
Border Sub-Menu Color	<input type="text"/>	Color	<input type="button" value="Color Picker"/>

Font Settings

Font Family	<input type="text" value="Arial"/>	<input type="button" value="Garamond"/>
Font Size (points)	<input type="text"/>	
Font Bold	<input type="text" value="Yes"/>	
Font Italic	<input type="text" value="No"/>	

Relational Settings	
Border Width (pixels)	<input type="text"/>
Border Between Elements	<input type="button" value="Yes"/>
Menu Text Position	<input type="button" value="Left"/>
Menu Position	<input type="button" value="Left"/>
Menu Vertical Position	<input type="button" value="Top"/>
Child Overlap	<input type="button" value="0.1"/>
Child Sub-Menu Overlap	<input type="button" value="0.1"/>
Offset from Top	<input type="text" value="15"/>
Offset from Left	<input type="text"/>
Left Padding	<input type="text"/>
Top Padding	<input type="text"/>

Lower half of the **Define DHTML Menu Features** screen

Other Miscellaneous Settings	
Time Sub-Menu is Visible (milliseconds)	<input type="text"/>
Unfold Direction	<input type="button" value="Left to Right"/>
Unfold Action	<input type="button" value="On Click"/>
Show Arrows	<input type="button" value="Yes"/>
Keep Highlighted Path	<input type="button" value="Yes"/>
Menu Element Height	<input type="text"/>
Menu Element Width	<input type="text" value="143"/>

** DHTML menus must be rebuilt for any changes to the menu features to take effect*

There are a large number of Menu Features for DHTML Menus. The features allow you to customize the appearance of the DHTML menu. Changing the menu features will allow you to modify the color settings, font settings, relational settings and several other parameters related to the DHTML menu presentation.

The initial values used for the Menu Features are the default values. If you intend to modify any of these values, take note of the original value so they can be restored later if you choose to "undo" you modifications.

Any modifications to the default settings are stored in a set of process specific graphic variables. The original settings are program defaults (i.e. they are NOT stored as system graphic variables). For more details, refer to [5.7 Technically Speaking](#).

These variables can radically change the appearance of your DHTML

menus and should be modified with caution.

After you have finished specifying your menu features, press the *Apply* button to update the variables. A completion message "Menu Features updated successfully" will be issued. The modified menu features will be used next time you modify a Left or Horizontal menu. To use a new menu separator image the appropriate menus must be re-built. The appropriate menus must be re-built to include changes to the Menu Features.

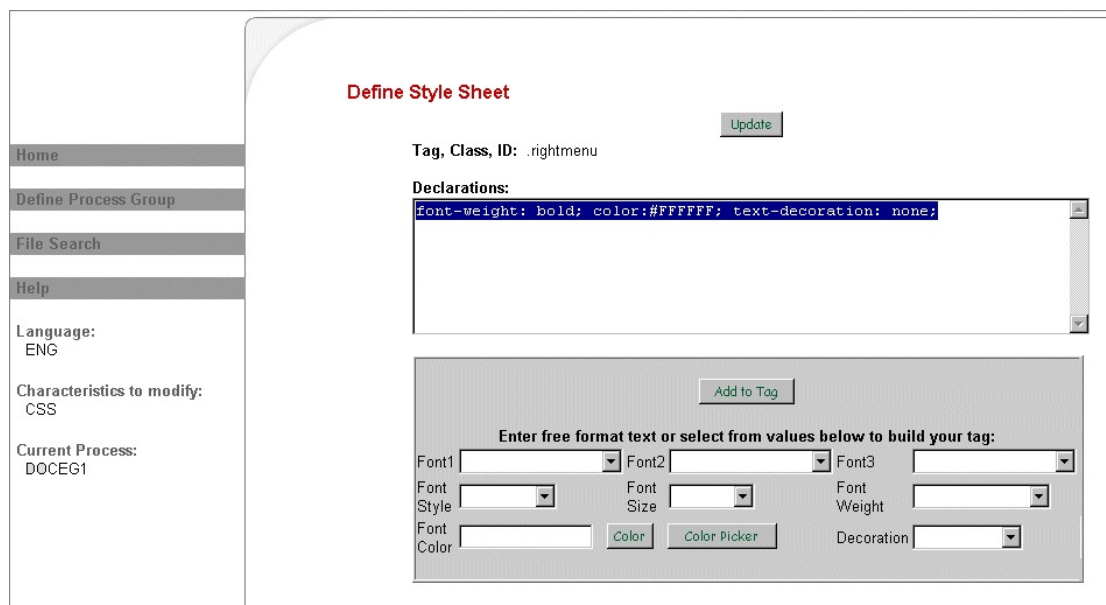
5.5 Multilingual Considerations

If you plan to execute your Web Function Application in multiple languages, start by creating your menus for a single language. When you are satisfied with the menu presentation and functionality, return to the [Wizard Home Page](#) and select a different language from the Language drop down in the wizard left hand menu. For more details, refer to [Language](#).

Repeat the steps used to create the menus in your first language.

If the menu was originally created in English, notice that your new menu definitions will already exist and have been copied from the English version of the menu.

If the menu was originally defined in a language other than English, the menu definitions will still be copied with the exception of the Description, Static Image name, MouseOver Image Name and any Optional JavaScript.



Notice that while the menu definitions have been copied (i.e. records defined on file DC@W32), the corresponding menu components do not exist for the current language in the Web Function Editor.

Modify each menu item as required for the new language, and then build the menu for the current language.

All menu details, other than Description, Static Image name, MouseOver Image Name and any Optional JavaScript are not language specific. If you change the value of any other fields, this change will modify the definition for all languages. Changes to the definition will not take effect until the menu is re-

built in the corresponding language.

When you delete a menu item, it will be deleted from all languages. Be sure to rebuild the menu in all languages to ensure the menus are consistent for each language.

5.6 Tips & Techniques

- It is recommended that you configure your Menu Features after defining and building your DHTML menus. This approach will allow you to see the default options and use the result as a basic model. If you then want to modify any of the menu features (e.g. font color of the top-level menu), modify only a few at a time then re-build to see how this has changed your menu.

This approach will give you a better understanding of what you have just changed and how to reverse the change if you do not like the resulting effect.

- The menu definitions (Standard and DHTML) are stored on the files DC@W31 and DC@W32. To use the Wizard in a different partition and access these menu definitions, the data from these files must be exported to the partition where the Wizard will be executed.
- The code within components or web pages used by the menu is the responsibility of the developer.
- Any optional JavaScript is the responsibility of the developer.
- When creating menus in multiple languages, configure a single language and test the menus before defining the menus in any other language. Only the Description, Static Image name, MouseOver Image Name and any Optional JavaScript will vary with language.
- When creating menus in multiple languages, one of which is English, configure the English menus first. The English version will then be used as the basis for menus in other languages.



[WIZ005 - Defining a DHTML Menu Component](#)

5.7 Technically Speaking

- [5.7.1 Menu Components](#)
- [5.7.2 Menu Features](#)

5.7.1 Menu Components

When you press the *Build Menu* button, the Web Functions Wizard generates an HTML page for the component. If you do not press the *Build Menu* button, your menu definition will be saved but any existing menu component will not correspond with the defined menu entries.

When you select the Build Menu button, if you are defining a Left Menu component the Web Functions Wizard generates an HTML page named as:

<process name>_LMENU2

where <process name> is the name of the LANSAs process.

If it is the Horizontal Menu component, then the page is named <process name>_HMENU2.

This naming convention allows the Wizard to support DHTML menu and standard menu definitions concurrently. You can define both a DHTML menu and a Standard menu for the same menu component. This will allow you to prototype and experiment with DHTML and Standard menus by varying your [Layout Features](#).

The <process name>_LMENU2 component contains the JavaScript definition of your menu. This component in combination with the <process_name>_LMENU and WIZMNUSCR components define a DHTML menu.

When you build a layout presentation, the component <process_name>_LMENU2 and the component WIZMNUSCR (executable JavaScript) are embedded in the layout page just after the body tag. The <process_name>_LMENU component will be embedded in the layout HTML component at the point where the menu will be displayed. This includes a simple table definition to anchor the menu. For more details, refer to [Customizing a Presentation Layout – Technically Speaking](#).

The following HTML code example shows the HTML of a Layout component, which includes a DHTML menu:

```
<!-- LANSAs for Web Functions Editor -->
<RDML SETTMPFLD="10" VALUE="1">
<RDML SETTMPFLD="11" VALUE="1">
<RDML SETTMPFLD="12" VALUE="1">
<RDML SETTMPFLD="13" VALUE="0">
<RDML SETTMPFLD="14" VALUE="0">
```

<!-- Layout Scheme #4 -->

<html>

<head>

<title><RDML MERGE="&FUNCTION"></title>

<RDML MERGE="&HEADER">

<RDML MERGE="&STYLE">

<RDML MERGE="&SCRIPT">

</head>

<body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0" bgcolor="

<RDML MERGE="*LW3CLNTCOLOR">" background="

<RDML MERGE="*LW3CLNTBKGND">" <RDML COMPONENT="FORI

<script type='text/javascript'>function Go(){return}</script>

<RDML PAGE="XXPROC01_HMENU2">

<RDML PAGE="WIZMNUSCR">

<noscript>Your browser does not support script</noscript>

<table border="0" cellpadding="0" cellspacing="0" width="100%">

<RDML NOTCONDITION="*LW3PCLR1_XXPROC01 ">

<tr height="84" valign="middle" align="center" bgcolor="

<RDML MERGE="*LW3COLOR1">" >

</RDML>

<RDML ONCONDITION="*LW3PCLR1_XXPROC01 ">

<tr height="84" valign="middle" align="center" bgcolor="

<RDML MERGE="*LW3PCLR1_XXPROC01 ">" >

</RDML>

<td width="150" >

<RDML ONCONDITION="&USETMPFLD14" VALUE="1">

 ">

</RDML>

 </td>

<td>

<RDML ONCONDITION="&USETMPFLD13" VALUE="1">

<RDML COMPONENT="STDBANNER">

</RDML>

</td>

</tr>

<RDML NOTCONDITION="*LW3PCLR1_XXPROC01 ">

<tr bgcolor="<RDML MERGE="*LW3COLOR1">" >

```

</RDML>
<RDML ONCONDITION="*LW3PCLR1_XXPROC01 ">
  <tr bgcolor="<RDML MERGE="*LW3PCLR1_XXPROC01 ">" >
</RDML>
  <td>&nbsp;</td>
  <td align="center">&nbsp;</td>
<RDML ONCONDITION="&USETMPFLD11" VALUE="1">
<RDML MERGE="&HMENU">
</RDML>
  </td>
</tr>
<tr>
  <td valign="top" align="center">
<RDML ONCONDITION="&USETMPFLD10" VALUE="1">
<br /><br /><RDML MERGE="&LMENU">
</RDML>
  </td>
  <td valign="top" align="center">
<RDML ONCONDITION="&USETMPFLD12" VALUE="1">
  <table border="0" cellpadding="0" cellspacing="0" width="100%">
    <tr>
      <td align="center">
</RDML>
      <br /><RDML FUNCTION>
<RDML ONCONDITION="&USETMPFLD12" VALUE="1">
      </td>
      <td width="130" align="right" valign="top"><br /><br />
<RDML MERGE="&RMENU"></td>
    </tr>
  </table>
</RDML>
  </td>
</tr>
</table>
</body>
</html>

```

5.7.2 Menu Features

Any modification to the default DHTML Menu Features will be stored in these graphic variables:

System Variable Description	Process level variable	Default	Allowed Values & Additional Information
Low Background Color	*LW3PDC01_<process name>	White	Color name of hex value
Low Sub-Menu Background Color	*LW3PDC02_<process name>	White	Color name of hex value
High Background Color	*LW3PDC03_<process name>	Black	Color name of hex value
High Sub-Menu Background Color	*LW3PDC04_<process name>	Black	Color name of hex value
Font Low Color	*LW3PDC05_<process name>	Black	Color name of hex value.
Font Sub-Menu Low Color	*LW3PDC06_<process name>	Black	Color name of hex value
Font High Color	*LW3PDC07_<process name>	White	Color name of hex value
Font Sub-Menu High Color	*LW3PDC08_<process name>	White	Color name of hex value
Border Color	*LW3PDC09_<process name>	Black	Color name of hex value
Border Sub-Menu Color	*LW3PDC10_<process name>	Black	Color name of hex value.
Border Width	*LW3PDT01_<process name>	1 pixel	Number interpreted as pixels.
Border Between Elements	*LW3PDT02_<process name>	1	0 = no border 1=border

Font Family	*LW3PDT03_<process name>	None
Font Size	*LW3PDT04_<process name>	9 points Number interpreted as points
Font Bold	*LW3PDT05_<process name>	1 0=normal 1=bold
Font Italic	*LW3PDT06_<process name>	0 0=normal 1=italic
Menu Text Centered	*LW3PDT07_<process name>	Left Left, center or right
Menu Centered	*LW3PDT08_<process name>	Left Left, center or right
Menu Vertical Centered	*LW3PDT09_<process name>	Top Top, middle or bottom
Child Overlap	*LW3PDT10_<process name>	0.2 Number between 0 and 1
Child Vertical Overlap	*LW3PDT11_<process name>	0.2 Number between 0 and 1
Start Top	*LW3PDT12_<process name>	0 Offset for vertical menu
Start Left	*LW3PDT13_<process name>	0 Offset for Horizontal Menu
Left Padding	*LW3PDT14_<process name>	3 Space between left border and text
Top Padding	*LW3PDT15_<process name>	2 Space between top border and text
Disappear Delay	*LW3PDT16_<process name>	1000 msec. Milliseconds sub-menu stays open.

Menu Wrap	*LW3PDT17_<process 1 name>	1=Wrap 0=no wrap
Right To Left	*LW3PDT18_<process 0 name>	1=Right to left 0=Left to right
Unfolds on Click	*LW3PDT19_<process 0 name>	1=unfolds on click 0=Unfolds on MouseOver
Show Arrows	*LW3PDT20_<process 1 name>	1=Show Arrows 0=Hide Arrows
Keep Highlight	*LW3PDT21_<process 1 name>	1=Selected path stays highlighted 0=path highlights do not stay
Menu Element Height	*LW3PDT22_<process 20 name>	
Menu Element Width	*LW3PDT23_<process 150 name>	

6. Style Sheets

The Web Functions Wizard allows you to quickly create a Cascading Style Sheet (CSS) to be used with your Web function application. You can use a CSS to define:

- the presentation of an HTML element, including the font (face, size and color), background color,
- the positioning of elements within your HTML page, and
- text decoration such as italics or underline.

Style sheets can be created in the Wizard using a free-format editor or using a tag assistant to build simple CSS rules.

Designing simple style sheets is easy. To get started, you need to understand some HTML and some basic CSS concepts.

For example, to set the text color of the 'H1' elements to blue you could define the simple CSS rule:

```
H1{color : blue}
```

This rule consists of two main parts:

- the selector ('H1') and
- the declaration ('color : blue').

The selector may be an HTML element (tag), a class, sub-class or id. The selector is the link between the HTML document and the style sheet. All HTML elements are possible selectors.

The **declaration** has two parts: the property ('color') and the value ('blue'). The 'color' property is one of about 50 properties which determine the presentation of an HTML page.

In your Web Function Application, modifications to a style sheet will impact the process nominated as the current process on the left hand menu bar. The style sheet customization applies to all the LANSA functions defined in the current process. Style sheets are partition specific.

Any modifications to your style sheet will be immediately available in your Web function application (i.e. no function compiles or layout configuration changes are required).

Before changing the System Default Style Sheet component (current process DEFAULT), be sure that you have read [Impact on Existing](#)

Applications.

For more details, refer to [Cascading Style Sheets](#) , a good reference book on Dynamic HTML (DHTML) or the World Wide Web Consortium at <http://www.w3c.org>

6.1 Define Style Sheet – Main Page

Once you select the Style Sheet characteristic and press [Go](#) on the [Wizard Home Page](#), you will be presented with the *Define Style Sheet* window.

Selector	Select an Action
.rightmenu	Modify Delete
.selrightmenu	Modify Delete
.leftmenu	Modify Delete
.selleftmenu	Modify Delete
.hrzmenu	Modify Delete
.selhrzmenu	Modify Delete

Use the [Add a Tag, Class or Id](#) button to add a selector to the style sheet.

Use the [Modify](#) link associated with a specific selector to review the selector's associated declarations.

Use the [Delete](#) link associated with a specific selector to delete the tag, class or id. This will not remove any references to this selector in your Web Application.

After you have added, modified or deleted a style sheet selector, you will be returned to the Define Style Sheet Main Page. You can then continue to modify your style sheet, or press the [Build Style Sheet](#) button to generate the style sheet. Any modifications to the selectors for the current process' style sheet will require the Style Sheet to be re-built.

The [Delete Layout](#) button will remove the style sheet component for the current process.



6.2 Define Style Sheet – Detail Page

Once you have selected Add a Class, Tag or Id, you will be presented with the *Define Style Sheet* screen requesting selector and declaration details.

Define Style Sheet

- Indicate a Class using a prefix of . or an ID with a prefix of #

Add Tag

Select Tag :

OR

Class, ID, Sub-Class or other Tag:

Declarations:

Add to Tag

Enter free format text or select from values below to build your tag:

Font1 Font2 Font3

Font Style Font Size Font Weight

Font Color Decoration

To define the selector you can:

- Select an HTML tag from the associated drop down, e.g. body:
- Enter a free-format entry in the Class, ID, Sub-Class or other Tag
- Use the drop down and free-format entry in combination to create a sub-class (also known as a pseudo-class), e.g. a:linked. The same selector could be defined using only the free-format entry.

To define the declarations you can:

- Use the free-format editor to enter the required properties and values, and then press the Add to Tag button to add this to the free-format text area.
- Use the Tag Assistant to build properties and values.

Commonly used properties and values can be built using the tag assistant. More

complex declarations must be entered using the free-format editor.

If you select to modify a selector from the Define Style Sheet Main Page, only the declaration part of the rule can be modified.

The style sheet component will not be modified until you select to Build Style Sheet.

For guidelines on defining CSS and sample style sheets refer to the World Wide Web Consortium at <http://www.w3c.org/style/css/>



WIZ006 - Defining a Style Sheet

6.3 Multilingual Considerations

Your style sheet should not include text that will vary by language. A single style sheet defined in your default system language will be used in all languages and will ensure consistent presentation of your HTML pages.

Style sheets describe how documents are presented on screens, in print or perhaps how they are pronounced. They change the presentation of a document without modifying the underlying HTML.

6.4 CSS Tips & Techniques

- The general format of a CSS rule is:

```
selector {property : value}
```

The Wizard will automatically insert the { } around the declarations (i.e. properties and values) when you build the style sheet.

- Each property must be separated by a semi-colon. For example:

```
H1{ color:"#C00000"; font-family: arial, garamond;}
```

- The Tag Assistant will not modify properties and values which are already defined as declarations. To modify a property or value using the Tag Assistant, remove the original reference from the free-format text area then use the Tag Assistant to generate the declaration again.
- An RDML component can be embedded within a style sheet to simplify maintenance even further. For example, you may have a Web component FONTPREF1 defined as a text component containing the default font family for your Web Function Application. The style sheet may then contain the following tags using the default font component:

```
<style type="text/css">  
body { font-family: <RDML COMPONENT="FONTPREF1">;}  
table { font-family: <RDML COMPONENT="FONTPREF2">;}  
</style>
```

- To reduce the size of a style sheet (and hence the download overhead), selectors can be grouped in a comma-separated group. For example:

```
H1, H2, H3{ font-family : helvetica }
```
- Style properties are generally inherited from the parent element to the child element. If no color has been assigned to the 'strong' element, the strong "LANSA" will inherit the color of the parent element. There are some obvious exceptions to this rule. For example, the 'background' property does not inherit, but the parent element's background will shine through by default. For example:

```
<H1><strong>LANSA</strong> Web Functions Wizard</H1>
```
- Comments can be added to your style sheet using /* to open the comment and */ to close the comment. For example:

```
H1{ color:"#0000FF" } /* blue */
```
- For more information on defining CSS, refer to a good reference book on

Dynamic HTML (DHTML) or the World Wide Web Consortium at
<http://www.w3c.org>



[WIZ006 - Defining a Style Sheet](#)

6.5 Technically Speaking

When you Build a Style Sheet, the Web Functions Wizard creates (or modifies) a process specific HTML page:

<process name>_STYLE

where <process name> is the name of your current LANSa process (this will be DEFAULT for the system default component). Using the Delete Style Sheet button will delete this component.

The <process name>_STYLE page contains the style sheet definition. You can view the style sheet page using the Web Functions Wizard or the Web Function Editor. The style sheet page is a partition specific Web component.

In the <process name>_LAYOUT component generated for your current process (refer to [Customizing a Presentation Layout](#)), there is an <RDML MERGE="&STYLE"> tag in the header section of the HTML. When you execute a LANSa function, the layout component is executed including any reference to the style sheet. LANSa for the Web will attempt to locate a process specific style sheet page (<process name>_STYLE). If this page is not found, the DEFAULT_STYLE page is used.

To remove a process specific style sheet component so the DEFAULT_STYLE will be used, select the Delete Style Sheet button from Define Style Sheet – Main Page or manually delete the <process name>_STYLE component using the Web Function Editor.

7. System Wide Graphic Variables

The Web Functions Wizard references several of the LANSA for the Web system graphic variables in conjunction with a set of Web Functions Wizard specific system graphic variables in the Wizard generated components.

To simplify the maintenance of these graphic variables, the Web Functions Wizard provides a single screen maintenance to review and update many of these values. Several of these graphic variables can also be modified at other logical points in the Web Functions Wizard interface. Alternatively, these graphic variables can be maintained using the Web Function Editor. The advantages of using the Web Functions Wizard are that you do not need to know the exact name of the graphic variable you want to change and that several values can be updated from a single screen.

For more details, refer to [Graphic Variables](#) .

When you select the System Variables characteristic from the [Wizard Home Page](#) you are presented with a single page maintenance. This page groups the graphic variables based on how they are utilized in a Web Function Application interface generated from the Wizard. You can configure the graphic variables associated with:

- [7.1.1 Background Settings](#) – related to the layout component
- [7.1.2 Menu Settings](#) – related to the layout and standard menu components
- [7.1.3 Images](#) – related to the layout component
- [7.1.4 Help Files Location](#) – related to the Wizard Help option

Reminder: Any changes made to the System Variables will impact all partitions in your LANSA system. For more details, refer to [Partition Wide versus System Wide](#)

7.1 Maintain System Variables

Once you select the System Variables characteristic and press *Go* on the [Wizard Home Page](#) (no process selection is required as you are working with system level objects), you will be presented with the *Maintain System Variables* window.

LANSAS
ADVANCED SOFTWARE MADE SIMPLE

e-Business Framework

Maintain System Variables

• WARNING: You are modifying the system default values

Apply

Background Settings Top Bottom

Default Background Color: #FFFFFF Color Color Picker

Background Image: Image

Menu Settings Top Bottom

Default Left/Horizontal Menu Background Color: #C0c0c0 Color Color Picker

Right Menu Background Color: Color Color Picker

Selection Image for Process Menu Items: sgr.gif Image

Menu Item Separator Image: hs_red.gif Image

Images Top Bottom

Company Logo: lansa.gif Image

LANSAS Messages Image: messages.gif Image

Help Files Location Top Bottom

Default Help Files Location: http://www.lansa.com/ebizhlp/

Process specific variables may override some of these values.

For colors, you can manually enter values by using the color name (e.g. red) or by using their hex code values (e.g. #FF0000). Alternatively, you may use the [7.2 Color & Color Picker](#) to select from a list of 216 browser-safe colors and a selected palette of grays. The *Color* button will display the color you have specified in the related field.

The [7.3 Image](#) button will display the specified image, if it exists in the image location configured for LANSAS for the Web. To locate an image use the [File Search](#) feature in the left menu, being careful to modify the file path as required when copying this to the image related graphic variable.

After you are satisfied with your configuration, press the *Apply* button.

For additional information, refer to [7.1.5 Tips & Techniques](#) or [7.1.6 Technically Speaking](#).

7.1.1 Background Settings

You can specify either an image and/or a color to be used in the background area of your functions.

In most cases, you should use either the background image or the background color. If both are specified, the background image is used. However, if the background image is partly transparent, the background color will also appear.

If you choose a Wizard supported color in combination with a layout scheme which includes curved corners when building your presentation layout (refer to [Select Layout](#)), it is highly recommended that the background color be set to white (#FFFFFF). For more details, refer to [Color Scheme](#).

These same values can be maintained using the Layout characteristic. To review the Color Scheme for the process, select **DEFAULT**.

All the layout color scheme values can be overridden by process specific settings.

The background settings are optional and can be left blank.



[WIZ003 - Customizing a Presentation Layout](#)

7.1.2 Menu Settings

The *Default Left/Horizontal Menu Background Color* and *Right Menu Background Color* are embedded in the Wizard generated layout components. These same values can be maintained using the [Layout](#) characteristic to review the [Color Scheme](#) for process **DEFAULT**.

The image specified for the *Menu item separator image* will be used if you have requested a menu separator between menu items (see the picture in [Customizing a Standard Menu](#)). This separator image will be used in both standard menu components. The Wizard contains a series of horizontal menu separators in various colors. These images are named as HS_<color>.GIF (where color is BLE, BLK, GRN, GRY, ORG, PRE, RED, YLW, CLEAR). Check your image location for a list of these images. The same value can be maintained using the *Menus* characteristic to review the [Menu Features](#) for process **DEFAULT**.

The Selection Image for Process Menu Items is not used in any components generated by the Wizard, but has been included for easy maintenance if you choose to use this value in your own components.

These values can be overridden by process specific settings.

The color settings are optional.

The *Default Left/Horizontal Menu Background Color* is an *Web Functions Wizard* specific system graphic variable. It will only work correctly in components generated by the *Web Functions Wizard*.



[WIZ003 - Customizing a Presentation Layout](#)

[WIZ004 - Customizing a Standard Menu Component](#)

7.1.3 Images

The company logo image will be used in your process layouts if you have selected to include the company logo in your layout using the [Layout Features](#).

The image for the LANSA messages is often displayed beside the LANSA messages. The Wizard message interface has been modified not to include the messages image. Refer to [Other Default Pages](#) for further information.

There are no process specific variables to override these values.

You must enter a value for both images if they are used in your Web Function Application. If a value is not entered, your Web pages will fail to locate and load the file.



[WIZ003 - Customizing a Presentation Layout](#)

7.1.4 Help Files Location

The Help File Location is used to locate the Wizard help when you select the Help option from the Wizard left menu.

The default help file location is <http://www.lansa.com/ebizhlp/wizard2.htm>

There is no process specific variable to override this value.

For more details refer to [Help Configuration](#).

7.1.5 Tips & Techniques

- Any changes made to these variables will impact all partitions in your LANSa system. Be very careful when making any changes to these values.
- Process specific variables will override some of these values. For details, refer to [Customizing a Presentation Layout](#) and [Customizing a Standard Menu](#).
- If your Maintain System Variable screen includes or excludes values, the details stored in the LANSa internal file DC@W33 have been modified since your last LANSa upgrade or install.
- Be sure to refer to [graphic variables](#) for more details.
- Browse list system graphic variables are not maintained on this page. To review *Browse List Graphic Variables*, select the *Browse Lists* characteristic from the *Wizard Home Page* to modify the browse list named DEFAULT. Refer to [Customizing Browse Lists](#) for further information.

7.1.6 Technically Speaking

You are specifying the values for these graphic variables:

*LW3CLNTBKGND	Default Background image
*LW3CLNTCOLOR	Default Background color
*LW3COLOR1	Default Left/Horizontal Menu Background
*LW3RMENUBKGND	Default Right Menu Background color
*LW3IMGMENUSELECT	Selection image for process menu
*LW3MENSEP	Standard Menu - Item Separator image
*LW3CPYLOGO	Company logo
*LW3IMGMESSAGES	Image used with messages
*LW3WIZHELP	Wizard Help File Location

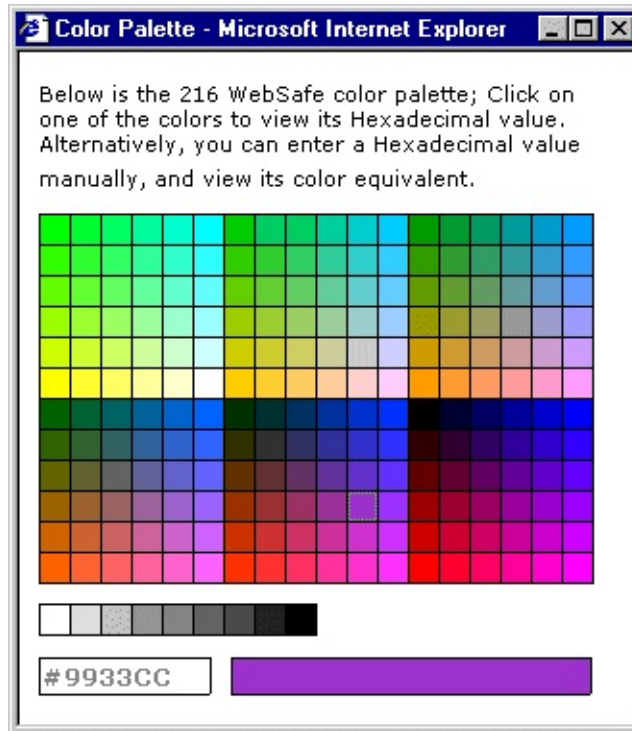
The values *LW3CLNTBKGND, *LW3CLNTCOLOR, *LW3COLOR1, *LW3RMENUBKGND, *LW3IMGMENUSELECT and *LW3MENSEP can be overridden using process specific graphic variables.

*LW3COLOR1 is an Web Functions Wizard-specific system graphic variable. It will only work correctly as a system graphic variable in components generated by the Web Functions Wizard.

For more details, refer to [Graphic Variables](#).

7.2 Color & Color Picker

The Web Functions Wizard provides a Color Picker which displays a list of the 216 browser-safe colors and a gray scale palette. Simply select the color of your choice from the color palette displayed.



After you select a color, the color number will be displayed in the appropriate field. The *Color* button is a helpful method of verifying your color choice. Press the *Color* button to view the color you have specified in the related field.



You may manually enter colors using their names or their numbers. For example, you may enter "red" or #FF0000.

When a color graphic variable is set to blank, LANSA for the Web will ignore this value and look for another default. A blank color is treated as if the variable does not exist.

For additional information, refer to [7.2.1 Tips & Techniques](#).



[WIZ002 - Customizing Browse List Variables](#)

7.2.1 Tips & Techniques

- It is possible to specify up to 16 million different colors. However, you can see significant variation in colors as they are used in different countries and different browsers.
- The color picker provides 216 browser-safe colors and a gray scale palette. You should try to use these colors for the majority of your presentations.

7.3 Image

The *Image* button can be used to display the specified image file and to check if the file exists in the image location configured for LANSAs for the Web.

Refer to the Installing LANSAs on Windows Guide for the default image file locations.



When specifying the file name, you do not need to specify the default directory names or path. For example, you should simply enter daisy.gif instead of /images/daisy.GIF.

If the file does not exist in the image file location, a picture will not be returned.



Once you have specified an image file, you must always enter a value in this field. To reset this field, you must manually delete the graphic variable using the *Web Function Editor*. Do not set the field to blank characters otherwise your application will receive "a file not found" error.

For additional information, refer to [7.3.1 Tips & Techniques](#).



WIZ002 - Customizing Browse List Variables

7.3.1 Tips & Techniques

- Remember that the size of the images used will impact the performance of your Web applications.
- If you wish to see a list of the image files available, use [File Search](#) to review the files stored in the image location specified on the Web server.
- Once you have specified an image file, the Wizard creates an image file graphic variable. To reset this field, you must manually delete the graphic variable using the Web Function Editor. Do not set the field to blank characters otherwise your application will receive a file not found error. Your application will search for a blank file name.

8. Customizing Browse Lists

The Web Functions Wizard will present all the LANSAs for the Web graphic variables associated with a browse list in a single screen so that you can quickly and easily identify your changes.

The browse list name is the name that you have used in your RDML function. For example, if you have used a `DEF_LIST NAMED(#LIST01)` command, then you will need to modify the list LIST01 to change how this list is displayed.

A browse list named DEFAULT will always be included in your list of browse list names. The DEFAULT browse list will update the system default graphic variables for browse lists. You should not use DEFAULT as a browse list name in any of your functions.

Browse list customization will apply to all browse lists with the same name anywhere in your LANSAs system. You should carefully consider your browse list names when designing your Web functions. Implementing naming standards for browse lists is very helpful.

When you [8.1 Select Browse List](#) characteristic from the Wizard Home page, a specific Process does not need to be selected. The list of browse lists available to be reviewed will be based on all browse lists used in RDML functions which are included in the processes available in the Process drop down (not just the current process). To review all browse lists in your Web Function Application, it is useful to Define a Process Group based on the processes which make up your Web Function Application. For more details, refer to [Define Process Group](#).

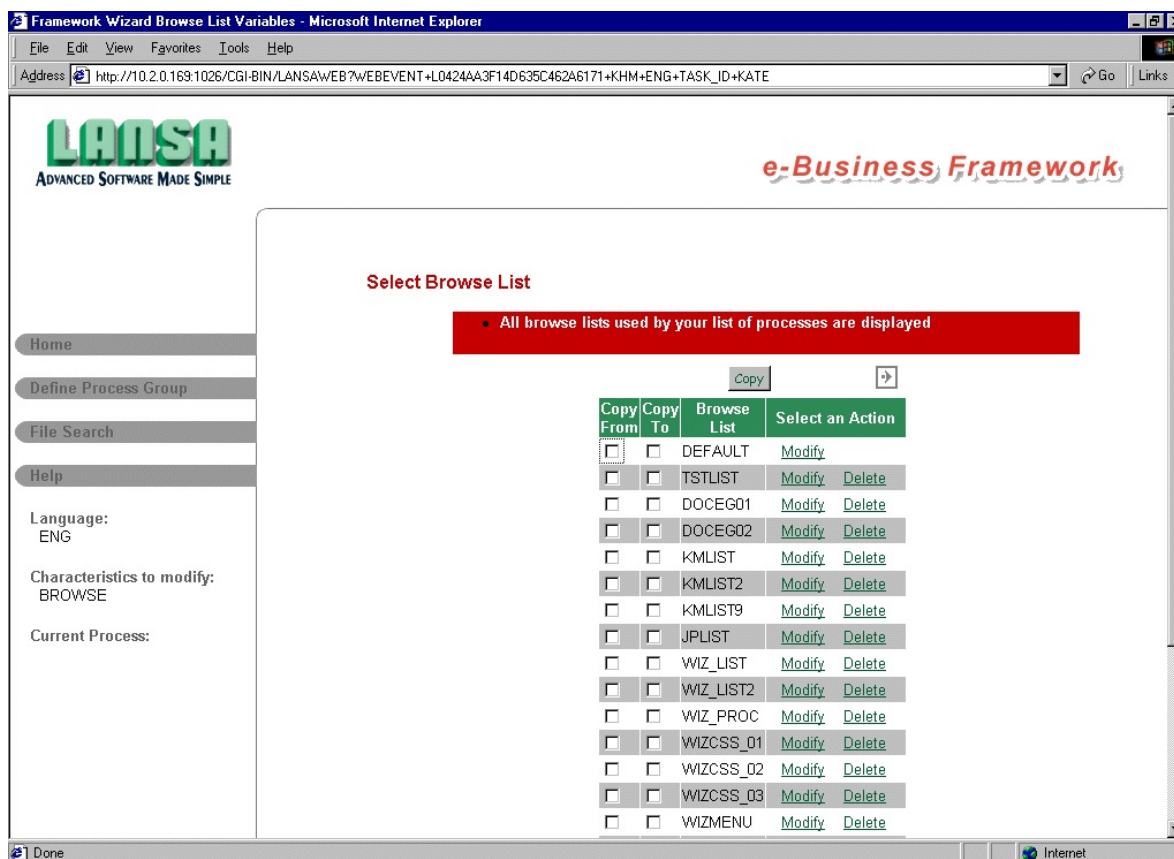


WIZ002 - Customizing Browse List Variables

Reminder: The changes made will impact browse lists in all partitions in your LANSAs system. Refer to [Partition Wide versus System Wide](#) for information.

8.1 Select Browse List

Once you select the Browse Lists characteristic and press Go on the [Wizard Home Page](#) (no process selection is required), you will be presented with the *Select Browse List* window.



The list displayed is based on your Process Group selection. If no Process Group has been defined, the list will include all the browse lists defined in the RDML of all of the Web-enabled functions in the current partition.

Use the Modify link associated with a specific browse list to review the browse list graphic variables.

Use the Delete link associated with a specific browse list to delete the associated graphic variables. If a browse list does not have specific graphic variables associated with the list, then the DEFAULT list values will be used. This option is not available for the browse list DEFAULT as these are the system default values.

To Copy the graphic variable values associated with a list to one or more other browse lists, check the list to be copied from in the "Copy From" column then

check all lists the values are to be copied to in the "Copy To" column. When you have selected all the required lists, press the Copy button at the top or bottom of the page.

Use the forward and back arrows at the top and bottom of the list of names to review more browse lists. These arrows will only be displayed if there are more browse list names to view.

For more details, refer to [8.2.5 Tips & Techniques](#) and [8.2.6 Technically Speaking](#) .

8.2 Maintain Browse List

Once you have selected a browse list to modify, the associated graphic variable values are shown. The Web Functions Wizard describes the browse list graphic variables with easy-to-understand descriptions.

You can configure all the graphic variables associated with a browse list including:

- [8.2.1 Browse List Colors and Column Headings](#)
- [8.2.2 Images for Browse List Selection](#)
- [8.2.3 Miscellaneous Attributes](#)
- [8.2.4 Presentation Attributes](#)

Maintain WIZ_LIST2 Browse List

• **WARNING: Any modifications to a browse list variable will apply to all lists with this name in all partitions**

Apply Preview

Browse List Colors Top Bottom

Background color for odd numbered rows	white	Color	Color Picker
Foreground color for odd numbered rows	black	Color	Color Picker
Background color for even numbered rows	silver	Color	Color Picker
Foreground color for even numbered rows	black	Color	Color Picker

Column Headings Top Bottom

Background color for column headings	seagreen	Color	Color Picker
Foreground color for column headings	white	Color	Color Picker

Images Top Bottom

Selection image for this browse list	ballshad.gif	Image
--------------------------------------	--------------	-------

Miscellaneous Attributes Top Bottom

HTML Attributes used in conjunction with the <table> tag

Presentation Attributes Top Bottom

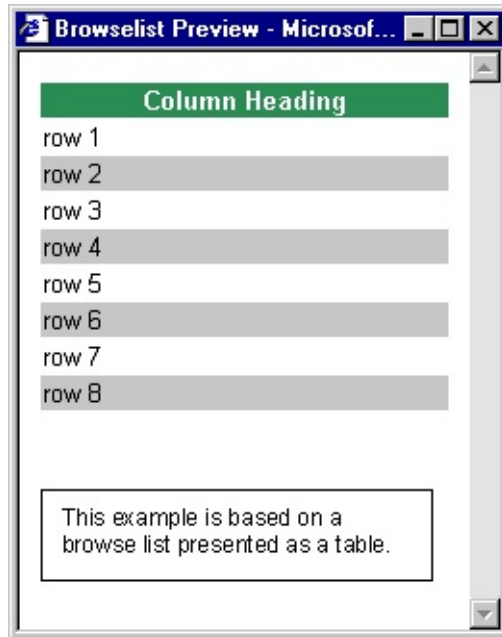
- Table Borders
- Column Headings
- Selection Image
- Table Tags

You do not have to configure all the variables displayed. Any variables left blank will adopt the system wide definitions or the default values assumed in LANSA for the Web.

Notice that the browse list name you selected to maintain is included in the screen heading for easy verification.

When you have finished specifying values, press the Preview button to see a

sample presentation based on your graphic variable selection. If you select table tags in the *Presentation Attributes* your browse list will not be presented as an HTML table and many of your browse list variables will be ignored.



When you are satisfied with your browse list graphic variable values, press the Apply button to update the variables.

Remember, when you are modifying the browse list named DEFAULT, you are updating the system default graphic variables for browse lists.

8.2.1 Browse List Colors and Column Headings

For a specific browse list the Web Functions Wizard allows you to set the following colors:

- Background color for odd numbered rows
- Foreground color for odd numbered rows
- Background color for even numbered rows
- Foreground color for even numbered rows
- Background color for column headings
- Foreground color for column headings.

You can manually enter values for colors, by using the color name (e.g. red) or by using their hex code values (e.g. #FF0000). Alternatively, you may use the [Color Picker](#) to select from a list of 216 browser-safe colors and a selected palette of grays. The [Color button](#) will display the color you have specified in the related field.

The Wizard does not validate your entries. You should use the Color or Image buttons to confirm your selections.



[WIZ002 - Customizing Browse List Variables](#)

8.2.2 Images for Browse List Selection

For a specific browse list the only image to maintain is the:

- Selection Image for Browse Lists

The image for browse list selection will only be used with procedural functions. This field is ignored if the browse list is used with WEBEVENT functions. For more details, refer to the *Web Functions* section of the web documentation.

The *Image* button will display the specified image, if it exists in the image location configured for LANSA for the Web. To locate an image use the [File Search](#) feature in the left menu, being careful to modify the filepath as required when copying this to the image related graphic variable.

The browse list selection image is usually a very small graphic file. You should select an image based on the content of your browse list rows and columns. There are a number of pre-shipped images, which can be used. For example, BALL_BLUE.GIF or BALLSHAD.GIF are good image files.

Once you have specified an image file, you must always enter a value in this field. To reset this field, you must manually delete the graphic variable using the Web Function Editor. Do not set the field to blank characters. If a value is not entered, your Web pages will fail to locate and load the file.



[WIZ002 - Customizing Browse List Variables](#)

8.2.3 Miscellaneous Attributes

The Miscellaneous Attributes is a free-format entry where you may specify the attributes of the HTML <TABLE> tag including border, cellpadding, cellspacing, align and border. These attributes will change the appearance of the HTML table used to present the browse list.

The values entered will override the default table definitions. For example, if you enter border="10" cellspacing="3" cellpadding="5" align="left" bgcolor="pink", your browse list will appear as a table with this tag: <table border="10" cellspacing="3" cellpadding="5" align="left" bgcolor="pink">

Be sure to enter these attributes in lowercase characters with value enclosed in quotes.

If you check the Table Tags option in the Presentation Attributes these values will not be applied as the browse list will not be implemented as an HTML table.



[WIZ002 - Customizing Browse List Variables](#)

8.2.4 Presentation Attributes

For a specific browse list the Web Functions Wizard allows you the Presentation Attributes:

- Table Borders
- Column Headings
- Selection Image
- Table Tags

The Presentation Attributes are controlled by a set of check boxes. The default is for all tags to be unchecked.

If you want to remove the border around the browse list check *Table Borders*. When you check the *Table Borders* option any reference to borders in the [8.2.3 Miscellaneous Attributes](#) will no longer apply.

If you want to remove the column heading in the browse list check *Column Headings*.

If you want to remove the selection image in the browse list check *Selection Image*. *Selection Images* will only be displayed in browse lists in procedural functions.

If you do not want a <TABLE> HTML tag around your browse list check the *Table Tags* option. If you check *Table Tags* the [8.2.3 Miscellaneous Attributes](#) will no longer apply.

Carefully test your application if you change the Presentation Attributes. If the *Table Tags* option is checked, the data will not be presented in a table and the Miscellaneous Attributes have no influence on the presentation.



8.2.5 Tips & Techniques

To simplify the maintenance of all the browse lists in your Web Function Application, it is recommended that you use the [Define Process Group](#) to set the boundary of your Web Function Application. When you select the Browse List characteristic from the Home page and press Go, all the browse lists included in your Process Group will automatically be displayed. You can then customize a single browse list. When you are satisfied with the browse list presentation, you can standardize all browse lists in your Web Function Application by copying the original browse list to all other browse lists in your process group.

- Browse lists which are presented as an HTML <TABLE> may also be impacted by your style sheet definitions.
- If the Maintain System Variable screen includes or excludes values, the details stored in the LANSAs internal file DC@W33 have been modified since your last LANSAs upgrade or install.
- When a color graphic variable is set to blank, LANSAs for the Web will ignore this value and look for another default. A blank color is treated as if the variable does not exist.
- To reset the selection images for a browse list, you must delete the *LW3BLI_<list name> graphic variable using the Web Function Editor.
- Including hyperlinks in a browse list element will overwrite the associated font color settings.
- The image for browse list selection will only be used with procedural functions. It will not be used with browse lists in WEBEVENT functions.
- You should think carefully about the names of the browse lists used in your RDML functions. Implementing browse list naming standards will help manage your application development.
- The FRWEBENQ01 template allows you to define the browse list name by entering a unique 2 character identifier. If you are creating templates for Web functions, you should use a similar approach to ensure unique browse list names are generated.
- Use the Preview button for a general idea of how your browse list will appear in your Web Function Application. For a more accurate representation of your browse list, you need to run your application and view it in the context it has been used.



8.2.6 Technically Speaking

You are specifying the values for these graphic variables:

DEFAULT browse list	Other Browse List	Description
*LW3BLACELLBCOLOR	*LW3BLACB_<list name>	background color alternate row
*LW3BLACELLFCOLOR	*LW3BLACF_<list name>	foreground color alternate row
*LW3BLCCCELLBCOLOR	*LW3BLCCB_<list name>	background color for row name>
*LW3BLCCCELLFCOLOR	*LW3BLCCF_<list name>	foreground color for row name>
*LW3COLHDGBCOLOR	*LW3BLBC_<list name>	background color for column heading
*LW3COLHDGFCOLOR	*LW3BLFC_<list name>	foreground color for column heading
*LW3BRWLSTSELIMAGE	*LW3BLI_<list name>	selection image
*LW3BRWLSTMISC	*LW3BLMS_<list name>	miscellaneous attributes
*LW3BRWLSTSELECT	*LW3BL_<list name>	presentation attributes

where the <list name> is the name you specified for the browse list in your RDML function.

For more details, refer to the [Browse List Graphic Variables](#).

9. Application Import/Export

After using the Web Functions Wizard to customize the presentation interface for your Web function application, you will need to include the objects generated by the Wizard when you package or export the Web function application to a production environment.

If you intend to run the Wizard in another LANSAsystem you may choose to export your current Wizard configuration (i.e. specific process pre-selections) to the other LANSAsystem. When you install the Wizard, you have a standard configuration with no information about previous process selections. For example, if you run the Wizard over process TEST01 and build the layout based on the first layout schema, the graphic variable *LW3LAYOUT will be set to WIZLAYO01. Next time you view the Select Layout page, the first schema will be pre-selected as the current value.

- [9.1 What Do I Need to Export?](#)
 - [9.1.1 Export/Import Between Systems](#)
 - [9.1.2 Export/Import Between Partitions in the Same System](#)
- [9.2 Exporting Browse List Changes](#)
- [9.3 Exporting Process Changes](#)
- [9.4 Exporting System Wide Changes](#)
 - [9.4.1 Exporting Wizard Runtime Objects](#)
 - [9.4.2 Exporting Customized Wizard Settings](#)
- [9.5 Import/Export Considerations](#)

9.1 What Do I Need to Export?

The Web Functions Wizard creates objects at both the system and the partition levels. The objects you need to export will depend on how the export is being used. There are two different types of exports and imports which you need to consider:

1. Export from one LANSAsystem and import into a different LANSAsystem, i.e. [9.1.1 Export/Import Between Systems](#).
2. Export from one partition and import into another partition in the same system, i.e. [9.1.2 Export/Import Between Partitions in the Same System](#).

Your export list will depend upon the type of export and import you are performing.

If your Web function application is intended to run in a non-multilingual partition, you should execute the Wizard over your application in a non-multilingual partition. If your Web function application is intended to run in a multilingual partition, you should execute the Wizard over your application in a multilingual partition. The Wizard is only supported as a non-multilingual product on LANSAsystem for iSeries.

9.1.1 Export/Import Between Systems

If you are exporting your application for import into a completely separate LANSa system, your export list must include all graphic variables (system variables) and Web components created by the Wizard.

If exporting from a multilingual partition, ensure you include all languages required in your target partition.

For a list of Web objects, refer to [9.2 Exporting Browse List Changes](#), [9.3 Exporting Process Changes](#) and [9.4 Exporting System Wide Changes](#) .

9.1.2 Export/Import Between Partitions in the Same System

If you are exporting from one partition and importing into another partition which is part of the same LANSAs system, then you will not be required to export any graphic variables.

Graphic variables use LANSAs system variables which are common to all partitions.

You will need to export all layout pages and Web components you have created while executing the Wizard as these objects are partition specific.

If exporting from a multilingual partition, ensure you include all languages required in your target partition.

For a list of Web components, refer to [9.3 Exporting Process Changes](#) and [9.4 Exporting System Wide Changes](#) .

9.2 Exporting Browse List Changes

If you are moving to a different LANSAs system, you may need to export some or all of the browse list variables:

- *LW3BLACB_<list name> background color alternate row
- *LW3BLACF_<list name> foreground color alternate row
- *LW3BLCCB_<list name> background color for row
- *LW3BLCCF_<list name> foreground color for row
- *LW3BLBC_<list name> background color for column heading
- *LW3BLFC_<list name> foreground color for column heading
- *LW3BLI_<list name> selection image
- *LW3BLMS_<list name> miscellaneous attributes
- *LW3BL_<list name> presentation attributes

where <list name> is the name you specified for the browse list.

Again, if you are moving your LANSAs application to another partition in your current LANSAs system, then you do not have to export any browse list changes made by the Wizard because the graphic variables are defined at the system level.

9.3 Exporting Process Changes

If you are exporting a LANSAs Web function application which has been configured with the Web Functions Wizard, your process specific Web components will be automatically exported with the process definition. These components might include:

- <process name>_LAYOUT
- <process name>_LMENU for the *left* menu component
- <process name>_LMENU1 for the *standard* menu component
- <process name>_LMENU2 for the *DHTML left* menu component
- <process name>_RMENU for the *right* menu component
- <process name>_HMENU for the *horizontal* menu component
- <process name>_HMENU1 for the *standard horizontal* menu component
- <process name>_HMENU2 for the *DHTML horizontal* menu component
- <process name>_STYLE for *style sheets* used in a layout
- <process name>_STDBANNER for *banner* components used in layouts

where:

<process name> is the name of your LANSAs process used with the Wizard.

When you export a process, all process related system variables with the prefix *LW3P will be added to the export list. Remember, you do not have to export any process specific graphic variables unless you are planning to import into a partition in a different LANSAs system. Modify the objects to be exported as required. If you are moving to a different system, you may need to export some or all of these variables:

*LW3PMSP_<process name> Image for menu separator.

*LW3RMBG_<process name> Color setting for Right Menu Component

*LW3PBGC_<process name> Default background color for the Client area.

*LW3PBGI_<process name> Background image for the Client (body) area

*LW3PCLR1_<process name> Default Layout Left/Horizontal Menu Color

Where the <process name> is the name you specified for a specific process. Refer to [9.4.2 Exporting Customized Wizard Settings](#) for other process graphic variables you may choose to exclude from your export.

Note: *LW3PCLR1_<process name> is specific to the Web Functions Wizard and is not supported in standard LANSAP Web function applications.

If your Web function application includes a DHTML menu, you must export the component WIZMNUSCR. Refer to [9.4.1 Exporting Wizard Runtime Objects](#).

9.4 Exporting System Wide Changes

If you have customized the default layout, you may need to export some or all of these Web components:

- DEFAULT_LAYOUT
- DEFAULT_LMENU for the left menu component
- DEFAULT_LMENU1 for the standard menu component
- DEFAULT_LMENU2 for the DHTML left menu component
- DEFAULT_RMENU for the right menu component
- DEFAULT_HMENU for the horizontal menu component
- DEFAULT_HMENU1 for the standard horizontal menu component
- DEFAULT_HMENU2 for the DHTML horizontal menu component
- DEFAULT_STYLE for banner components used in layouts
- STDBANNER for the default banner component.

DEFAULT_LMENU1, DEFAULT_LMENU2, DEFAULT_HMENU1 and DEFAULT_HMENU2 are specific to the Web Functions Wizard and are not supported in standard LANSAs Web function applications.

Warning: Use extreme care when exporting the DEFAULT browse list variables as these values may impact other Web function applications running in your target system.

Remember, you must create Web page component definitions for the default layout pages and menu component pages before you can add these pages to your export list. A quick method of exporting all of your DEFAULT pages is to create a dummy process named DEFAULT. When you export the DEFAULT process, all the DEFAULT_<component> Web components are automatically exported with the process and you do not need to register the Web components.

Before you import these default layouts into another partition or system, be sure to review [Impact on Existing Applications](#).

Remember, you do not have to export any system wide graphic variables unless you are planning to import into a partition in a different LANSAs system. If you are moving to a different system, you may need to export some or all of these variables:

Browse Lists Defaults

--	--

*LW3BLACELLBCOLOR	background color alternate row
*LW3BLACELLFCOLOR	foreground color alternate row
*LW3BLCCCELLBCOLOR	background color for row
*LW3BLCCCELLFCOLOR	foreground color for row
*LW3COLHDGBCOLOR	background color for column heading
*LW3COLHDGFCOLOR	foreground color for column heading
*LW3BRWLSTSELIMAGE	selection image
*LW3BRWLSTMISC	miscellaneous attributes
*LW3BRWLSTSELECT	presentation attributes

System Wide Defaults

*LW3CLNTBKGND	client background image
*LW3CLNTCOLOR	background color client area
*LW3CPYLOGO	company logo
*LW3IMGMESSAGES	images used with messages
*LW3IMGMENUSELECT	selection image for process menu
*LW3MENUCOLOR	background color for menu area
*LW3MENUSEP	menu separator image
*LW3RMENUBKGND	right menu background color
*LW3COLOR1	Default Layout Left/Horizontal Menu Color

*LW3COLOR1 is specific to the Web Functions Wizard and is not supported in standard LANSAs Web function applications.

Again, if you are moving your LANSAs application to another partition in your

current LANSAs system, then you do not have to export any graphic variables made by the Wizard because the graphic variables are defined at the system level.

9.4.1 Exporting Wizard Runtime Objects

If your Web function application includes any DHTML menus you must include the component WIZMNUSCR in your exported Web function application. This component contains the generic JavaScript required to execute the DHTML menu.

9.4.2 Exporting Customized Wizard Settings

A number of graphic variables are used internally by the Wizard to store the previous selections associated with a process. They will be automatically added to the export list if you export the associated process definition.

These customized wizard settings are only used when running the Wizard over the associated process.

There is no equivalent system wide graphic variable for the values. If you use the Web Functions Wizard to modify process DEFAULT, an associated set of graphic variables will be defined but these values will not be used as default values for other processes.

- *LW3PLAY_<process name> Default Layout selection
- *LW3PLIN_<process name> Default Layout Inclusion/Exclusion flags
- *LW3PRMS_<process name> Include Right Menu Separator
- *LW3PLMS_<process name> Include Left Menu Separator
- *LW3PDHM_<process name> DHTML Menu Support Required
- *LW3PDST_<process name> Default Button/Tab Style
- *LW3PDCL_<process name> Default Button/Tab Color
- *LW3PDC01_<process name> Low Background Color
- *LW3PDC02_<process name> Low Sub-Menu Background Color
- *LW3PDC03_<process name> High Background Color
- *LW3PDC04_<process name> High Sub-Menu Background Color
- *LW3PDC05_<process name> Font Low Color
- *LW3PDC06_<process name> Font Sub-Menu Low Color
- *LW3PDC07_<process name> Font High Color
- *LW3PDC08_<process name> Font Sub-Menu High Color
- *LW3PDC09_<process name> Border Color
- *LW3PDC10_<process name> Border Sub-Menu Color

*LW3PDT01_<process name> Border Width
*LW3PDT02_<process name> Border Between Elements
*LW3PDT03_<process name> Font Family
*LW3PDT04_<process name> Font Size
*LW3PDT05_<process name> Font Bold
*LW3PDT06_<process name> Font Italic
*LW3PDT07_<process name> Menu Text Centered
*LW3PDT08_<process name> Menu Centered
*LW3PDT09_<process name> Menu Vertical Centered
*LW3PDT10_<process name> Child Overlap
*LW3PDT11_<process name> Child Vertical Overlap
*LW3PDT12_<process name> Start Top
*LW3PDT13_<process name> Start Left
*LW3PDT14_<process name> Left Padding
*LW3PDT15_<process name> Top Padding
*LW3PDT16_<process name> Disappear Delay
*LW3PDT17_<process name> Menu Wrap
*LW3PDT18_<process name> Right To Left
*LW3PDT19_<process name> Unfolds on Click
*LW3PDT20_<process name> Show Arrows
*LW3PDT21_<process name> Keep Highlight
*LW3PDT22_<process name> Menu Element Height
*LW3PDT23_<process name> Menu Element Width

These graphic variables are specific to the Web Functions Wizard and are not supported in standard LANSAs Web function applications.

9.5 Import/Export Considerations

Following are some important considerations when importing and exporting your LANSAs applications, which have been customized using the Web Functions Wizard.

- A LANSAs system must be at Release 8.0 and EPC538 (or greater) in order to use an application created with the Web Functions Wizard. In general, you must always import into a system which is at the same level or higher than the exporting system. In particular, you must be very careful with changes to the Web skeleton when applications are moved to another system and then recompiled.
- The Web Functions Wizard import (LWEBWIZ2) does not have to be installed in a partition which is used only for executing an application.
- If you are planning to import LWEBWIZ2 into a partition being used for development, be sure to execute this import before you execute your application import, otherwise LWEBWIZ2 may overwrite your application import.
- Remember that graphic variables are system wide. When you import into a system, the graphic variables will impact more than one partition.
- You are responsible for moving any custom image files used by your applications.
- If your Web function application is intended to run in a non-multilingual partition you should execute the Wizard over your application in a non-multilingual partition. If your Web function application is intended to run in a multilingual partition you should execute the Wizard over your application in a multilingual partition.
- If you are importing applications which were created on a LANSAs system which does not use the Wizard, refer to [Impact on Existing Applications](#).

Tip:

You can create an export of your system variables as a method of backing up your system configuration. If you are using Version 9.x of the Web Functions Wizard, and you are importing into a Version 10.0 or greater system, please refer to [Existing LANSAs for the Web Installation](#).

10. Troubleshooting

Find the solution to these common problems:

- [10.1 Wizard does not show the most current layout definition](#)
- [10.2 Wizard does not change the presentation layout of a function](#)
- [10.3 Company Logo does not appear in the presentation layout](#)
- [10.4 Banner does not appear in the presentation layout](#)
- [10.5 Click Tracking does not log events](#)
- [10.6 Menu Component does not appear in a function](#)
- [10.7 Image file is not found when executing your application](#)
- [10.8 Presentation controlled by graphic variables has been changed but you did not alter the variables](#)
- [10.9 Application function or linked page does not Start in a New Window](#)
- [10.10 Not authorized to change a graphic variable when using the Web Function Editor](#)
- [10.11 Changes to a graphic variable are not committed when using the Wizard](#)
- [10.12 Standard Menu Items do not align correctly](#)
- [10.13 Online Help is not accessible from the Wizard](#)
- [10.14 Curve image in corner of presentation layout is not displayed](#)
- [10.15 Web Application does not include Wizard Interface in all supported Languages](#)
- [10.16 Web Application does not include Wizard Interface when exported to a non-Multilingual partition](#)

Refer to [10.17 Other Resources](#) for additional information about the Web Web Functions Wizard.

10.1 Wizard does not show the most current layout definition

While executing the Wizard multiple times, you may not see the most current information. When you access a screen, it shows the old data. Your last changes are not shown.

While executing the Wizard, your latest changes seem to be ignored and the old settings are being used instead.

Solution

It is very important that you have your browser properly configured when using the Web Functions Wizard. You do not want to be using cached images when you customize your application as your browser will display old settings. You should set your browser to retrieve the most current page each time a page is accessed using the Wizard.

If you are using an Internet Explorer browser (recommended), be sure to edit your Internet Settings for Temporary Internet files as described in [Browser Configuration](#).

That is, your browser must check for newer versions of stored pages: Every visit to the page.

Document in cache must be compared to document on network: Every time.

10.2 Wizard does not change the presentation layout of a function

You have created a process layout or modified the default layout, but the functions are not using the new layout or components.

Solution

If you have older functions (created before installing LANSAs Release 8.0 EPC538), these functions do not include the commands to include the Wizard layouts. Review the HTML page for your function and search for a <RDML LAYOUT> tag. If this tag is not present, recompile your function so that the HTML page is regenerated. The proper tags will be included.

If you have created your own skeletons in file DC@W22 and you wish to use the Web Functions Wizard for these functions, you may need to modify your skeletons according to the default shipped skeleton (WEBSKEL in file DC@F28 in the LANSAs data library).

If you are experiencing this problem with a newly created function, use the Web Function Editor and check that the correct LAYOUT pages exist for the process in question. If you do not see the <process name>_LAYOUT or DEFAULT_LAYOUT page, execute the Wizard again.

10.3 Company Logo does not appear in the presentation layout

You have elected to include the Company Logo using the Layout Features then built your layout, but your function does not show a company logo when it executes.

Solution

You must define the Company Logo by modifying the appropriate System Variable in the Wizard. Ensure an image file has been entered and that the file exists.

Use the Web Function Editor and check the correct LAYOUT page exists, <process name>_LAYOUT or DEFAULT_LAYOUT. Review the HTML and look for the tag <RDML SETTMPFLD="14" VALUE="1"> near the top of the page. If this tag does not appear or the value is not set to "1", then you have not included the component. Use the Wizard to include the missing Layout Feature then rebuild your layout.

10.4 Banner does not appear in the presentation layout

You have elected to include a Banner using the Layout Features then built your layout, but your function does not show a banner when it executes.

Solution

You must create a banner Web component named STDBANNER. Use the Web Function Editor to create the component. You can also create a process specific banner named <process name>_STDBANNER.

Use the Web Function Editor and check the correct LAYOUT page exists, <process name>_LAYOUT or DEFAULT_LAYOUT. Review the HTML and look for the tag <RDML SETTMPFLD="13" VALUE="1"> near the top of the page. If this tag does not appear or the value is not set to "1", then you have not included the component. Use the Wizard to include the missing Layout Feature then rebuild your layout.

For more details, refer to [Banner](#).

10.5 Click Tracking does not log events

You have elected to include a Click Tracking using the Layout Features then built your layout, but there are no details in your trace files (DC@W26 and DC@W27) after running your Web function application.

Solution

Create a component `<process name>_TRACE` or `DEFAULT_TRACE` with the appropriate `TRACE_DETAILED` or `TRACE_SUMMARY` statement embedded in the page.

Use the Web Function Editor and check the correct `LAYOUT` page exists, `<process name>_LAYOUT` or `DEFAULT_LAYOUT`. Review the HTML and look for the tag `<RDML TRACE>` near the top of the page. If this tag does not appear, then you have not included the component. Use the Wizard to include the missing Layout Feature then rebuild your layout.

For more details, refer to [Enabling Event Logging](#).

10.6 Menu Component does not appear in a function

You have created a left, right or horizontal menu component but it does not appear in your presentation layout.

Solution 1

When you define your presentation layout, you must specify which menu components to include, and for Left and horizontal Menus you must specify whether a Standard or DHTML Menu should be included. You can define a menu component but not include it in your layout. Review the HTML page for your function and search for these tags:

<RDML SETTMPFLD="10" VALUE="1"> for the left menu component

<RDML SETTMPFLD="11" VALUE="1"> for the horizontal menu component

<RDML SETTMPFLD="12" VALUE="1"> for the right menu component

If these tags do not appear or the value is not set to "1", then you have not included the component. Use the Wizard to change the layout and include the missing component.

If you have defined a Standard menu, ensure your Layout Features is checked to include the appropriate Standard menu.

If you have defined a DHTML menu, ensure your Layout Features is checked to include the appropriate DHTML menu.

If the missing menu is a DHTML menu, ensure you partition include the component WIZMNUSCR.

The <process name>_LAYOUT component used by your application must be generated by the Wizard to display menus which have been generated by the Wizard. For more details, refer to [Customizing a Presentation Layout](#) to understand the component architecture used by the Web Functions Wizard.

Solution 2

If your presentation layout includes a DHTML Menu with options referencing a component, ensure each component is defined in the Web Function Editor's "Maintain Components" as a script.

Further check your component is a simple single line definition without any line breaks.

Notes:

<RDML COMPONENT> defined in "Maintain Components" as a script will

NOT force a line break.

<RDML COMPONENT> defined in "Maintain Components" as a page will force a line break.

10.7 Image file is not found when executing your application

When you execute your Web function application, you are receiving an error saying that an image cannot be found. When you check the Web Functions Wizard, the relating image file name is blank. It appears as if no file has been specified.

For example, in your finished application the Company Logo image cannot be found.

Solution

View your HTML source from the browser to see what file name is being used. If no file name has been specified, then a blank file name has been specified in the Wizard. You must enter a valid file name.

If an image has been entered, check that the image exists in the image location configured for LANSAs for the Web server. Use the [Image](#) button to verify the Image exists.

This situation also occurs when you specify a file name for an image and then change the field to blanks. Once a name has been specified, the graphic variable is created. If you change the file name to blank, it does not delete the variable but stores the blank value. If you do not want the process specific image used, you must manually delete the associated graphic variable using the Web Function Editor.

10.8 Presentation controlled by graphic variables has been changed but you did not alter the variables

While testing your application, you notice that the presentation of browse lists or logos, etc. has changed but you have not altered the graphic variables.

Solution

Check the current value of any process specific and default graphic variables relating to the presentation problem. The variables may have been changed by another developer.

You must remember that graphic variables are defined at the system level. If another developer in a different partition has modified a graphic variable using the Web Functions Wizard or the Web Function Editor, it will impact your application. Also, if any import containing graphic variables has been made into the system, then all partitions will be impacted.

10.9 Application function or linked page does not Start in a New Window

When creating a standard menu item using the Wizard, you selected the option to *Start in a New Window*. When you execute the menu item, a new browser window is not started. The linked Web page or linked function executes in the same window.

Solution

Check the definition of the menu item to make sure that you have checked the *Start in a New Window* option for the URL or LANSAs function as required. If you check the *Reuse Job* option, then the linked page or function will not start in a new window. In order for the same job to be reused, the function must execute in the same window. Only the URL or LANSAs function links can be started in a New Window.

10.10 Not authorized to change a graphic variable when using the Web Function Editor

When attempting to use the Web Function Editor to manually change or delete a graphic variable, which was created using the Web Functions Wizard, you received the following message:

"You are not authorized to change system variable ..."

Solution

When you execute the Wizard, you create LANSAs graphic variables, which use LANSAs system variables. The owner of these system variables is based on the user profile used when executing the Wizard.

If you wish to edit these graphic variables using the Web Function Editor, you must have authority to the system variables. The owner may grant authority to change/delete these variables, or you may logon with the Web Function Editor using the owner's profile.

To determine the owner of the system variables, you may use the LANSAs Housekeeping options and review the access to the LANSAs objects.

To determine the user profile being used when you execute the Wizard, refer to Wizard Security Configuration following.

If Web Server authentication has been set up, then you will enter a Web Server profile and password, and you should be able to determine the application/date server user profile being used when you execute the Wizard. If no authentication has been configured, then anonymous user access is being used. You must determine the profile associated with the DFTUSR.

Wizard Security Configuration

The LANSAs for the Web security and authentication rules apply to the Wizard processes and functions. If you do not have security defined for your LANSAs for the Web system, you may wish to set up process level authentication so that you have controlled access to the Wizard. (Note: The LANSAs for the Web Administrator includes user profile and password security.)

To set up process level authentication, refer to [Configure LANSAs for the Web Security](#) in the *Installing LANSAs on IBM i Guide*.

You will want to secure this process:

WIZARD2

Note: You can also use task tracking with the Wizard. If you have task tracking

enabled, refer to [Task Tracking](#).

10.11 Changes to a graphic variable are not committed when using the Wizard

When running the Web Functions Wizard and attempting to apply changes you receive a message:

"Work performed on <graphic variable name> has not been committed"

Solution

The partition the Wizard is running is probably uses Task Tracking. Check that a task id has been included in the Wizard URL when you execute the Web Functions Wizard.

10.12 Standard Menu Items do not align correctly

The Standard menu generated by the Web Functions Wizard does not align correctly.

Solution

This is a particular problem with Standard Horizontal Menus. Ensure you are not using a combination of Button and Tab styles. If your menu includes images ensure the images are not larger than any buttons height (in horizontal menu) or width (in vertical menus). In vertical menus, the widest menu item will determine the size of any buttons displayed.

10.13 Online Help is not accessible from the Wizard

When you press the Online Help button in the left menu of the Wizard, you are not able to access the Web Functions Wizard documentation.

Solution

The location of the online documentation can be changed using the *LW3WIZHELP graphic variable. Check this value using either the Wizard characteristic System Variables or the Web Function Editor. Review the value of this variable. Check that the LANSAP Web Functions Wizard documentation exists in the specified location by entering the location into the URL of your browser.

The online help documentation can be copied from the installation CD-ROM.

10.14 Curve image in corner of presentation layout is not displayed

While executing your Web function application, the Layout Presentation does not display the curved corner image.

Solution

Curved corner images are included in your Layout component if you select Layout Schema 1, 3 or 5 in combination with a Wizard Support color for the Left/Horizontal Menu.

If you originally build your layout using a Wizard Supported Color and Layout Schema 1, 3 or 5, the curved corner images are embedded in your <process name>_LAYOUT component. If you later change your color schema, and do not select a Wizard Supported Color, an appropriate image does not exist in the image location configured for LANSA for the Web server. Either rebuild your layout to exclude the references to the curved corners images, or change your color scheme and select a Wizard Supported Color.

If you originally build your layout not using a Wizard Supported Color and Layout Schema 1, 3 or 5, the curved corner images are not embedded in your <process name>_LAYOUT component. If you later change your color schema and do select a Wizard Supported Color, the <process name>_LAYOUT component is not automatically modified to include the references to images for the corners. Rebuild your layout to include the references to the curved corners images.

10.15 Web Application does not include Wizard Interface in all supported Languages

When executing your Web function application in different languages, the Wizard generated components do not execute consistently.

Solution

To run your Web function application in multiple languages, you must generate the appropriate components in the required languages. Generally, if you start by generating an interface in the default language, a component will always be found (it may not be in the correct language though). After you have created your default language interface, you only need to create separate menu components for each language as these are the only components which usually include text requiring translation (e.g. menu item descriptions).

Check that the default partition language where you are executing the Web function application is the same as the default partition language in the partition where the Wizard was executed. If they are different, use the Wizard to generate all the components in the default language of your target system.

10.16 Web Application does not include Wizard Interface when exported to a non-Multilingual partition

When executing your Web function application in non-multilingual partition, the Wizard generated components do not execute.

Solution

To run your Web function application in non-multilingual partition, you must execute the Wizard in a non-multilingual partition in order to generate components with a language of NAT.

For more details, refer to [Multilingual Versus Non-multilingual Objects](#).

10.17 Other Resources

You will find the most current product updates and additional information on the LANSA Web site located at www.lansa.com/support.

Refer also to the *Troubleshooting* sections in the *Installing LANSA on Windows* guide.

Contact your local LANSA agent for additional support.

Wizard Tutorials

What are the Wizard Tutorials?

The Web Functions Wizard Tutorials are a set of exercises designed to introduce and reinforce the fundamental skills required to use the Wizard.

To install the tutorials, refer to [Tutorial Installation](#).

The following tutorials are included:

- [WIZ001 - Executing the Wizard](#)
- [WIZ002 - Customizing Browse List Variables](#)
- [WIZ003 - Customizing a Presentation Layout](#)
- [WIZ004 - Defining a Standard Menu Component](#)
- [WIZ005 - Defining a DHTML Menu Component](#)
- [WIZ006 - Defining a Style Sheet](#)
- [WIZ007 - Adopting a Presentation Layout](#)

Who Should Use the Tutorials?

The Web Functions Wizard Tutorials can be used by novice and experienced LANSA for the Web developers. In order to use the tutorials, you must be able to create LANSA processes and functions. Experience with the LANSA for Web utilities is an asset.

How Do I Use the Tutorials?

It is recommended that you complete the tutorials in sequence. The processes and functions created in the first tutorials are reused in later tutorials.

To check that your LANSA partition is properly set up to use the tutorials, refer to the *Tutorial Installation* section of the installation guide for the platform you are working on.

Tutorial Installation

The tutorial is installed with the Wizard following the steps in the *Installing LANSA on Windows* guide or the *Installing LANSA on iSeries* guide.

Online tutorials are also included for the *Web Functions*.

What Partition Should I Use?

It is recommended that you use the DEM partition for the tutorial. The DEM is automatically installed with a LANSA for iSeries system and is quickly set up

using Partition Initialization on a Visual LANSA system. The DEM system contains the Personnel System demonstration and all required files used by the tutorial.

If you do not use the DEM partition, you can set up another partition with the Personnel System files. Remember, the partition must be properly **Web**-enabled.

If you do not use DEM, you can select any of your partitions when using the tutorials. Testing or training partitions are ideal. You will need to import the Personnel System demonstration files into the partition.

If you do not have the Web Functions Wizard or the Personnel System installed, then refer to the Installation guide for either IBM i or Windows, as appropriate.

How Many Developers Can Use the Training?

There is no limit to the number of developers who may use the training at the same time. However, it is important that developers have a unique identifier for their work.

Developers should use an object prefix *ii*, based on their initials or assigned by a system coordinator. For example, you will be asked to create a process named *iiPROC01*. If your initials are JS, you create a process named *JSPROC01*.

This *ii* prefix must be unique in the LANSA system (not just the partition) as it is used to create system variables. System variables are shared by all partitions.

Checklist

Check that these tasks have been completed before using the tutorials:

- Have you **Web** enabled the partition?
- Do you have the Personnel System demonstration files (DEPTAB, SECTAB, PSLMST, etc.) installed in the partition?
- Have you installed the Web Functions Wizard in the partition?

You will require this information about your specific installation:

- The site address for your **Web server**
- The partition being used
- A user profile and password with authority to the LANSA partition
- A user profile and password if process authentication has been set up.

Your Feedback

Your feedback regarding these tutorials will help us improve the overall quality of the LANSA documentation and training. Please email your comments to

lansatraining@LANSA.com.au

WIZ001 - Executing the Wizard

Objective:

- To configure your browser for using the Web Functions Wizard.
- To execute the Wizard from your browser to confirm that your system is properly set up.
- To access the online help text for the Wizard.

To achieve this objective, follow these steps:

- [Step 1. Configure Your Browser](#)
- [Step 2. Execute the Web Functions Wizard](#)
- [Step 3. Execute the Online Help](#)
- [Summary](#)

Before You Begin

You may wish to review these topics:

- [Browser Configuration](#)
- [Running the Web Functions Wizard](#)
- [Help Configuration](#)
- [Considerations for Using the Wizard](#)

In order to complete this tutorial, you must have installed the Web Functions Wizard in your partition.

Step 1. Configure Your Browser

In this step, you will configure your browser so that it always gets a new version of the Web page instead of using the cached version of the page.

Start the browser on your PC. The Internet Explorer browser, version 5.5 or greater, is recommended.

If you are using an Internet Explorer browser (recommended)

1. Select the *Tools* menu and choose *Internet Options*.
2. Click on the *Settings* button listed under *General* tab and *Internet Settings for Temporary files*.
3. For the option *Check for newer versions of stored pages*, select the *Every visit to the page* radio button.
4. Press *OK* to save your changes.

Step 2. Execute the Web Functions Wizard

In this step, you will execute the Wizard from your browser and display the welcome screen.

1. Enter the URL for the Wizard in this format:

```
http://<server domain name>/cgi-bin/lansaweb?procfun+wizard2+wiz02+<ppp>
```

where:

<server domain name> is the domain name or IP address of your Web server

<ppp> is the LANSAs partition where the Wizard is installed

- The Web Functions Wizard home page should appear as seen in the [Wizard Home Page](#).
Notice the drop down to select the Language to save as. The selected value will default to language nominated on the URL for the Wizard.
2. If the Wizard function does not execute properly, refer to [Troubleshooting](#). If you are asked to enter a profile and password, check with your LANSAs for the Web system administrator to get proper authority to access the Web system.
 3. **OPTIONAL:** You may wish to add a bookmark in your browser so that you can quickly execute the Wizard.

Step 3. Execute the Online Help

In this step, you will open the online help so that you can review the information about the wizard as you complete the tutorials.

1. In the left menu, you should see four options: Home, Define Process Group, File Search and Help.
2. Press the Help option. A new browser window will appear with a Help topic pre-selected based on your current window. If the online help does not execute properly, refer to [Troubleshooting](#).
3. You may wish to leave this window open as you complete the tutorials so that you can review any required information.
4. OPTIONAL: Check the directory where the Help files have been loaded by reviewing the value of the *LW3WIZHELP graphic variable.

Use the *Web Function Editor* to review the value of the *LW3WIZHELP graphic variable or use the Wizard System Variables option. Be sure that the location nominated for the Help files is accessible to all LANSAs developers using your LANSAs system.

Summary

Important Observations

- The Wizard is a LANSAs function. It is executed like any LANSAs for the Web function and follows the same rules as all LANSAs for the Web functions.
- By default, there is no security set up for the Wizard. You can use process authentication or other authentication options to secure the Wizard.

Tips & Techniques

- You can change the location opened when using the online help by setting the value of the *LW3WIZHELP system variable. Ensure the Help files are moved to the nominated file directory.
- You may wish to add process level authentication to the Wizard to prevent unauthorized users from modifying your application layouts.

What I Should Know

- Which browser configuration is required for the Web Functions Wizard.
- How to execute the Web Functions Wizard using a standard browser.
- How to set the language used to save the customizations,
- How to change the location of the online help.

Other Tutorials

You are now ready to complete [WIZ002 - Customizing Browse List Variables](#)

WIZ002 - Customizing Browse List Variables

Objective:

- To select a group of processes for the Wizard to work with.
- To configure the presentation of a specific browse list using the Wizard.

To achieve this objective, you will go through these sections:

- [Step 1. Create Test Processes and Functions](#)
- [Step 2. Execute the Wizard](#)
- [Step 3. Modify Browse List Colors](#)
- [Step 4. Modify Browse List Presentation Attributes](#)
- [Step 5. Copying Browse List Variables](#)
- [Summary](#)

Before You Begin

Review all topics in the section:

- [Customizing Browse Lists](#)

In order to complete this tutorial you must have completed tutorial:

- [WIZ001 - Executing the Wizard](#)

Step 1. Create Test Processes and Functions

In this step, you will create two test processes and test functions. The test functions will use the FRWEBENQ01 template to create *WEBEVENT functions which can be used to display information in the Section (SECTAB) and Personnel Skills (PSLSKL) files.

1. Using LANSAs, sign on to the partition nominated for the tutorials (usually DEM).
2. Create a new LANSAs process named iiPROC01 Test Process, where ii are your initials. (If the process already exists, select a different set of characters for ii.)
3. Use the Miscellaneous option for your iiPROC01 process and change the Enabled for the Web flag to Y.
4. Working with your iiPROC01 process, create a new function named iiFN001 Display Sections. Be sure to specify that the function is generated from an Application Template.
5. Select the FRWEBENQ01 template and answer as shown below:

TEMPLATE QUESTION	ANSWER
Enter the name of the file to be used	SECTAB
Select keys to use for search	Select DEPARTMENT
Fields to appear in browse list	Select DEPARTMENT, SECTION, SECTDESC
How many entries per page	5
Fields to appear in detail	Select all fields
What is the unique 2 character prefix	ii (where ii are your initials)

6. Repeat Steps 2 and 3 to create a second process named **iiPROC02 Test Process2**.

7. Working with your iiPROC02 process, create a new function named iiFN002 Display Personnel Skills. Be sure to specify that the function is generated from an Application Template.
8. Select the FRWEBENQ01 template and answer as shown below:

TEMPLATE QUESTION	ANSWER
Enter the name of the file to be used	PSLSKL
Select keys to use for search	Select EMPNO
Fields to appear in browse list	Select EMPNO, SKILCODE
How many entries per page	5
Fields to appear in detail	Select all fields
What is the unique 2 character prefix	ii (where ii are your initials)

9. Review, but do not change, the RDML code in your new functions. You should see a browse list named iiLIST2 in both functions. This list is used in your REQUEST and DISPLAY statements. You will be customizing the display of this specific browse list.
10. Compile your iiFN001 and iiFN002 functions.
11. Once the compiles have completed successfully, open a browser window and execute your functions to review the default list presentation. To execute your functions execute the following URLs:

`http://<server domain name>/cgi-bin/lansaweb?procfun+iiPROC01+iiFN001+<ppp>`

and

`http://<server domain name>/cgi-bin/lansaweb?procfun+iiPROC02+iiFN002+<ppp>`

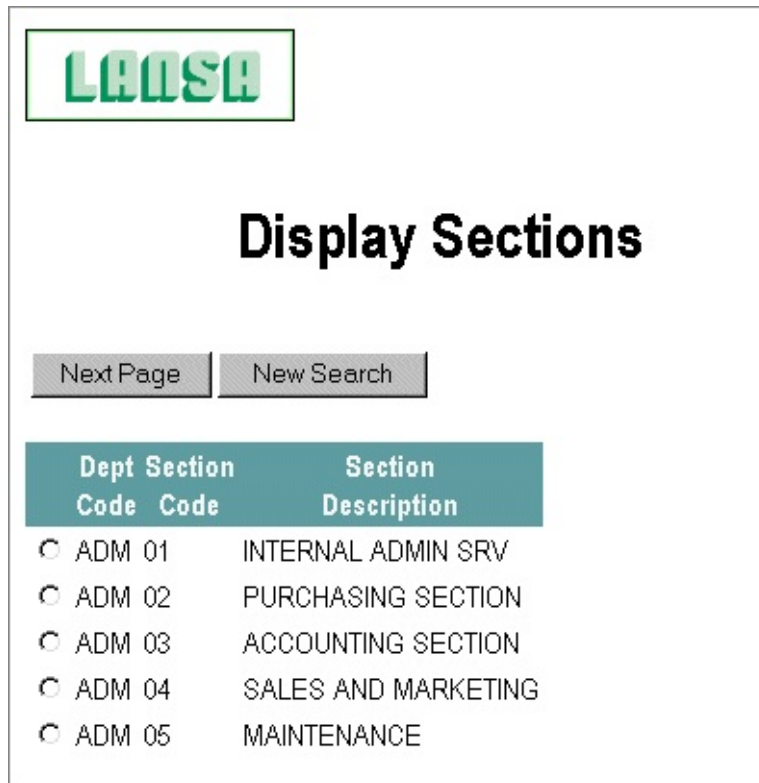
where:

<server domain name> is the domain name or IP address of your Web server

<ppp> is the LANSAPART partition where the Wizard is installed

ii are your initials

12. Press Next to see all Sections or Personnel Skills. Your function might appear like this:



LANSA

Display Sections

Next Page New Search

Dept Code	Section Code	Section Description
<input type="radio"/>	ADM 01	INTERNAL ADMIN SRV
<input type="radio"/>	ADM 02	PURCHASING SECTION
<input type="radio"/>	ADM 03	ACCOUNTING SECTION
<input type="radio"/>	ADM 04	SALES AND MARKETING
<input type="radio"/>	ADM 05	MAINTENANCE

If your browse list appears significantly different, it simply means that the system wide browse list defaults have already been configured. The system wide settings could have been defined from any LANSA partition.

Step 2. Execute the Wizard

In this step, you will use the Web Functions Wizard to select a subset of processes to be used with the Wizard.

1. Open a second browser window and execute the Wizard as described in WIZ001, [Step 2. Execute the Web Functions Wizard](#).
2. When the Home Page appears select the Process drop down to review the list of processes. You should see a list of all Web-enabled processes in the partition.
3. Select the Define Process Group option. The Define Process Group Window appears.
4. Enter ii, your initials from Step 1, then press the Apply Selection button. This returns you to the Wizard Home page.
5. Review the processes available for selection in the Process drop down. You will see a list of processes which start with ii and "DEFAULT" as the last entry in the drop down.

Note that the text associated with the Process drop down has changed from All Processes to Subset of Processes.

Step 3. Modify Browse List Colors

In this step, you will use the Web Functions Wizard to customize the colors used with the browse list named iiLIST2 as used in your iiFN001 and iiFN002 functions. In this step, you will manipulate only the color variables. You will be able to see how the changes made to browse lists are partition wide (and even system wide).

1. From the Wizard Home Page, select the Browse Lists option from the Characteristics to Modify drop down and then press Go. (You cannot use the Enter key to submit your request.)
2. When the Select Browse List screen appears, select the modify option for list iiLIST2 (where ii are your initials).

Note that there is no current process selected in your left menu. The browse lists available for review are extracted from all the processes available in the process drop down.

3. You can now customize the presentation of your browse list.

Note that all input fields should be blank. If values appear, then you have selected a list name that has been used by another developer. Close the browser window. Select a new prefix ii and repeat Step 1.

Use the Color Picker and set the colors for your browse list. You may also enter the HTML color names such as red, blue, green, etc. Be sure to use the Color button to check your color selection.

Leave the Background color for the column headings as a blank entry so that the default is still used.

4. Press the Preview button to review your selection. If you are satisfied with the result, press the Apply button to update your color selection. You will see a completion message to confirm you changes have been applied.
5. Using your first browser window, execute your functions iiFN001 and iiFN002 again to test your changes. Your browse lists might appear like this:

LANSA

Display Sections

Next Page

New Search

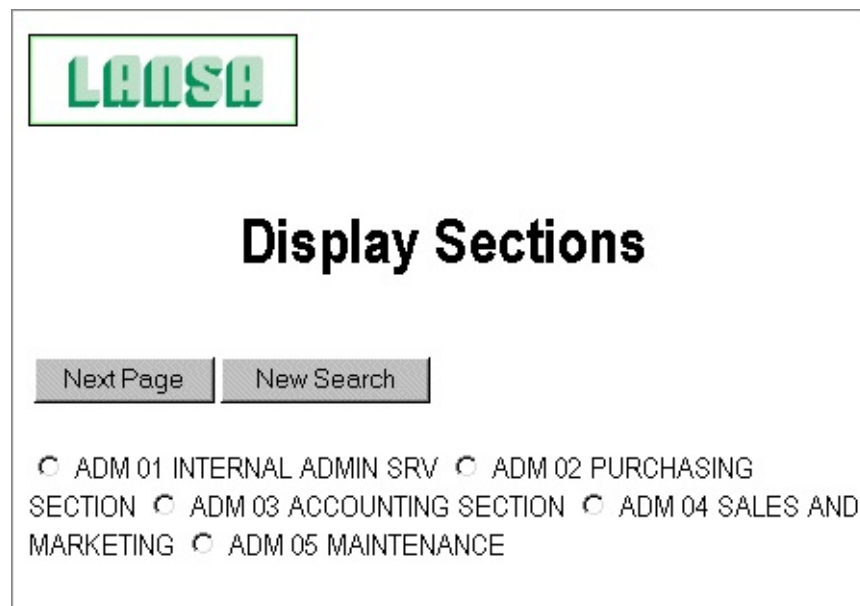
Dept Code	Section Code	Section Description
<input type="radio"/>	ADM 01	INTERNAL ADMIN SRV
<input type="radio"/>	ADM 02	PURCHASING SECTION
<input type="radio"/>	ADM 03	ACCOUNTING SECTION
<input type="radio"/>	ADM 04	SALES AND MARKETING
<input type="radio"/>	ADM 05	MAINTENANCE

Notice that you do not need to recompile your function and the list appearance has been modified in both functions.

Step 4. Modify Browse List Presentation Attributes

In this step, you will continue to customize the presentation of the browse list, using the Wizard, by altering the presentation attributes.

1. Execute the Web Functions Wizard again to set the browse list variables for **iiLIST2** (where ii are your initials).
2. Change the Presentation Attributes so that Column Headings and Table Tags are not used, i.e. check the boxes.
3. Press Apply to update the variables.
4. Using your first browser window, execute your test function to review the changes. You should see the data displayed without column headings. Also, your color settings are not applied as the data is not in an HTML table. Your list will appear similar to this:

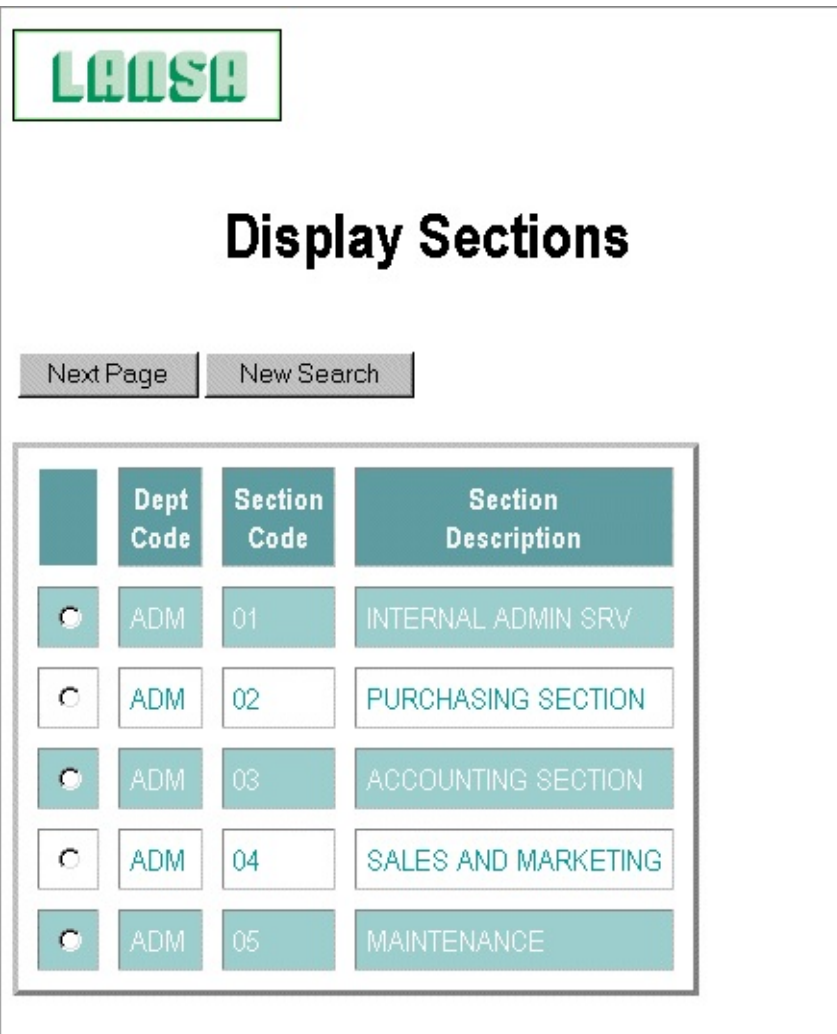


5. Using your Wizard browser Window, execute the Web Functions Wizard again to set the browse list variables for **iiLIST2**.
6. Reset the Presentation Attributes so that Column Headings and Table Tags are used, i.e. uncheck the boxes.
7. For the Selection image for the browse list, enter wiztick.gif.
8. If you are using an Internet Explorer browser, try using the Image button to preview the image.
9. Set the HTML Attributes used in conjunction with the <table> tag as follows

border="3 " cellpadding="5" cellspacing="10". Be sure to enter this text using upper case characters.

10. Press Apply to update the variables.

11. Execute your test function to review the changes. Remember, you have created a *WEBEVENT function so the browse list image is not used. Your list appears something like this:



The screenshot displays a web application interface. At the top left is a logo for 'LANSA' in green text. Below the logo is the title 'Display Sections' in a large, bold, black font. Underneath the title are two buttons: 'Next Page' and 'New Search'. The main content is a table with four columns: a selection column, 'Dept Code', 'Section Code', and 'Section Description'. The table contains five rows of data, each with a radio button in the selection column.

	Dept Code	Section Code	Section Description
<input checked="" type="radio"/>	ADM	01	INTERNAL ADMIN SRV
<input type="radio"/>	ADM	02	PURCHASING SECTION
<input checked="" type="radio"/>	ADM	03	ACCOUNTING SECTION
<input type="radio"/>	ADM	04	SALES AND MARKETING
<input checked="" type="radio"/>	ADM	05	MAINTENANCE

Step 5. Copying Browse List Variables

In this step, you will edit the function **iiFN002** created in Step 1 so that it uses a new browse list. Using the browse list copy feature, you can copy the presentation from your **iiLIST2** browse list.

1. Edit the RDML in the function **iiFN002** created in Step 1.
Use find and change in the HTML Editor to change all occurrences of **iiLIST2** to **iiLIST9**.
2. Compile and then execute your modified function. Notice that this function no longer uses your defined browse list settings because you have changed the browse list name.
3. Execute the Web Functions Wizard. When the Home Page screen appears, select the Browse Lists option from the Characteristics to Modify drop down and then press Go.
4. When the Select Browse List screen appears, check Copy from list iiLIST2 and Copy To iiLIST9 (where ii are your initials), then press the Copy button. You will see a completion message to confirm your changes have been applied.
5. Execute your function **iiFN002**. Notice that this function now appears identical to the browse list **iiLIST2** used in **iiFN001**.

Remember, this is a one time copy. If you change the presentation of **iiLIST2**, there will be no change to **iiLIST9**. You must perform the copy from **iiLIST2** to **iiLIST9** again.

Summary

Important Observations

- The Wizard Home Page initially includes all Web-enabled processes in the Process drop down. You can use Define Process Group to use a generic search by process name to reduce the number entries in this drop down list.
- The Browse Lists available to modify are based on the list names used in the RDML function. Modifications to browse list variables will impact all lists with the specified name in the LANSAs system.
- The changes you defined in this tutorial will impact all lists named iiLIST2 or iiLIST9 in all partitions in your system.
- You can specify color names manually by entering their number (e.g. #FF0000) or their name (e.g. red). Alternatively the Color Picker can be used to select a color from the Web-safe palette.
- You do not have to enter values for all browse list options. If you do not enter a value, the default variables will be used.
- Your functions do not have to be recompiled to reflect changes to the browse list variables. The changes are immediate.
- Modifications to the browse list DEFAULT will update the default system variables used for browse lists. The browse list name DEFAULT should not be used in your RDML code.

Tips & Techniques

- Open more than one browser window when using the Wizard. Use one browser window to execute the Wizard and use another browser window to test your functions.
- You may wish to alter the Presentation Attributes and HTML Attributes when your browse list uses components. For example, your tables may not display column heading and table borders. For more details, refer to [Browse List Graphic Variables](#).

What I Should Know

- How to use the Web Functions Wizard to customize the presentation of a specific browse list.
- How to modify the list of processes available in the Process drop down on the Wizard Home Page.

- Browse list variables are set based on the name of the browse list.
- Browse list settings are system wide.
- Browse list setting can be copied from one browse list to another browse list.
- If specific browse list values are not defined the system default values are applied.
- Browse list DEFAULT will modify the system default values for browse lists.

Other Tutorials

You are now ready to complete [WIZ003 - Customizing a Presentation Layout](#).

WIZ003 - Customizing a Presentation Layout

Objective:

- To select a layout presentation for a specific process.
- To define a color scheme and choose the layout features to be included in the layout presentation for a specific process.

To achieve this objective, you will go through these steps:

- [Step 1. Test Process/Function and Banner Component](#)
- [Step 2. Create a Process Layout](#)
- [Step 3. Test Your New Layout](#)
- [Step 4. Change the Color Scheme and Layout Features](#)
- [Summary](#)

Before You Begin

You may wish to review these topics:

- [Customizing a Presentation Layout](#)
- [Select Layout](#)
- [Color Scheme](#)
- [Layout Features](#)

In order to complete this tutorial, you must have completed:

- WIZ002, [Step 1. Create Test Processes and Functions.](#)

Step 1. Test Process/Function and Banner Component

You will need a test process and function to complete this tutorial. You must complete Wizard Tutorial 2, [Step 1. Create Test Processes and Functions](#), to create the iiPROC01/iiFN001 process/function. (The second Process iiPROC02 and function iiFN002 is not required for this tutorial).

If your partition does not have a standard banner (STDBANNER) Web component defined, you will need to use the Web Function Editor to create one. This banner is partition specific and may be used in the layouts created by the Wizard.

1. Use the Web Function Editor to create a new banner component named STDBANNER.

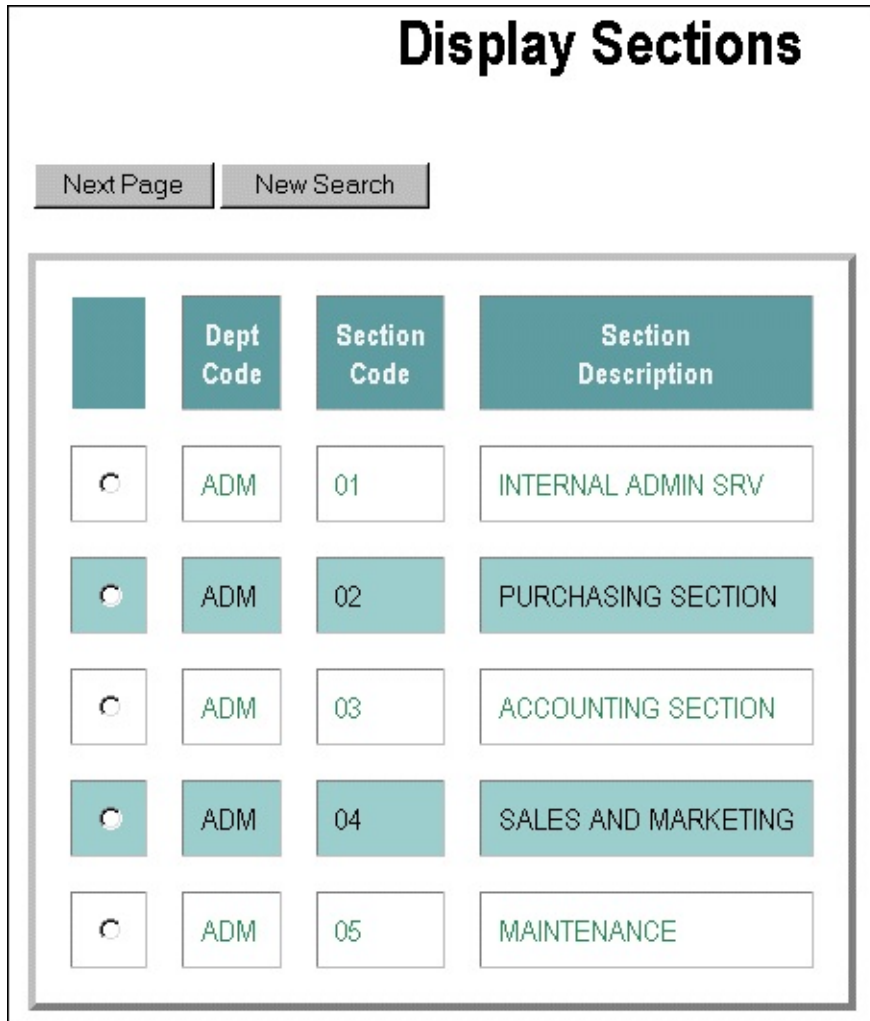
You can use the `banner_lansa.gif` and `banner_lansace.gif` image with links to the <http://www.lansa.com> Web address.

Step 2. Create a Process Layout

1. Execute the Web Functions Wizard. When the Wizard Home Page appears, set the Characteristic to Modify as Layout and select process iiPROC01, where ii are your initials. Press Go.
2. When presented with the Select Layout screen, select the first schema and then press the Build Layout button.
3. When the message "New Layout Build Complete" has been issued you can test your function. Leave your browser at this panel. In Step 4, you will use this panel to modify the process specific variables.

Step 3. Test Your New Layout

1. Using a second browser, execute your test function iiFN001. Your function should appear something like this:



	Dept Code	Section Code	Section Description
<input type="radio"/>	ADM	01	INTERNAL ADMIN SRV
<input checked="" type="radio"/>	ADM	02	PURCHASING SECTION
<input type="radio"/>	ADM	03	ACCOUNTING SECTION
<input checked="" type="radio"/>	ADM	04	SALES AND MARKETING
<input type="radio"/>	ADM	05	MAINTENANCE

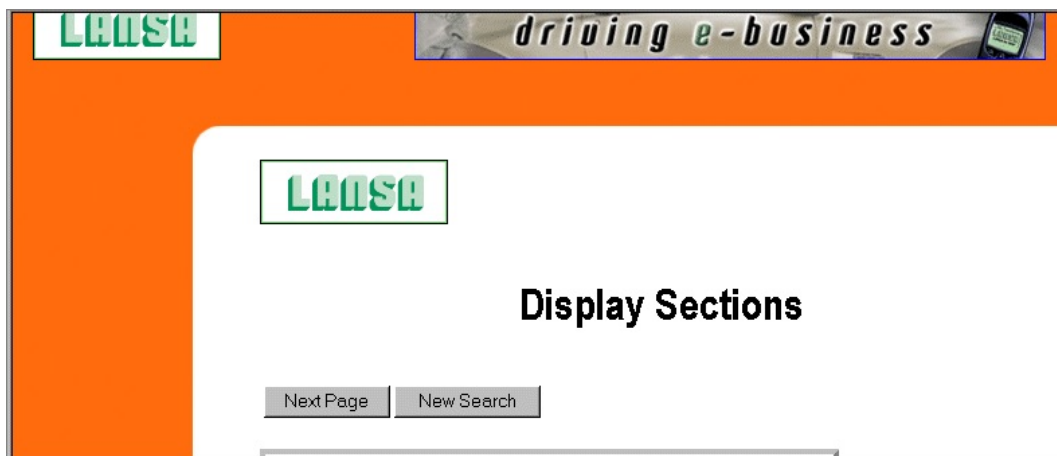
2. Notice that the default color scheme is used. No menu components, banner or company logo will appear as you have not indicated to include these features by updating the Layout Features. In [WIZ004 - Defining a Standard Menu Component](#) and [WIZ005 - Defining a DHTML Menu Component](#), you will create process specific menu components.
3. Using your first browser window, which should be displaying the Select Layout options, (as shown in the picture in [Select Layout](#)). Select the tab Color Scheme.
4. Change the process Background color to red.

5. Change the process Background Image to wizard2/daisy.gif. (You can use the File Search to assist locate the image). Press the Apply button.
6. Using your second browser, execute your test function iiFN001 again. Notice how the background color is displayed even though a background image is specified. You are able to see the background color because part of the background image is transparent.



Step 4. Change the Color Scheme and Layout Features

1. Return to your first browser window, which should be displaying the Color Scheme options (as shown in [Color Scheme](#)) for process iiPROC01.
2. Use the Color Picker to select white as your background color. Notice that the color picker will return a value of #FFFFFF when you select white, this is the hex value equivalent of white. White and #FFFFFF are both valid entries for the color white.
3. Clear the background image.
4. Set the Left/Horizontal Menu background color to Orange using the Wizard Supported Colors drop down.
5. Change the Right Menu Component background color to #EDB997. Press the Apply button to update your color scheme.
6. When the message "Color scheme successfully update" is presented, select the next tab to review your Layout Features.
7. Check to include a standard Left Menu, DHTML Horizontal Menu, Right Menu, Banner and Company Logo and then press the Apply button.
8. The Select Layout screen should be displayed with the message "Rebuild layout to apply changes to Layout Inclusions and Exclusions". Press the Build Layout button.
9. Return to your first browser window to execute your function and test your changes.



Summary

Important Observations

- The Web Functions Wizard is very quick and easy to use. You can try many different layouts to find a design to meet the needs of your Web function application.
- When Layout Schemas 1, 3 or 5 are selected, only Wizard supported left/horizontal menu background colors will include the curved corners. Any manually selected color will be implemented with squared corners in the layout presentation.
- Changes to the color schemes are automatically applied to the layout (i.e. a re-build of the Layout is not required).
- Any changes to the layout features require that the layout be re-built from the Select Layout tab.
- DHTML menus can only be defined for the Left and Horizontal Menu.
- Only one DHTML menu can be included on a layout.
- A background image and background color can both be used if the background image includes transparent areas in the image.
- Previous selections used to build your layout for a specific process are pre-selected when you return to the Wizard.
- System wide settings will be used if no process specific values have been created. (Remember: System Wide default layouts and menu components are partition specific.)

Tips & Techniques

- Be very careful with the definition of default layouts (process DEFAULT) for presentations and components. The examples in this tutorial are process specific so the layout changes only impact functions in a single process. If you define a system wide layout or components, the changes impact all of your functions in the partition (unless they have a specific layout).
- If you change your left/horizontal menu background color from a color of your own selection to a Wizard supported color and are you using layout schema 1, 3 or 5, you must rebuild the layout to include the curved corners.
- If you change your left/horizontal menu background color from a Wizard supported color to a color of your own selection and are you using layout schema 1, 3 or 5, you must rebuild the layout to exclude the curved corners

image references.

What I Should Know

- How to customize the presentation layout.
- How to modify the color scheme and layout features associated with a layout presentation.
- How to include menu components into the layout.

Other Tutorials

You are now ready to complete [WIZ004 - Defining a Standard Menu Component](#) and [WIZ005 - Defining a DHTML Menu Component](#).

WIZ004 - Defining a Standard Menu Component

Objective:

- To customize the left and right menu components.
- To demonstrate the different types of menu items which can be added to a standard menu component.

To achieve this objective, you will go through these steps:

- [Step 1. Create a Test Process with Menu Components](#)
- [Step 2. Change Menu Features](#)
- [Step 3. Build Left Menu](#)
- [Step 4. Include STDTABFLR](#)
- [Step 5. Build Right Menu](#)
- [Summary](#)

Before You Begin

You may wish to review this topic:

- [Customizing a Standard Menu](#)

In order to complete this tutorial, you must have completed:

- [WIZ003 - Customizing a Presentation Layout](#)

Step 1. Create a Test Process with Menu Components

You will need two test processes and functions to complete this tutorial. It is recommended that you complete

- Tutorial 2, [Step 1. Create Test Processes and Functions](#), to create the processes/functions iiPROC01/iiFN001 and iiPROC02/iiFN002, and
- Tutorial 3, [Step 4. Change the Color Scheme and Layout Features](#), to include the correct menu components in the layout for process iiPROC01. The process layout must have been defined to include standard left and right menu components.

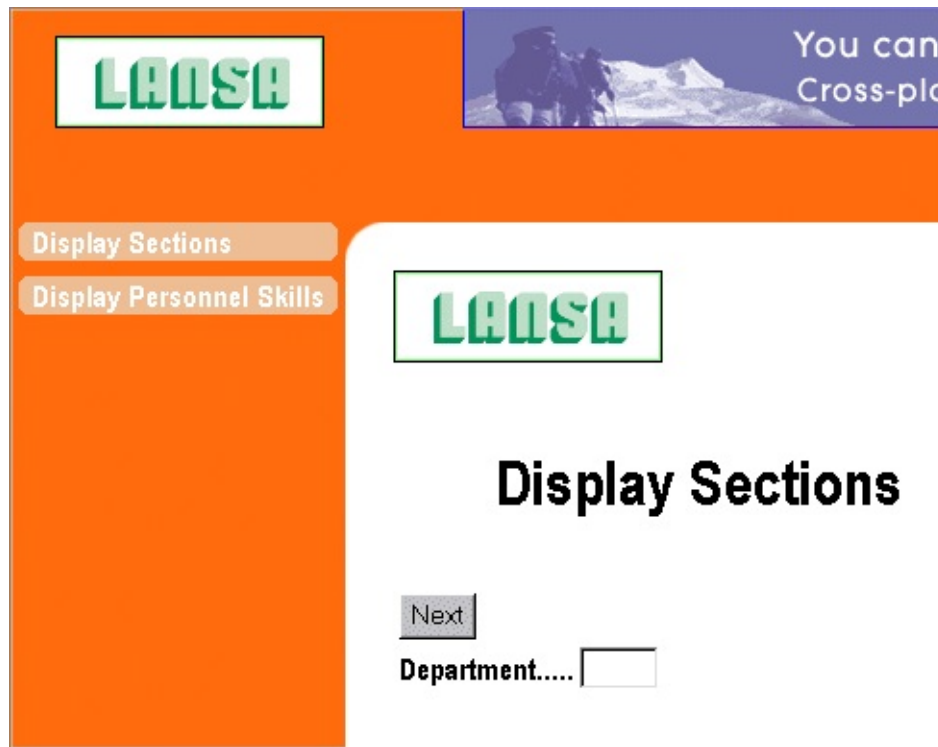
Step 2. Change Menu Features

In this step, you will change the default button style, used when creating menu items, and you will change the menu separator image used for the menu.

1. Execute the Wizard. When the Wizard Home page appears set the Characteristic to Modify as Menus and select process iiPROC01, where ii are your initials. Press Go.
2. When the Menu screen appears, notice the Left Menu tab is selected by default. Select the Menu Features tab.
3. Set the Menu separator image to hs_blk.gif . Check the image exists using the Preview button.
4. Set the Default Button Style (the first drop down) to Button Curved and the Button Color (the second drop down) to Light Orange. Press the Apply button. These changes will be tested in the next step.

Step 3. Build Left Menu

In this step, you will create the left menu component. The menu will include two dynamic buttons to link to another LANSA process and function.



1. Select the Left Menu Tab. Check your current process is set as iiPROC01, where ii are your initials.
2. Press the Add Menu Item button. Notice the left menu is selected by default.
3. When the Define Left Menu detail panel appears enter this information:

Sequence Number	10
Menu Type	<i>Dynamic Button</i>
Description	Display Sections
Button Style	Button Curved / Light Orange
Menu State	<i>Always shown in the Menu Component</i>
Link Type	<i>LANSA Function</i>
Process	iiPROC01

Function **iiFN001**
Reuse Job check box (i.e., yes)
Start in New Window uncheck box (i.e., no)

Notice your button style and color are pre-selected based on the values you set up using Menu Features in Step 2.

4. Press *Save Menu Item*.
5. When the Define Left Menu panel appears, press the *Add Menu Item* button again. Enter this information:

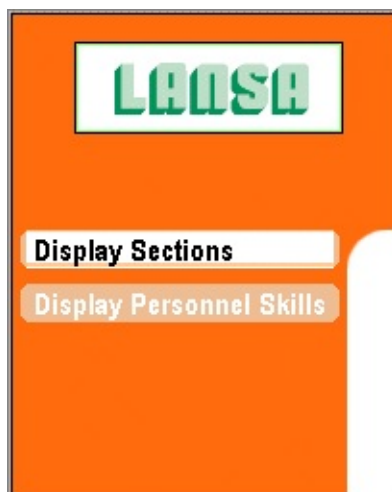
Sequence Number **20**
Menu Type *Dynamic Button*
Description **Display Personnel Skills**
Button Style **Button Curved / Light Orange**
Menu State *Always shown in the Menu Component*
Link Type *LANSA Function*
Process **iiPROC02**
Function **iiFN002**
Reuse Job check box (i.e. yes)
Start in New Window Uncheck box (i.e. no)

6. Press *Save Menu Item*.
7. When the Define Left Menu panel appears, press the *Build Menu* button. (Do not check the box to Include Menu Separator Between Menu Items.)
8. Using a second browser, execute your test function iiFN001. You should see two buttons in your left menu. Notice that when you select Display Personnel Skills, function iiFN002 has not been customized with the Wizard.
9. In the final application, you will want to have the same presentation layout for all functions so that you can navigate between all functions using the menus. In order to accomplish this goal, complete [WIZ007 - Adopting a Presentation Layout](#).

Step 4. Include STDATABFLR

Include the STDATABFLR field in your RDML screen definitions to support an alternate button color to indicate when a button is selected. If STDATABFLR does not already exist in your data dictionary, create it and define it as A(20).

1. Edit the RDML for your functions iiFN001 and iiFN002 to include the #STDATABFLR field as a hidden field (attribute *HIDDEN) in each of the screens. You may need to modify REQUEST, DISPLAY and GROUP_BY statements.
2. Recompile your functions.
3. Execute your iiFN001 test function. When you use the Left Menu Item to execute the Display Sections function, the left menu will appear:



Notice that the text color on the Display Section changes from white to black when the button is selected. This is determined by the process style sheet. For more details refer to [WIZ006 - Defining a Style Sheet](#).

Step 5. Build Right Menu

In this step, you will build the right menu. The right menu will contain items with information about the department selected. This example will use static text but could be created using Web components.

1. Using your first browser window, which should be displaying the Define Left Menu main screen. Select the tab Right. Check that your current process is your iiPROC01 process.
2. Press the Add Menu Item button and enter this information:

Sequence Number **10**

Menu Type *Description*

Description **Department Information**

Menu State *Always shown in the Menu Component*

Link Type *No link*

3. Press *Save Menu Item*.
4. When Define Right Menu panel appears, press the *Add Menu Item* button again and enter this information:

Sequence Number **20**

Menu Type *Description*

Description **ADM
 Administration**

Menu State *Only show if the field condition is TRUE*

Field to check: **DEPARTMENT**

Value to check: **ADM**

Link Type *No link*

5. Press *Save Menu Item*.
6. When Define Right Menu panel appears, press the *Add Menu Item* button again and enter this information:

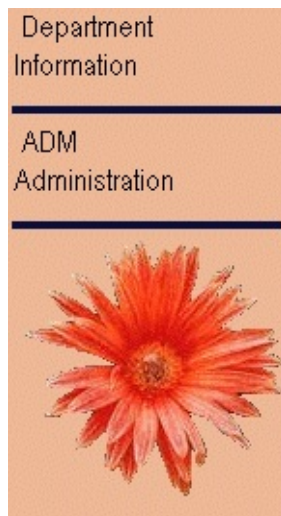
Sequence Number **30**

Menu Type	<i>Single Image</i>
Static Image	wizard2/daisy.gif
Menu State	<i>Always shown in the Menu Component</i>
Link Type	<i>No link</i>

7. Press *Save Menu Item*.
8. When the Menu Builder appears, check the box for *Include Menu Separator Between Menu Items*, and then press the *Build Menu* button.
9. Using a second browser, execute your test function iiFN001. Notice that the Right Menu only includes the first and third items, i.e. the descriptive text Department Information and image.



10. Now display the sections for the ADM department. When the section list is displayed, the information about the ADM department will be displayed.



Summary

Important Observations

- The presentation layout defined for iiPROC01 is process specific. The iiPROC02 functions are not impacted. In [WIZ007 - Adopting Presentation Layout](#) you will make these layouts the same.
- Menu items can link to a component or URL or call a LANSAs process or function.
- A menu item can also be displayed as simple text, image(s), a dynamic button or a Web page/component.
- The Menu State is a powerful feature which allows you to define dynamic menus.
- Select the Include Menu Separator Between Menu Items just before your build the menu. This check box is reset each time you add a new item to the menu.
- The descriptive text used for the menu item can include HTML tags.
- Include the hidden field STDTABFLR in your RDML screen definitions to automatically enable on/off presentation for your dynamic buttons.

Tips & Techniques

- Menu items can link to a component or URL or call a LANSAs process or function.
- A menu item can be displayed as simple text, image(s), a dynamic button or a Web page/component. Use a Link type of No link to add simple text to your menus.
- Use the Image button to check that the image you have specified exists.
- To search for an image, use the File Search facility on the left hand menu of the Wizard. For more details, refer to [File Search](#).
- You can create very powerful and dynamic menus by using Web components in the menus.
- Think carefully when specifying the Reuse Job and Start in New Window options. You must understand the type of function executing and the navigation required to and from the function.
- If you are planning to consistently use dynamic buttons in your functions, modify the application template to include the STDTABFLR field.

- You can use the Web Function Editor and include the STDTABFLR field in your DEFAULT_HIDDEN or the <process>_HIDDEN HTML page so that it is included in your functions. You will not have to modify the RDML code to include this field.

What I Should Know

- How to add items to a standard left or right menu component.
- How to use link types no link and LANSA function.
- How to use the menu state to control when an item will appear.
- How to set up the dynamic buttons default values.
- How to set up the menu separator image.
- How to use the STDTABFLR field with dynamic buttons to create on/off buttons.

Other Tutorials

You are now ready to complete [WIZ005 - Defining a DHTML Menu Component](#).

WIZ005 - Defining a DHTML Menu Component

Objective:

- To customize the horizontal menu component.
- To demonstrate the construction of a DHTML menu component.

To achieve this objective, you will go through these steps:

- [Step 1. Create a Test Process with Menu Components](#)
- [Step 2. Horizontal Menu Components](#)
- [Step 3. Change DHTML Menu Features](#)
- [Summary](#)

Before You Begin

Note: The DHTML menu script used by the Wizard (component WIZMNUSCR) is not supported in Version 6 of the Netscape browser. If you expect your Web function applications to be used with Netscape V6, it is recommended that you use Standard Menus in your interface design.

You may wish to review this topic:

- [Customizing a DHTML Menu](#)

In order to complete this tutorial, you must have completed:

- [WIZ003 - Customizing a Presentation Layout.](#)

Step 1. Create a Test Process with Menu Components

You will need a test process and function to complete this tutorial. It is recommended that you complete:

- WIZ002 - Customizing Browse List Variables, [Step 1. Create Test Processes and Functions](#), to create the iiPROC01/iiFN001 process/function (the second Process iiPROC02 and function iiFN002 is not required for this tutorial), and
- WIZ003 - Customizing a Presentation Layout, [Step 4. Change the Color Scheme and Layout Features](#), to include the correct menu components in the layout for process iiPROC01. The process layout must have been defined to include a DHTML Horizontal menu component.

Step 2. Horizontal Menu Components

In this step, you will create a horizontal menu component which will act as a navigation bar for your functions. Your horizontal menu will contain three main options: Home, Links and Help.

1. Execute the Wizard. When the Home Page appears, set the *Characteristic* to Modify as DHTML Menu and select process iiPROC01, where ii are your initials. Press Go.
2. Select the tab Horizontal. Notice the left menu is selected by default and you have the options to select Left or Horizontal Menu to be customized.
3. Press the Add Menu Item button.
4. When the Define DHTML Horizontal detail panel appears, enter this information:

Sequence Number	1
Menu Type Description	Functions Wizard
Link Type	LANSA Function
Process	WIZARD2
Function	WIZ02
Start in new Window	check box (i.e. yes)

5. Press *Save Menu Item*.
6. When the Define DHTML Horizontal detail panel appears, press the *Add Menu Item* button again and enter this information:

Sequence Number	2
Menu Type Description	Useful Links
Link Type	No Link

7. Press *Save Menu Item*.
8. When the Define DHTML Horizontal detail panel appears, press the *Add Menu Item* button again and enter this information:

Sequence Number	3
Menu Type Description	Help
Link Type	URL
URL	<p><RDML MERGE= " *LW3WIZHELP " >/wiz02.htm</p> <p>Note: The value <RDML MERGE= " *LW3WIZHELP " > must be uppercase to evaluate correctly.</p>
Start in new Window	uncheck box (i.e. no)

9. Press *Save Menu Item*.
10. When the Define DHTML Horizontal detail panel appears, press the Sub-Menu for the new Useful Links menu item. Notice a new menu is presented with a different parent menu identifier. Press the *Add Menu Item* button and enter this information:

Sequence Number	1
Menu Type Description	LANSA Home Page
Link Type	URL
URL	http://www.lansa.com
Start in new Window	uncheck box (i.e. no)

11. Press *Save Menu Item*.

12. When the Define DHTML Horizontal panel appears, *Add Menu Item* button again and enter this information:

Sequence Number	2
Menu Type Description	Qantas
Link Type	URL
URL	http://www.qantas.com.au
Start in new Window	uncheck box (i.e. no)

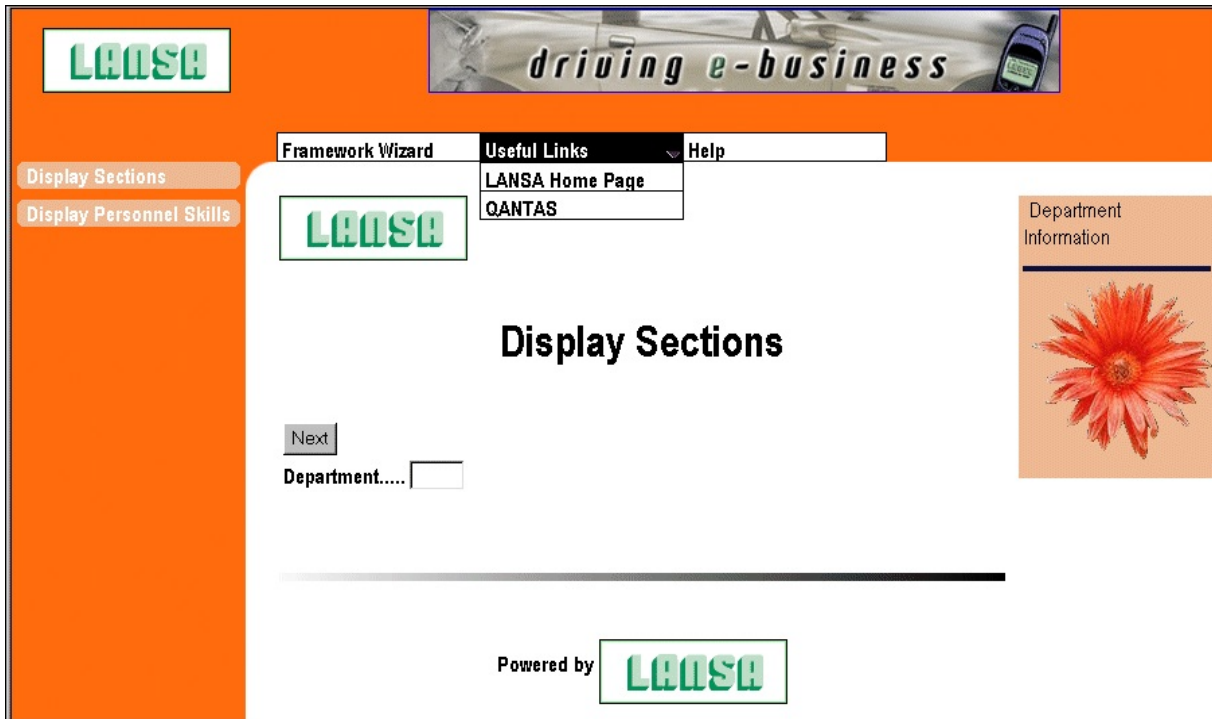
13. Press *Save Menu Item*.

14. When the Define DHTML Horizontal panel appears, press the Parent Menu button. Notice this returns you to your original menu with parent menu identifier 1.

15. Press the Build Menu button.

Step 3. Change DHTML Menu Features

1. Using a second browser, execute your test function iiFN001. Click on Useful Links to view the sub-menu options. Your function might appear something like this:



2. Notice that the default DHTML Menu Features are used.
3. Using your first browser window, which should be displaying the Define Horizontal DHTML Menu main screen, select the Menu Features tab.
4. Change the Font Color, Font Sub-menu Color, Border Color and Border Sub-menu Color to #823706.
5. Change the Offset from Top to -15 and Offset from Left to 50. Press the button Apply. Rebuild your menu.
6. Using your second browser, execute your test function iiFN001 again. Notice how the font color, borders on the DHTML menu and position of the menu have changed.

Summary

Important Observations

- You can create very powerful menus using the DHTML menus.
- A DHTML Menu Item can use the same Link Types as a Standard Menu Item.
- A DHTML Menu Item does not have a Menu State (which is available with Standard Menu Items).
- DHTML menus can include a sub-menu for a specific menu item.
- The URL link can include RDML tags.
- The appearance of your DHTML menu can be modified by changing the Menu Features.
- You need to re-build your DHTML menu after changing the Menu Features.
- Components used in DHTML menus must be defined as **script** type components in the Web Function Editor.

Tips & Techniques

- Menu items can link to a component or URL or call a LANSAs process or function.
- Use a Link type of No link to add simple text to your menus or indicate a sub-menu name.
- Think carefully when specifying the Reuse Job and Start in New Window options. You must understand the type of function executing and the navigation required to and from the function.

What I Should Know

- How to customize a DHTML menu.
- How to modify the appearance of a DHTML menu using the Menu Features.

Other Tutorials

You are now ready to complete [WIZ006 - Defining a Style Sheet](#).

WIZ006 - Defining a Style Sheet

Objective:

- To review the existing style sheet definition.
- To create style sheet tags.

To achieve this objective, you will go through these sections:

- [Step 1. Review the Style Sheet Definition](#)
- [Step 2. Modify the Style Sheet Definition](#)
- [Step 3. Delete a Style Sheet Tag](#)
- [Summary](#)

Before You Begin

You may wish to review this topic:

- [Style Sheets](#)

In order to complete this tutorial, you must have completed:

- [WIZ004 - Defining a Standard Menu Component.](#)

Step 1. Review the Style Sheet Definition

In this step, you will modify the style sheet tag which controls the text color used on the Left menu buttons.

1. Execute the Wizard. When the Wizard Home Page appears, set the Characteristic to Modify as Style Sheet and select process iiPROC01, where ii are your initials. Press Go.
2. When presented with the Define Style Sheet screen, notice that several class tags are already defined. The existing Style Sheet was created when your layout was built for the first time in Tutorial 3 - Customizing a Presentation Layout, Step 2.
3. Select the Modify action for the class .leftmenu.
4. When the Define Style Sheet details screen appears, change the color value #FFFFFF to #823706. Press the Update button to continue.
5. When the Define Style Sheet screen appears, press the Build Style Sheet button.
6. When the message "Process Style Sheet successfully created" has been issued, you can test your function iiFN001 using a second browser window. Notice the text color on the left menu buttons has changed from white to dark orange.

Step 2. Modify the Style Sheet Definition

In this step, you will create style sheet tags to control the font used in the body of your HTML pages, the heading level 1 font color and the table cell font color.

1. Return to your first browser window, which should be displaying the Define Style Sheet main screen for process iiPROC01.
2. Press the Add a Tag, Class or ID button.
3. Select the body from the Tag drop down. Build the Declarations using the Declaration assistant below the free format area. Set Font 1 to Arial and Font 2 to Garamond, and then press Add to Tag button.
4. Press the Add Tag button.
5. When the Define Style Sheet main screen appears, press the Add a Tag, Class or ID button.
6. Select the h1 from the Tag drop down. Set the font color to #823706 in the Declaration assistant, and then press Add to Tag button.
7. Press the Add Tag button.
8. When the Define Style Sheet main screen appears, press the Add a Tag, Class or ID button.
9. Select the td from the Tag drop down. Use the color picker to set the font color to pink (select any pink from the palette), and then press Add to Tag button.
10. Press the Add Tag button.
11. When the Define Style Sheet screen appears, press the Build Style Sheet button.
12. When the message "Process Style Sheet successfully created" has been issued you can test your function iiFN001 using the second browser window. Notice the font of all text has changed to Arial (if you have Arial as a supported font), the function title text has changed to dark orange and various pieces of text are pink.

Step 3. Delete a Style Sheet Tag

1. Return to your first browser window, which should be displaying the Define Style Sheet main screen for process iiPROC01.
2. Select the Delete action for the tag td.
3. When the Define Style Sheet screen appears, press the Build Style Sheet button.
4. When the message "Process Style Sheet successfully created" has been issued, you can test your function iiFN001 using a second browser window. Notice that all the pink text has reverted back to the default value.

Summary

Important Observations

- Style sheets are a very quick and effective way to modify your Web function application.
- Changes to the style sheet are automatically applied to the layout (i.e. a build of the Layout or Menu is not required).
- To define a style sheet effectively, you will require some knowledge of the HTML tags used by your Web function application.
- Style sheets can include tags, classes, ids, sub-classes or any other construct supported by Cascading Sstyle Sheets.
- Common tags attributes can be selected and generated using the Declaration assistant.
- Tag attributes are not validated.
- It is your responsibility to remove any duplication and correct errors in the free-format entry area.
- It is important to test your style sheet using all the browsers you intend to support with your Web function application.
- Browse lists are not the only HTML table constructs in LANSA for the Web components.

Tips & Techniques

- Browse list characteristics can be modified using the Browse list characteristic in the Wizard or using style sheets to modify various table tags.
- When using style sheets be careful that you understand how and where the specified tags are used in your application.
- Always test your style sheet to ensure the declarations you have defined, work as expected in your Web function application.
- Ensure you test your style sheet in all browsers you support.

What I Should Know

- How to define, modify and test a style sheet.

Other Tutorials

You are now ready to complete [WIZ007 - Adopting a Presentation Layout](#).

WIZ007 - Adopting a Presentation Layout

Objective:

- To adopt the presentation layout from a specific process.
- To show how changes can be made to the adopted from process and are inherited to the adopting process.

To achieve this objective, you will go through these steps:

- [Step 1. Adopt the Layout](#)
- [Step 2. Modify the Components in the Adopted From Process](#)
- [Step 3. Process Specific Changes When Using Adopted Layout](#)
- [Summary](#)

Before You Begin

You may wish to review these topics:

- [Adopt Layout](#)
- [Adopt Layout-Tips & Techniques](#)
- [Adopt Layout Technically Speaking](#)

In order to complete this tutorial, you must have completed:

- [WIZ004 - Defining a Standard Menu Component](#)
- [WIZ005 - Defining a DHTML Menu Component](#)

Step 1. Adopt the Layout

In this step, you will adopt the Layout Presentation from process iiPROC01 ([WIZ003 - Customizing a Presentation Layout](#)) to process iiPROC02.

1. Execute the Wizard. When the Wizard Home page appears, set the Characteristic to Modify as Layout and select process iiPROC02, where ii are your initials. Press Go.
2. When presented with the Select Layout screen, select the Adopt Layout tab.
3. From the Select Process to Adopt from drop down select your first process iiPROC01. Press the Adopt Process button.

As this process did not have a previous layout defined, there is no need to check any of the optional checkboxes. These checkboxes indicate you want to remove the associated process variable, so the color will be adopted from the selected process.

4. When the message "Process Layout successfully Adopted" has been issued, use a second browser window to execute your test function iiFN001. From the Left Menu, execute the *Display Personnel Skills* function. Notice how this page now appears exactly as iiFN001.

Step 2. Modify the Components in the Adopted From Process

In this step, you will modify the layout color scheme and layout features in the adopted from process (iiPROC01) to show how these changes are inherited to adopting process.

1. Return to your first browser window, execute the Wizard again to change the Color Scheme for iiPROC01.
2. Set the Left/Horizontal Menu background color as Light Gray using the Wizard Supported Colors drop down.
3. Change the Right Menu Component background color to #C0C0C0. Press the Apply button to update your color scheme.
4. When the message "Layout Color scheme successfully updated" is presented, select the next tab to review you Layout Features.
5. Uncheck the Banner and Company Logo check boxes, and then press the Apply button.
6. The Select Layout screen should be displayed with the message "Rebuild layout to apply changes to Layout Inclusions and Exclusions". Select a different layout schema, then press the Build Layout button.
7. Return to your second browser window to execute your function and test your changes. The Display Personnel Skills function in iiPROC02 will have the same color schemes and components as the parent process iiPROC01.

Step 3. Process Specific Changes When Using Adopted Layout

In this step, you will define a process specific standard menu component to override the adopted layout.

1. Return to your first browser window, execute the Wizard again. Define a Standard Menu Component for the Right Menu of your iiPROC02 process.
2. When the Define Right Menu panel appears, press the Add Menu Item button and enter this information:

Sequence Number 10

Menu Type	Description
Description	iiPROC02 Specific Menu Item
Menu State	Always shown in the Menu Component
Link Type	No link

3. Press *Save Menu Item*.
4. When the Define Right Menu appears, press the *Build Menu* button.
5. Press the Home option in the Wizard left menu. When the Wizard Home Page appears, select to modify the Layout for your iiPROC02 process.
6. Select the Tab to modify the Color Scheme. Change the Right Menu Component background color to limegreen. Press the Apply button.
7. Return to your second browser window to execute your test function iiFN001. Notice the right menu.
8. Execute the Display Personnel Skills function, iiFN002. Notice the right menu has a lime green background color. The right menu background color is not inherited because a process specific value exists.
9. Return to your first browser window, which should be displaying the Color Scheme options for process iiPROC02. Select the Adopt Layout tab.
10. From the Select Process to Adopt from dropdown list, select your first process iiPROC01. Check the Inherit Right Menu Background Color. Press the Adopt Process button.
11. Return to your second browser window to execute your test function iiFN001. Notice the right menu.

12. Execute the Display Personnel Skills function, iiFN002. Notice the right menu has a gray background color. The right menu background color is again inherited because a process specific value no longer exists.

Summary

Important Observations

- When a process is first adopted, the presentation layout, style sheet and menu components are adopted.
- You can create process specific overrides when you have adopted a layout. The adopting process definitions are used over the adopted layout definitions (with the exception of DHTML menus).
- Process specific color variables can be removed when adopting from another process.
- DHTML menus are always adopted. Any process specific DHTML menu's in the adopting process will be ignored if the process layout has been adopted from another process.

Tips & Techniques

- When you have created process specific settings, the adopted layout will no longer be used (except for DHTML menus).
- Remove process specific color variables, to revert back to the parent colors, by using the check boxes provided when adopting a process.
- Layout Features will always be inherited from the adopted process (e.g. if the adopted process includes a standard left menu, the adopting process will include a standard left menu).
- If you wish to reset the iiPROC02 process to its default, you can use the Wizard to delete the process layout, delete the menu components and delete the color scheme. Alternatively, you can use the Web Function Editor and delete the iiPROC02_LAYOUT, iiPROC02_LMENU, iiPROC02_RMENU, and iiPROC02_HMENU pages and the associated graphic variables.

What I Should Know

- How to adopt a process layout.
- How to create process specific overrides when using an adopted layout.
- How to modify the color scheme for adopted and adopting processes.

Appendix A. Dynamic Buttons

The Web Functions Wizard is shipped with a selection of images to support the construction of button and tabs in standard menus. A separate sub-directory of images is shipped to support the Wizard execution.

In this appendix you will find:

[Dynamic Button and Tab Images](#)

[Dynamic Button and Tab Colors](#)

Dynamic Button and Tab Images

Both On (selected) and Off (not selected) effects have been supplied for each dynamic button style and color. Each button is built from three images. A left image, a right image and a middle image that stretches over the description associated with the button.

Button/Tab Style Style Suffix Not Selected Image Selected Image

Button Square	btn1		
Button Curved	btn2		
Tab Square	Tab1		
Tab Curved	Tab2		

For example, a red curved dynamic button would use six images (three for not Selected and three for Selected) as follows:

Not Selected images

- btn2_l_red.gif
- btn2_c_red.gif
- btn2_r_red.gif

Selected images

- btn2_sell_red.gif
- btn2_selc_red.gif
- btn2_selr_red.gif

Dynamic Button and Tab Colors

The following table lists all the button and tab colors supported by the Wizard. The Color Suffix is used internally to build the corresponding image name. The name of the image to use is built by concatenating the button style, button color suffix and button segment (left / middle / right). For example, the left side of a square, selected, red, dynamic button would use the image btn1_sell_red.gif.

Color	Color Suffix	Hex Value
Dark Red	Dred	#82001A
Red	Red	#FF0033
Pink	Lred	#EC92A4
Dark Purple	Dprp	#4E1A4E
Purple	Prp	#993399
Light Purple	Lprp	#C4A0C4
Dark Orange	Dorg	#823706
Orange	Org	#FF6C0C
Light Orange	Lorg	#EDB997
Dark Green	Dgrn	#395138
Green	grn	#709E6D
Light Green	lgrn	#BACAB9
Dark Blue	dblu	#1A1A68
Blue	blu	#3333CC
Light Blue	lblu	#9999CC
Dark Gray	dgry	#343434
Mid Gray	gry2	#686868
Gray	gry1	#B2B2B2
Light Gray	lgry	#E5E5E5

