ORACLE

PEOPLESOFT

PeopleTools 8.54: Fluid User Interface Developer's Guide

November 2016

ORACLE

PeopleTools 8.54: Fluid User Interface Developer's Guide
CDSKU

License Restrictions Warranty/Consequential Damages Disclaimer

Warranty Disclaimer

Restricted Rights Notice

Hazardous Applications Notice

Trademark Notice

Third Party Content, Products, and Services Disclaimer

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Contents

Preface ..... **ix**
Understanding the PeopleSoft Online Help and PeopleBooks ix
PeopleSoft Hosted Documentation ix
Locally Installed Help ix
Downloadable PeopleBook PDF Files ix
Common Help Documentation x
Field and Control Definitions x
Typographical Conventions x
ISO Country and Currency Codes xi
Region and Industry Identifiers xi
Using and Managing the PeopleSoft Online Help xii
PeopleTools Related Links xii
Contact Us xii
Follow Us xii
Chapter 1: Getting Started with Fluid User Interface Development **15**
Fluid User Interface Overview 15
Implementing PeopleSoft Fluid User Interface 15
Chapter 2: Understanding PeopleSoft Fluid User Interface **17**
Understanding the Need for a Fluid User Interface 17
Understanding the Characteristics of Fluid Applications 18
Understanding How to Build Fluid Applications 21
Considerations for PeopleSoft Fluid Application Implementation 22
Chapter 3: Creating Pages for PeopleSoft Fluid Applications **25**
Selecting Page Types 25
Creating the Primary Fluid Page 27
Creating a New Primary Fluid Page 27
Cloning and Existing Page for Fluid Deployment 27
Choosing a Layout Page 28
Setting Properties for Fluid Pages 32
Setting Use Properties 33
Setting Fluid Properties 34
Defining the Layout of the Main Fluid Page 35
Chapter 4: Defining Components for Fluid Applications **41**
Adding Fluid Pages to a Fluid Component 41
Setting Component Properties for Fluid Components 42
Setting Fluid Component Properties 42
Setting Internet Properties for Fluid Components 44
Working with Search Pages 45
Understanding Search Pages for Fluid Applications 45
Working with Search Pages for Real Time Component Search 47
Working with Search Pages for Oracle SES-based Search 48
Comparing the Availability of Fluid Search Features to Classic Search Features 49
Chapter 5: Adding Page Controls **51**
Managing Common Control Concepts and Settings for Fluid Pages 51
Understanding the Makeup of a Control on a Fluid Page 51
Setting Common Page Control Fluid Properties 52
Working with Group Boxes 53

Using Group Boxes for Fluid Pages....54
Setting Group Box Properties....56
Working with Layout Group Boxes....59
Working with Container Group Boxes....59
Working with Accordion Group Boxes....59
Creating Custom Headers....60
Working with Related Field Group Boxes....60
Testing Group Box Layout....60
Working with Grouplets....61
Working with Other Page Controls....64
Working with Grids....64
Working with Edit Boxes and Long Edit Boxes....65
Working with Subpages....67
Working with HTML Areas....67
Working with Check Boxes....67
Working with Push Button/Hyperlink....68
Working with Static Images and Static Text....69
Working with Frames....69
Working with Horizontal Rules....69
Applying Styles....69
Understanding Fluid Styling....69
Identifying PeopleTools Style Sheets....70
Overriding Delivered Style Classes....70
Setting Min/Max Width on Content....72
Hiding Columns in a Flex Grid Using Style....73
Working with Flex Grid Column Alignment....74
Working with JavaScript....74
Delivered JavaScript....75
Custom JavaScript Guidelines....76
Considering Accessibility for Fluid Pages....76
Chapter 6: Working with PeopleCode for the Fluid User Interface....81
Working with Fluid Page and Component PeopleCode....81
PeopleCode Processing Order....81
Page Type PeopleCode....81
Using Fluid Component Level PeopleCode....81
Working with Grids....82
Meta HTML....82
Additional PeopleCode Constructs....83
Determining Browser and Device Type....83
Using Drag and Drop PeopleCode....84
Working with Grouplet PeopleCode....84
Working With DoModal Popup PeopleCode....85
Working with the Back Button....85
Managing Styles....85
Chapter 7: Managing Fluid User Interface Deployments....87
Configuring Fluid Mode Web Profile Settings....87
Web Profile Configuration - General Page....87
Web Profile Configuration - Look and Feel Page....88
Setting Fluid Homepage Personalizations....89
Configuring the NavBar to Appear in the Classic Portal Header....89
Configuring Fluid Mode PeopleTools Options Settings....91

Managing Fluid Homepages....92
Configuring Tile Options....93
Enabling Fluid User Interface for PC's and Laptops....97
Enabling the Web Profile Settings for Fluid Applications on the Desktop....97
Setting the Homepage Personalization Options for Fluid Applications....97
Configuring the NavBar to Appear in the Classic Portal Header....98
Managing Mobile Device Access....98

Preface

Understanding the PeopleSoft Online Help and PeopleBooks

The PeopleSoft Online Help is a website that enables you to view all help content for PeopleSoft Applications and PeopleTools. The help provides standard navigation and full-text searching, as well as context-sensitive online help for PeopleSoft users.

PeopleSoft Hosted Documentation

You access the PeopleSoft Online Help on Oracle's PeopleSoft Hosted Documentation website, which enables you to access the full help website and context-sensitive help directly from an Oracle hosted server. The hosted documentation is updated on a regular schedule, ensuring that you have access to the most current documentation. This reduces the need to view separate documentation posts for application maintenance on My Oracle Support, because that documentation is now incorporated into the hosted website content. The Hosted Documentation website is available in English only.

Note: Only the most current release of hosted documentation is updated regularly. After a new release is posted, previous releases remain available but are no longer updated.

Locally Installed Help

If your organization has firewall restrictions that prevent you from using the Hosted Documentation website, you can install the PeopleSoft Online Help locally. If you install the help locally, you have more control over which documents users can access and you can include links to your organization's custom documentation on help pages.

In addition, if you locally install the PeopleSoft Online Help, you can use any search engine for full-text searching. Your installation documentation includes instructions about how to set up Oracle Secure Enterprise Search for full-text searching.

See *PeopleTools Installation* for your database platform, "Installing PeopleSoft Online Help." If you do not use Secure Enterprise Search, see the documentation for your chosen search engine.

Note: Before users can access the search engine on a locally installed help website, you must enable the Search portlet and link. Click the Help link on any page in the PeopleSoft Online Help for instructions.

Downloadable PeopleBook PDF Files

You can access downloadable PDF versions of the help content in the traditional PeopleBook format. The content in the PeopleBook PDFs is the same as the content in the PeopleSoft Online Help, but it has a different structure and it does not include the interactive navigation features that are available in the online help.

Common Help Documentation

Common help documentation contains information that applies to multiple applications. The two main types of common help are:

- Application Fundamentals
- Using PeopleSoft Applications

Most product families provide a set of application fundamentals help topics that discuss essential information about the setup and design of your system. This information applies to many or all applications in the PeopleSoft product family. Whether you are implementing a single application, some combination of applications within the product family, or the entire product family, you should be familiar with the contents of the appropriate application fundamentals help. They provide the starting points for fundamental implementation tasks.

In addition, the *PeopleTools: Applications User's Guide* introduces you to the various elements of the PeopleSoft Pure Internet Architecture. It also explains how to use the navigational hierarchy, components, and pages to perform basic functions as you navigate through the system. While your application or implementation may differ, the topics in this user's guide provide general information about using PeopleSoft Applications.

Field and Control Definitions

PeopleSoft documentation includes definitions for most fields and controls that appear on application pages. These definitions describe how to use a field or control, where populated values come from, the effects of selecting certain values, and so on. If a field or control is not defined, then it either requires no additional explanation or is documented in a common elements section earlier in the documentation. For example, the Date field rarely requires additional explanation and may not be defined in the documentation for some pages.

Typographical Conventions

The following table describes the typographical conventions that are used in the online help.

Typographical Convention	***Description***
Key+Key	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For Alt+W, hold down the Alt key while you press the W key.
. . . (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe (|).
[ ] (square brackets)	Indicate optional items in PeopleCode syntax.

Typographical Convention	*Description*
& (ampersand)	When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object. Ampersands also precede all PeopleCode variables.
⇒	This continuation character has been inserted at the end of a line of code that has been wrapped at the page margin. The code should be viewed or entered as a single, continuous line of code without the continuation character.

ISO Country and Currency Codes

PeopleSoft Online Help topics use International Organization for Standardization (ISO) country and currency codes to identify country-specific information and monetary amounts.

ISO country codes may appear as country identifiers, and ISO currency codes may appear as currency identifiers in your PeopleSoft documentation. Reference to an ISO country code in your documentation does not imply that your application includes every ISO country code. The following example is a country-specific heading: "(FRA) Hiring an Employee."

The PeopleSoft Currency Code table (CURRENCY_CD_TBL) contains sample currency code data. The Currency Code table is based on ISO Standard 4217, "Codes for the representation of currencies," and also relies on ISO country codes in the Country table (COUNTRY_TBL). The navigation to the pages where you maintain currency code and country information depends on which PeopleSoft applications you are using. To access the pages for maintaining the Currency Code and Country tables, consult the online help for your applications for more information.

Region and Industry Identifiers

Information that applies only to a specific region or industry is preceded by a standard identifier in parentheses. This identifier typically appears at the beginning of a section heading, but it may also appear at the beginning of a note or other text.

Example of a region-specific heading: "(Latin America) Setting Up Depreciation"

Region Identifiers

Regions are identified by the region name. The following region identifiers may appear in the PeopleSoft Online Help:

- Asia Pacific
- Europe
- Latin America
- North America

Industry Identifiers

Industries are identified by the industry name or by an abbreviation for that industry. The following industry identifiers may appear in the PeopleSoft Online Help:

- USF (U.S. Federal)
- E&G (Education and Government)

Using and Managing the PeopleSoft Online Help

Click the Help link in the universal navigation header of any page in the PeopleSoft Online Help to see information on the following topics:

- What's new in the PeopleSoft Online Help.
- PeopleSoft Online Help acessibility.
- Accessing, navigating, and searching the PeopleSoft Online Help.
- Managing a locally installed PeopleSoft Online Help website.

PeopleTools Related Links

Oracle's PeopleSoft PeopleTools 8.54 Documentation Home Page (Doc ID 1664613.1)

"PeopleTools Product/Feature PeopleBook Index" (PeopleTools 8.54: Getting Started with PeopleTools)

PeopleSoft Information Portal

My Oracle Support

PeopleSoft Training from Oracle University

PeopleSoft Video Feature Overviews on YouTube

Contact Us

Send us your suggestions Please include release numbers for the PeopleTools and applications that you are using.

Follow Us

Get the latest PeopleSoft updates on Facebook.

Follow PeopleSoft on Twitter@PeopleSoft_Info.

Chapter 1

Getting Started with Fluid User Interface Development

Fluid User Interface Overview

The PeopleSoft Fluid User Interface is designed to be a significant enhancement to the PeopleSoft's "classic" user interface, which has been the interface display on browsers for PeopleSoft end users for well over a decade. The PeopleSoft Fluid User Interface moves away from pixel-perfect page layout and provides greater flexibility with the enhanced use of cascading style sheets (CSS3), HTML5, and JavaScript (if needed).

PeopleSoft application fluid pages scale gracefully from large screen devices such as laptops and desktops down to the reduced viewing space of tablets and smartphones. Many commercial websites use a similar design model where the presentation and layout of information is adjusted dynamically to conform to the dimensions of the user's device. The Fluid User Interface design approach gives developers just this type of control over the user experience. When a larger screen size is detected (a screen with more "real estate") the application content will adjust and conform accordingly to fill the space. Similarly, if a smaller screen size is detected, non-essential information can be removed and the presentation of content will adjust to "flow" in a very usable fashion.

This documentation focuses on the tools, the approaches, and the techniques that application implementation teams will use to create, configure, and customize fluid application pages.

Note: The PeopleSoft Fluid User Interface is an evolving technology. As information becomes available or is requested, PeopleTools will post updates as needed on the following My Oracle Support page: Oracle's PeopleSoft PeopleTools 8.54: PeopleSoft Fluid User Interface Documentation Updates (Doc ID 1909955.1).

Implementing PeopleSoft Fluid User Interface

Developing or customizing PeopleSoft Fluid User Interface applications involves many of the same tools and techniques used for developing or customizing classic PeopleSoft applications. A working knowledge and some expertise in the most current, industry-standard internet development tools and techniques is also required. The following list contains the main tools and technologies Oracle recommends developers are familiar with prior to beginning a fluid application implementation.

- PeopleSoft Application Designer
- PeopleCode
- PeopleTools Portal Technology
- PeopleSoft Pure Internet Architecture

- HTML 5.0
- Cascading Style Sheets 3.0
- JavaScript

Chapter 2

Understanding PeopleSoft Fluid User Interface

Understanding the Need for a Fluid User Interface

A traditional, or classic, PeopleSoft application displayed in a browser assumes the screen size of the device running the application is of a standard, relatively predictable size, such as a browser running on a desktop or laptop computer. The classic PeopleSoft application *will run* on a mobile device, such as a tablet or cell phone, but it *is not* tailored or optimized for the mobile device.

With the traditional PeopleSoft application, the interface display is defined in Application Designer using a fixed layout, based on and bound to a pixel system. In the fixed layout, a developer can meticulously align fields, labels, group boxes, and so on, and be reasonably assured the page will appear in the browser running on a desktop or laptop just as it appears in the design-time layout. The traditional PeopleSoft application development design-time layout provides a WYSIWYG (what you see is what you get) scenario.

However, the fixed layout does not adapt -- expand or contract-- based on the device or the size of the screen on which it renders. A classic PeopleSoft application does not display dynamically. While an end user can access and use a traditional PeopleSoft application with a cellular phone, for example, most likely the user will need to shrink or expand the display in order to interact with the fields, with maybe only a small percentage of the page being visible at a time, losing the context of the information on the page.

With the emergence and popularity of mobile devices, namely tablets and smart cellular phones, PeopleSoft applications need to display and run on a wide range of mobile devices. For example, a small, light-weight self-service application must be equally usable on a browser running on a desktop, laptop, iPhone, Android, iPad, and so on.

To address this business demand, PeopleTools enables you to develop and customize components and pages for a *fluid display* when running on supported mobile devices. PeopleSoft applications that have been developed and enabled for fluid display will appear dynamically, depending on the device and screen size used for access, all based on a single page layout definition. That is, you create a *single* page definition, using the flexible fluid design approach, and it will display as needed on all the different supported devices, be it an Andriod phone or iPhone or tablet.

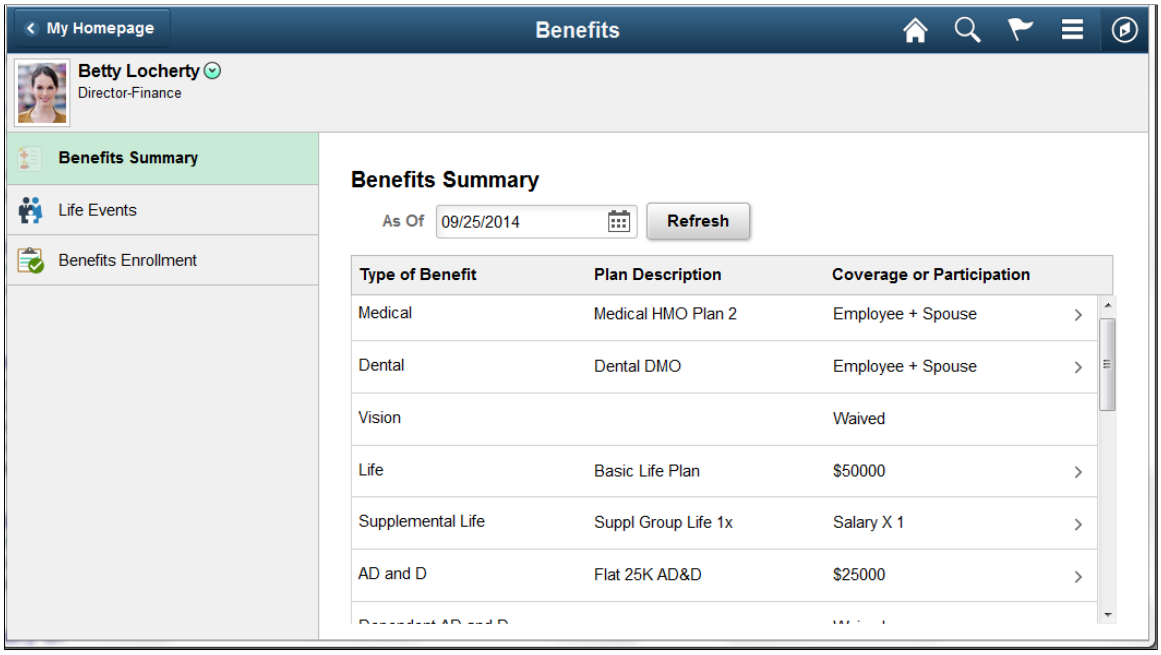
Terms used to describe similar design approaches in the industry include, *fluid design*, *responsive design, relative* layout, and *elastic* layout, to name a few. For PeopleTools, we refer to this approach as *fluid,* as in *fluid design, fluid applications, fluid mode* and so on. Regardless of the term used, the goal remains the same, which is to enable organizations to deploy applications to all the devices being used to conduct business such that the content of the PeopleSoft page resizes and adapts to the size of the browser window of the device

Understanding the Characteristics of Fluid Applications

A PeopleSoft application enabled for fluid mode recognizes the device used to access it, and displays so that the application appears naturally, as expected by the user of a certain device type. For example, the following shows a PeopleSoft application page enabled for fluid display appearing on a browser on a laptop or a tablet device.

Image: Fluid page

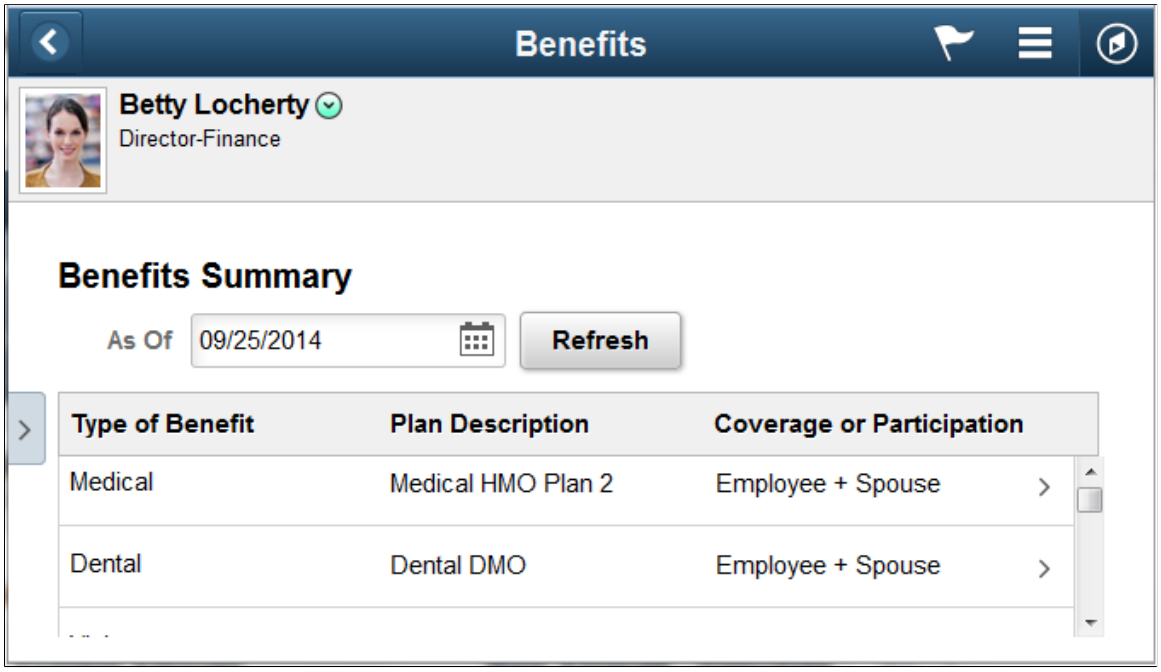
This example shows a fluid page displayed on a laptop or tablet.



If you access that same page using a cell phone and view it in landscape mode, it would appear similar to the following example.

Image: Fluid page in landscape mode

The following example shows how a fluid page may appear on a smartphone in landscape mode.



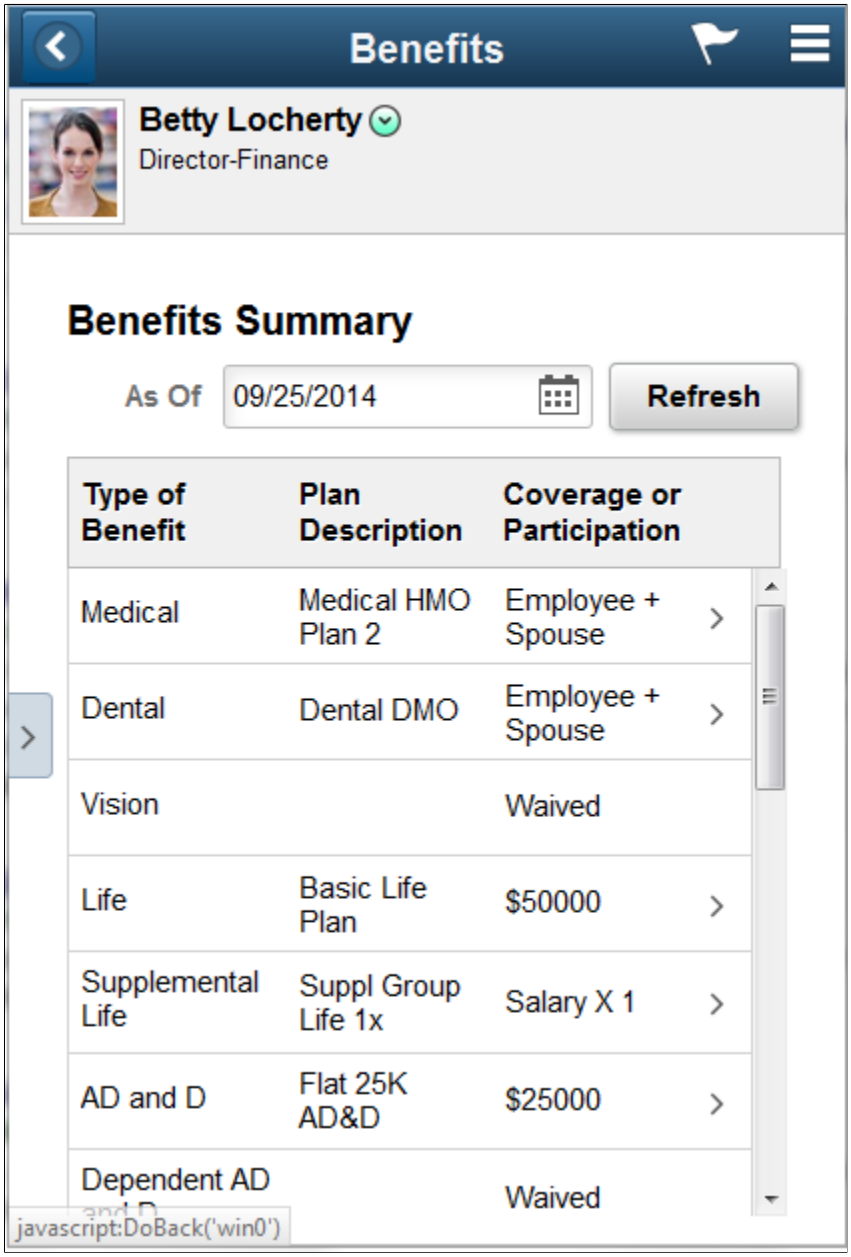
Notice that the page shrinks to fit the screen size of the device. The elements in the header are the same, only closer together, and the grid displays only the most critical columns. The user can use typical gestures, such as swiping (on homepages only), to scroll through the fluid page.

If you access the same page using a cell phone and view it in portrait mode, you will notice additional changes as the page further adapts and responds to the adjusted screen display. The PeopleSoft Fluid User Interface is *adaptive* in that the system detects the device and generates the UI appropriate to that device, and it is *responsive* in that the CSS defined for the application responds to user actions on the client device, such as changing from portrait to landscape mode.

Note: The PeopleSoft Fluid User Interface adaptive qualities require a server trip, while the responsive qualities do not.

Image: Fluid page in portrait mode

The following example shows a fluid page displayed on a smartphone in portrait mode.



Notice that the system further shrinks the page, while maintaining the fundamental structure.

Related Links

"Working With Fluid Homepages" (PeopleTools 8.54: Applications User's Guide)
"Working With Fluid Pages and Controls" (PeopleTools 8.54: Applications User's Guide)

Understanding How to Build Fluid Applications

PeopleSoft Fluid applications are built in large part the same as you build a traditional or classic PeopleSoft application, using the same general application development steps in Application Designer. You create fields, records, and build SQL tables just as you would a traditional PeopleSoft application, but when it comes to designing pages and configuring components, you depart somewhat from the steps and techniques used for building traditional PeopleSoft applications.

The following table contains the traditional PeopleSoft application development steps as they pertain to developing fluid applications.

Development Step	***Fluid Consideration***
1. Design application.	Review your in-house development cycle and documentation.
2. Create fields.	Create fields just as you would for a traditional PeopleSoft application. In most cases, the same fields that already exist would be used for any fluid development projects.
3. Create record definitions.	Create records just as you would for a traditional PeopleSoft application. In most cases, the same records that already exist would be used for any fluid development projects.
4. Build SQL tables.	Build SQL tables using the same techniques as for traditional PeopleSoft applications.
5. Create pages.	While you still use Application Designer to create the fluid page definitions, the approach, the layout and controls used are completely different than what is used for traditional PeopleSoft page development. CSS is used heavily for appearance and layout. JavaScript can be used, if required.
6. Create components.	Components used for fluid pages must be configured specifically for that purpose. The fluid component options must be selected and set for fluid pages within that component to display in fluid mode.
7. Register components.	Classic and fluid applications both need to have the components registered using the registration wizard.
8. Test the application.	Use emulators, cellular phones, tablets, and laptops.

Note: The majority of this documentation focuses on creating pages and creating components for fluid display.

Application Designer is the main development tool used for building both traditional and fluid PeopleSoft applications, and PeopleCode is the main programming language used to interact with component processing events and implement business logic. However, with fluid applications, the application implementation teams that will develop and\or customize PeopleSoft fluid applications will need expertise in these additional areas (listed in order of importance and recommended experience):

- CSS 3
- HTML 5

- JavaScript

Important! With the flexibility provided by supporting creating and customizing HTML, CSS style sheets, and JavaScript, it is up to the developer to determine the amount of development work performed outside of the realm of PeopleTools. For example, PeopleTools cannot implement restrictions on what a developer adds to a freeform style sheet or to a JavaScript program. Developers customizing such elements need to be aware of performance and security implications.

Note: After reading this document, your application documentation, and any related information related to PeopleSoft Fluid User Interface, one very effective way to learn about fluid applications is to use the fluid applications delivered with your PeopleSoft application as examples. Consider and explore the intended audience, the capabilities of the pages, the page design, the page controls used, the component contents and settings, and so on. Especially, pay close attention to how CSS style classes are applied.

Considerations for PeopleSoft Fluid Application Implementation

Not all PeopleSoft application pages will be delivered for the PeopleSoft Fluid User Interface. The majority of PeopleSoft applications, components, and pages will, for the foreseeable future, continue to be delivered and accessed in classic mode, which is the way PeopleSoft applications have been accessed using desktops and laptops for over a decade. PeopleSoft application teams will be offering selected applications for fluid deployment that your organization can choose to enable and deploy.

In most cases, there is no quick and simple method for converting an existing custom classic application to a fluid application. Plenty of time should be spent analyzing requirements and discussing design options and considerations. In some cases, adjusting a classic page following the fluid approach to page design may be all that's required, while in other cases, you may consider building a new application from the ground up, perhaps reducing the number of fields and records required, and streamlining the entire data model.

Converting an application to fluid and bringing with it *all* the classic application techniques and elements may result in poor performance and usability on a mobile device. While some of the structural elements of the application must be analyzed and adjusted for fluid application deployment, such as page layout, some elements, such as PeopleCode business logic, can be used in either classic or fluid mode with little, if any, modification.

Before you consider deploying any PeopleSoft applications in fluid mode, it is important to consider the following items, which may help you to determine if fluid applications are appropriate for your user base at this time.

Consideration	***Discussion***
Not all applications (currently) are candidates for fluid deployment.	Currently, the applications that are most suited for fluid deployment are those that are already considered streamlined, self-service type applications. For example, accessing a page to check a new hire’s job status would be a more appropriate fluid application as opposed to accessing a component with a cell phone to complete a new hire’s “add employee” business process.

Consideration	*Discussion*
Are mobile devices such as cell phones and tablets used widely to access PeopleSoft applications at your site?	If your user base already accesses or has expressed a demand for accessing your PeopleSoft applications using mobile devices, fluid deployment would be a good option for you. However, if mobile devices are not part of your user base or application delivery strategy, you can elect to disable fluid deployment at the system level.
Application usage and workload?	Pages where there is heavy data entry required would not be suitable for phones or even tablets. The functional use of the page must be paramount.
Does an application *require* functionality that does not exist in the fluid user interface?	For example, some features, like spell check, rich text editor, Find In for grids, Download to Excel, modal movement or re-sizing, are not available in fluid.

Note: Currently, bidirectional languages (right-to-left), such as Arabic and Hebrew, are not certified with the PeopleSoft Fluid User Interface.

Chapter 3

Creating Pages for PeopleSoft Fluid Applications

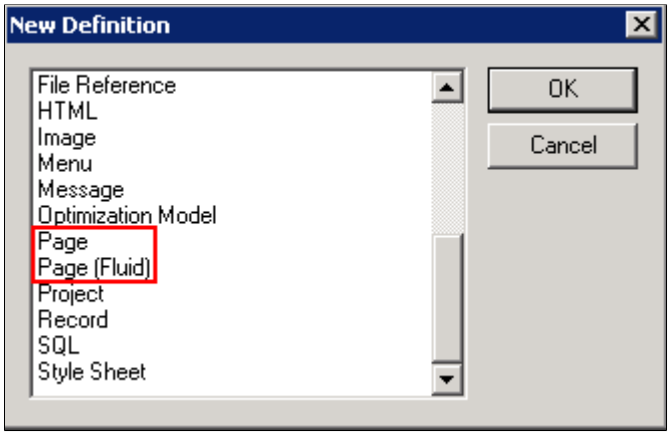
Selecting Page Types

When developing applications for fluid mode, you can create multiple page types for simultaneous display, each page serving a different role in accordance with HTML 5 specifications. The main page is referred to as the fluid page, which is a standard page that has been designated a Fluid Page on the Use tab of the Page Properties. The fluid page definition contains the bulk of the transaction fields and controls.

When creating a new fluid page or opening an existing fluid page, you can select *Page (Fluid)* from the definition list in Application Designer.

Image: New Definition dialog box

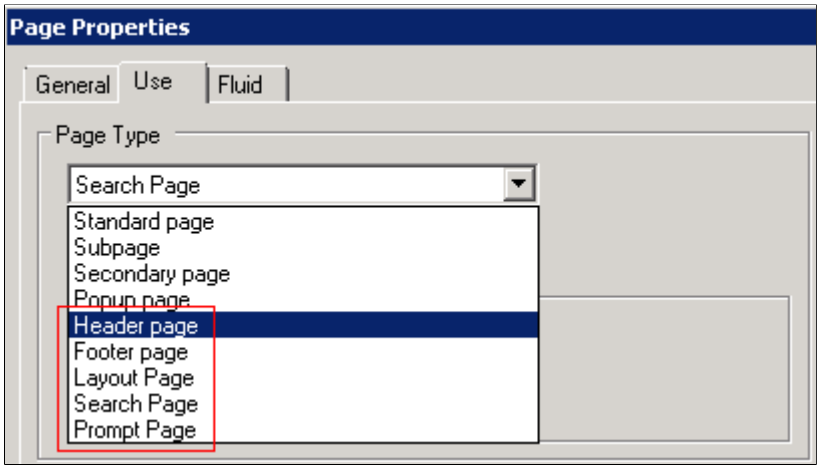
The following example shows the Page and Page (Fluid) definition types in the definition list in Application Designer.



You select the remaining page types for fluid mode, from the Page Type drop-down list on the Use tab of the Page Properties dialog.

Image: Page Properties dialog box

This example shows the fluid page types listed in the Page Type drop-down list on the Page Properties, Use tab.



While each fluid page type will be discussed separately in detail, the following table provides a brief description of each fluid page type.

Page Type	***Description***
Fluid Page	A fluid page is a standard page that has been designated a Fluid Page on the Use tab of the Page Properties. The fluid page definition contains the bulk of the transaction fields and controls.
Header Page	Page displayed in <header> section of the HTML acting as the banner area fixed at the top of every page. It does not scroll with the page). The header page appears at the top of the browser interface over the main page, containing elements used for basic navigation and orientation, like a back button, page title, home button, search button, and so on. PeopleTools supplies a default header page, PT_HEADERPAGE, but this can be overridden for a custom header page to be used by selecting No System Header Page in the Component Properties dialog box.
Footer Page	Page displayed in the <footer> section of the HTML at the bottom of the main page, containing elements related to the end of a transaction, such as a Save button. The footer page is fixed at the bottom of the content page and does not scroll.
Search Page	Page generated in <aside> section containing search pages, which include Find an Existing Value and Keyword Search. **Note:** An Add a New Value page is not provided by the search page definitions provided by PeopleTools.

Page Type	*Description*
Layout Page	You can designate a page definition as a layout page for use as a template for development reuse or for testing purposes, such as finalizing CSS work. **Note:** This is a base layout page, which can be used when selecting a new Page (Fluid). At that time you can select a layout page where the structure will be copied into your new page. You cannot apply a layout page after you have created a page or if you create a new a new page (regular).

Creating the Primary Fluid Page

The fluid primary page is the main content page, or target page. The primary page is used directly within the component and is not modal (unless used in the context of DoModalComponentPopup).

Creating a New Primary Fluid Page

To create a new primary fluid page based on an existing layout template:

1. In Application Designer, select File, New.
2. Select Page (Fluid) on the New Definition dialog box, and click OK.
3. Select an initial layout page (template) from the Choose Layout Page and click Choose.
4. Enter the page name on the Save As dialog box.
5. Indicate whether you'd like to save the PeopleCode associated with the layout page for your new page (Yes/No).
6. Define layout and add page controls according to the fluid techniques and guidelines described in this documentation.

Cloning and Existing Page for Fluid Deployment

To create a primary fluid page from an existing standard page:

1. Open the existing standard page in Application Designer.
2. (Recommended) Select File Save As, and rename the page to distinguish it from the standard, classic PIA page.

   Note: Generally, for any fluid page, there would be an original counterpart, unless the page or application is new, and intended only for a mobile, fluid deployment.

3. Define layout and add or remove page controls according to the fluid techniques and guidelines described in this documentation.
4. Access the Page Properties dialog box, select the Use tab, and enable the Fluid Page check box.

Note: For a classic page we define a visual layout with Application Designer. With fluid pages, you design HTML structure, where styles are applied to implement visual changes to that structure.

5. Apply CSS style classes as needed to arrive at the appropriate look and feel for your page.

Choosing a Layout Page

This section describes the page layout templates you can use as the basis for any new fluid pages. Select a template, and save the page with a name other than the template name. Then you may add or remove elements as needed for your purposes. Make sure to examine the template pages so that you understand how any existing PeopleCode or style application affects the interface.

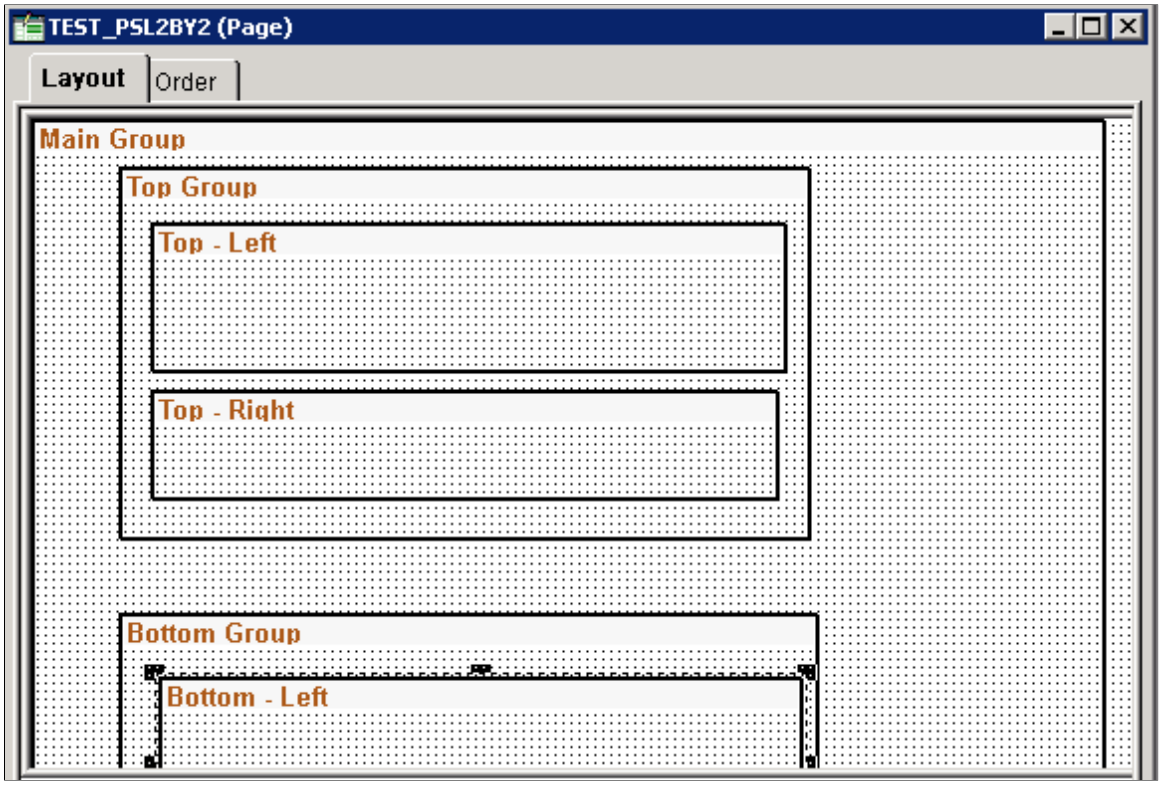
On the Choose Layout Page dialog box, you can select one of these options to use as a template for your new fluid page.

PSL_2BY2

The 2BY2 layout template contains a top group box and a bottom group box, with a left and right group box within each.

Image: Fluid Layout Template: PSL_2BY2

This example illustrates the delivered fluid page layout template: PSL_2BY2.

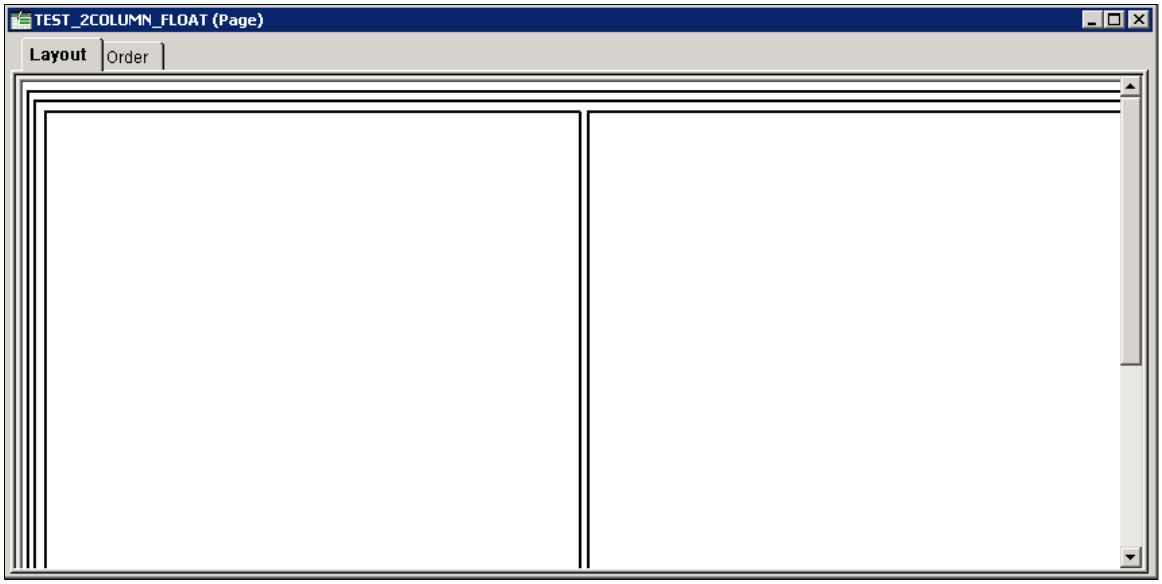


PSL_2COLUMN_FLOAT

The 2COLUMN_FLOAT template provides a simple starter pages where a two-column layout is needed.

Image: Fluid Layout Template: PSL_2COLUMN_FLOAT

This example illustrates the delivered fluid page layout template: PSL_2COLUMN_FLOAT.

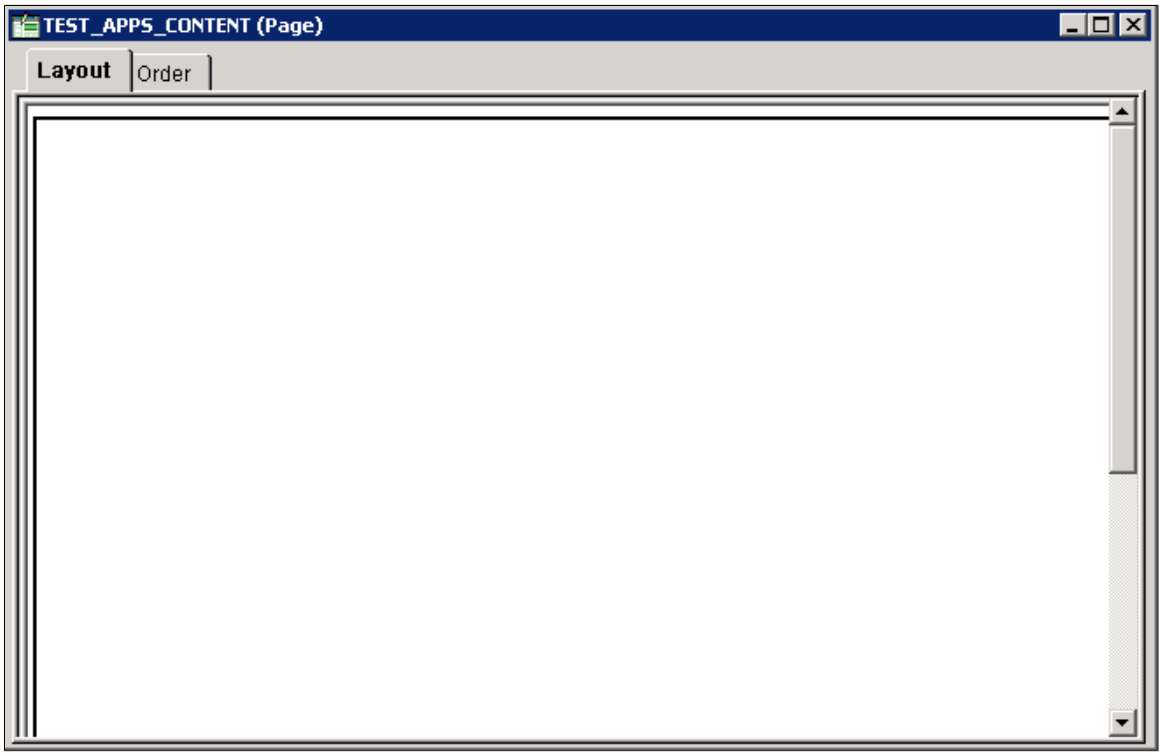


PSL_APPS_CONTENT

Used as the basis for any page. It contains a single group box container that can contain all additional elements in the page. The group box has the ps_apps_content style class applied.

Image: Fluid Layout Template: PSL_APPS_CONTENT

This example illustrates the delivered fluid page layout template: PSL_APPS_CONTENT.



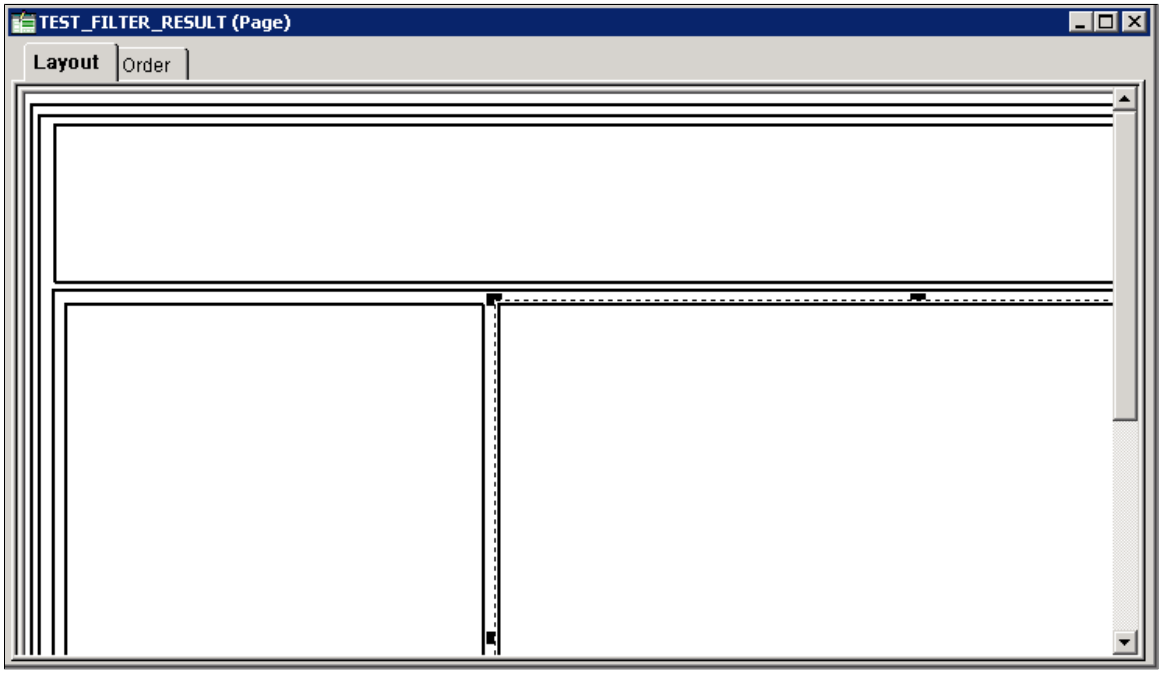
PSL_BOUNDING_BOX

Used to contain the contents within the box within the maximum allowable space. This page contains the PSL_APPS_CONTENT layout group box, with another inside with the ps_box-boundingbox style class applied.

PSL_FILTER_REQUESTS

Image: Fluid Layout Template: PSL_FILTER_REQUESTS

This example illustrates the delivered fluid page layout template: PSL_FILTER_REQUESTS.



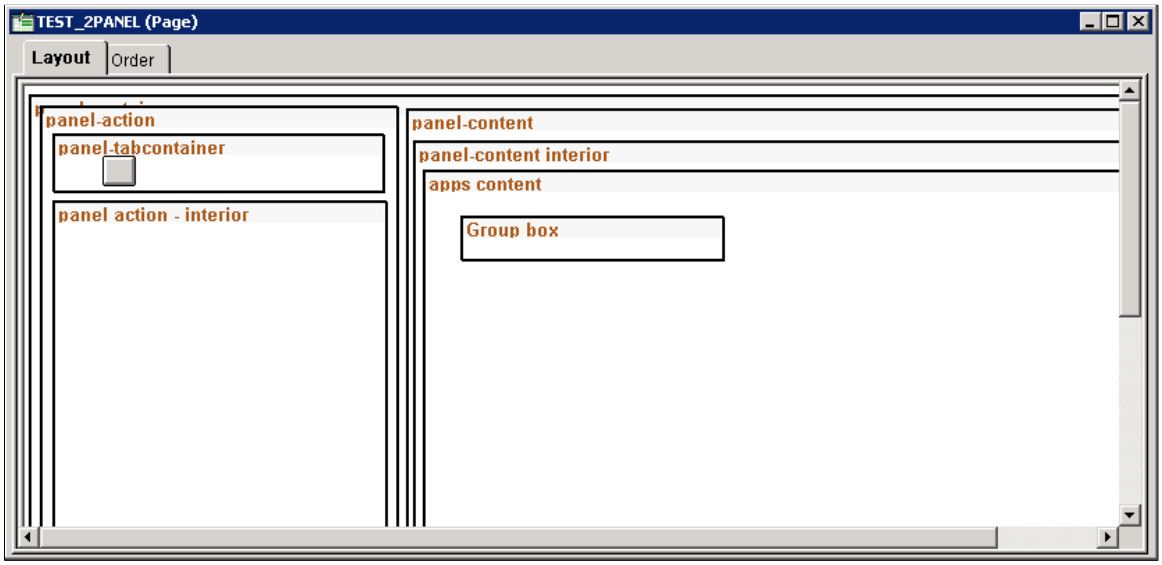
PSL_SIMPLE_SBP

A subpage template, similar to PSL_APPS_CONTENT layout template, only smaller in size.

PSL_TWOPANEL

Image: Fluid Layout Template: PSL_TWOPANEL

This example illustrates the delivered fluid page layout template: PSL_TWOPANEL.



PTPN_SEEALL, PTPG_NUI_VWR, PTPG_GRIDVIEWERNUI

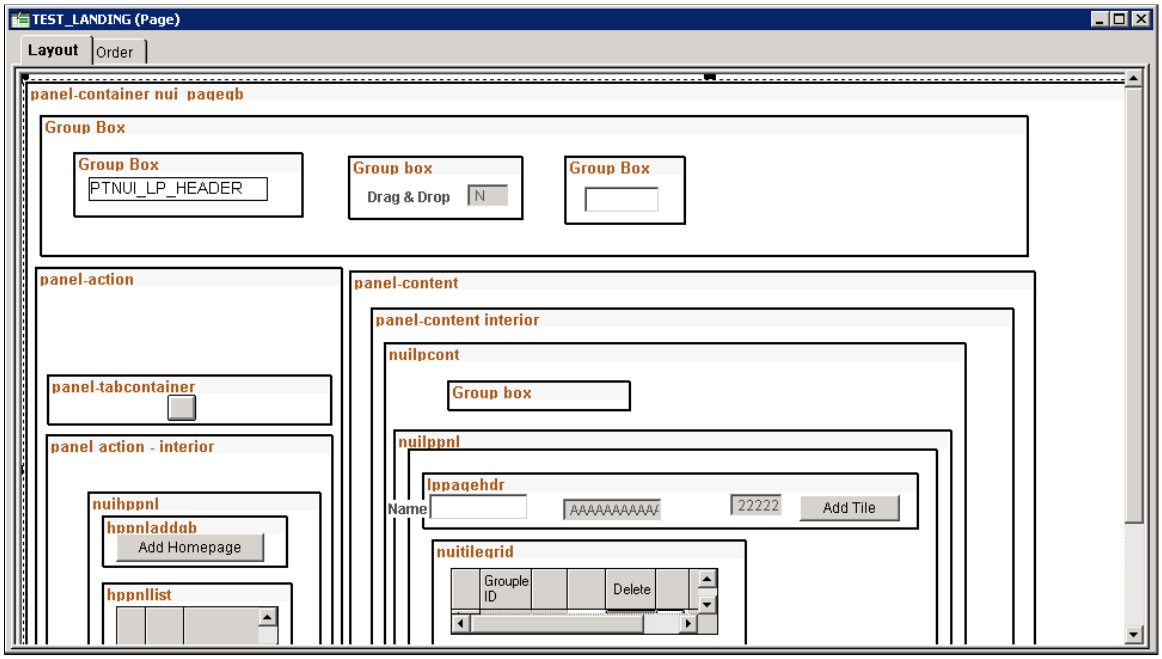
These layout pages are used to display pivot grids.

PT_LANDINGPAGE

Use to create custom fluid homepages.

Image: Fluid Layout Template: PT_LANDINGPAGE

This example illustrates the delivered fluid page layout template: PT_LANDINGPAGE.



Setting Properties for Fluid Pages

You create fluid pages using Application Designer and use the same Page Properties dialog box to set the properties for a fluid page. This topic describes the settings that are specific to fluid pages. For general page development information refer to the product documentation for Application Designer, the *Application Designer Developer'sGuide.*

Related Links

"Understanding Page Design and Development" (PeopleTools 8.54: Application Designer Developer's Guide)

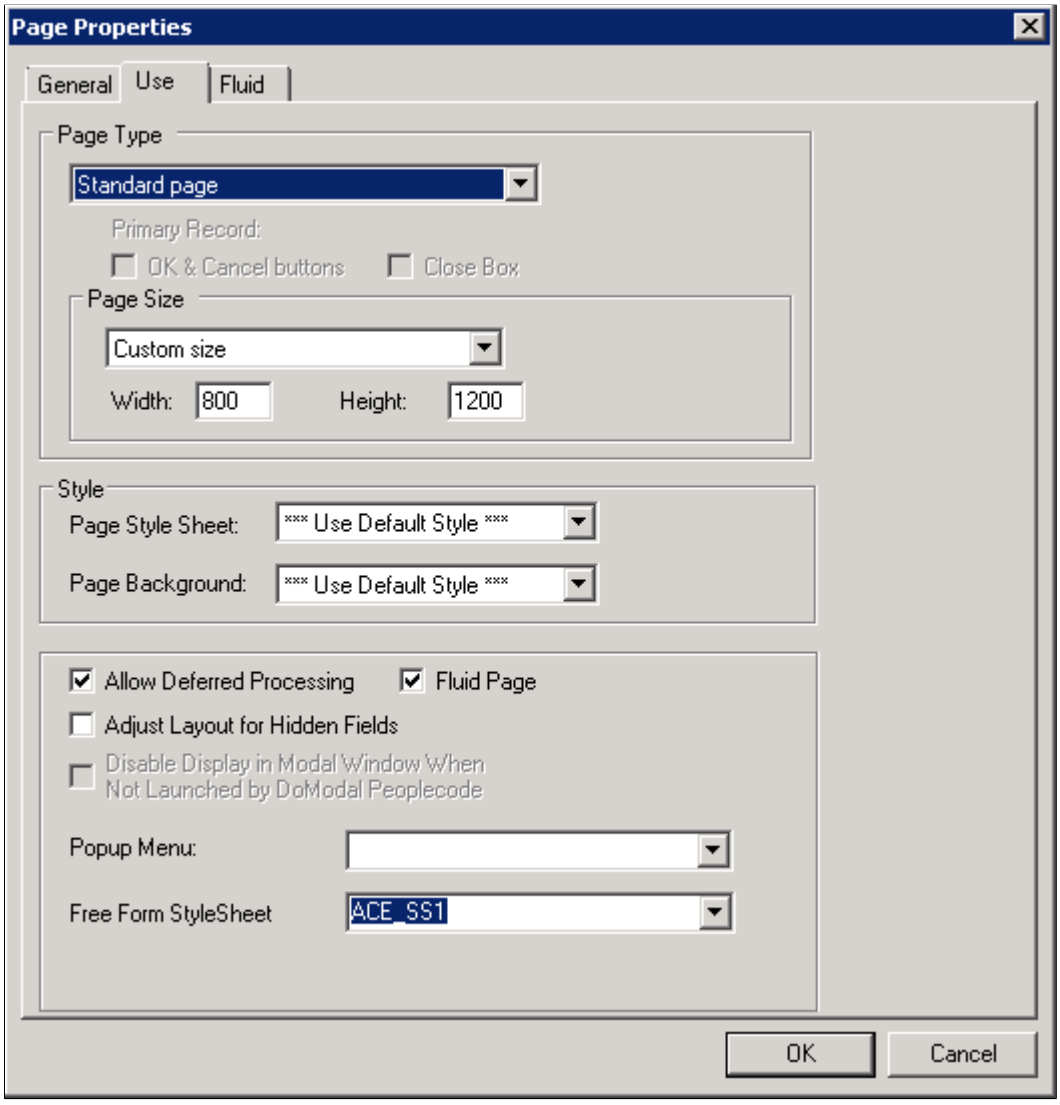
"Creating New Page Definitions" (PeopleTools 8.54: Application Designer Developer's Guide)

Setting Use Properties

On the Use tab in the Page Properties dialog box, some properties need to be described separately in the context of the fluid application (not classic PIA).

Image: Page Properties > Use tab

This example shows the Use tab in the Page Properties dialog box in Application Designer. The following descriptions cover the properties that are specific to fluid pages.



Page Type

Select the type of page for use in the fluid application, such as header, footer, search, and so on.

For more information on page types used for fluid development, see Selecting Page Types .

Fluid Page

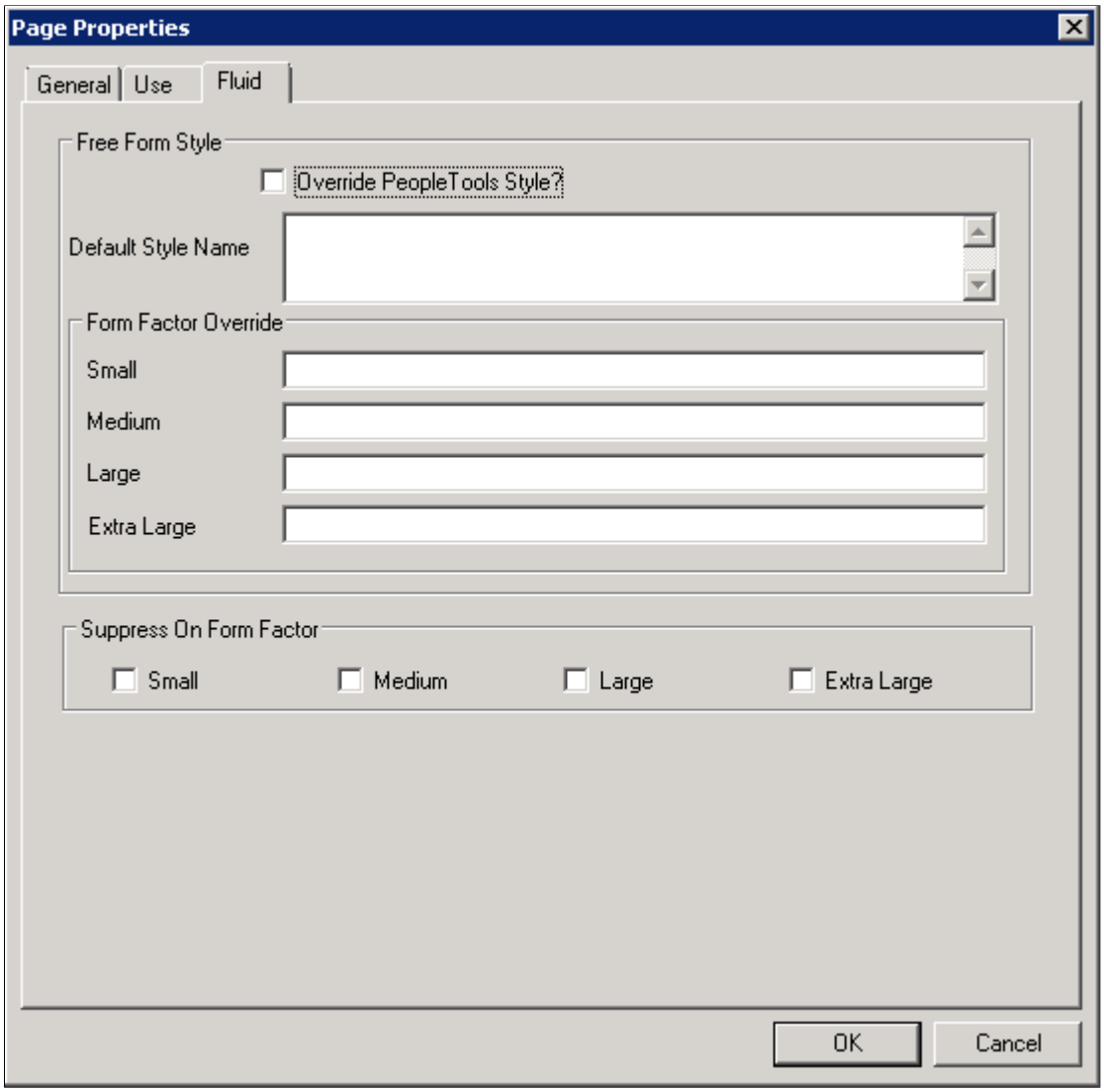
For a page to be displayed and rendered in fluid mode, the Fluid Page check box *must* be selected. This is required.

Setting Fluid Properties

The Fluid tab contains a collection of properties that apply only to fluid pages.

Image: Page Properties > Fluid tab

This example shows the Fluid tab in the Page Properties dialog box in Application Designer. The following descriptions cover the properties that are specific to fluid pages.



Free Form Style

Override Tools Style Select to bypass the system default style completely.

If not selected, the custom free form style specified will be appended to the default system style, such that both style sheets are used.

Default Style Name Enter the custom default style name to include.

Form Factor Override

Use to override the style based on the form factor (device size) of the device accessing the page. Form factors are represented in these ranges:

- Small: Applies to most smartphones.
- Medium: Applies to 7-inch tablets.
- Large: Applies to 10-inch tablets and laptops.
- Extra Large: Applies to large monitors.

Suppress on Form Factor

Use to prevent the page from displaying for specified form factors. Options are:

- Small
- Medium
- Large
- Extra Large

For example, if you select Extra Large, the page would not appear in fluid mode on the desktop PC. It would only appear in fluid mode on a phones, tablets, and laptops. This also aids performance because if it is not set to appear on a specific device it doesn't just prevent display of the page to the end user, the system does not send that page to the device at all, meaning the system saves on network traffic, rendering, and CSS processing.

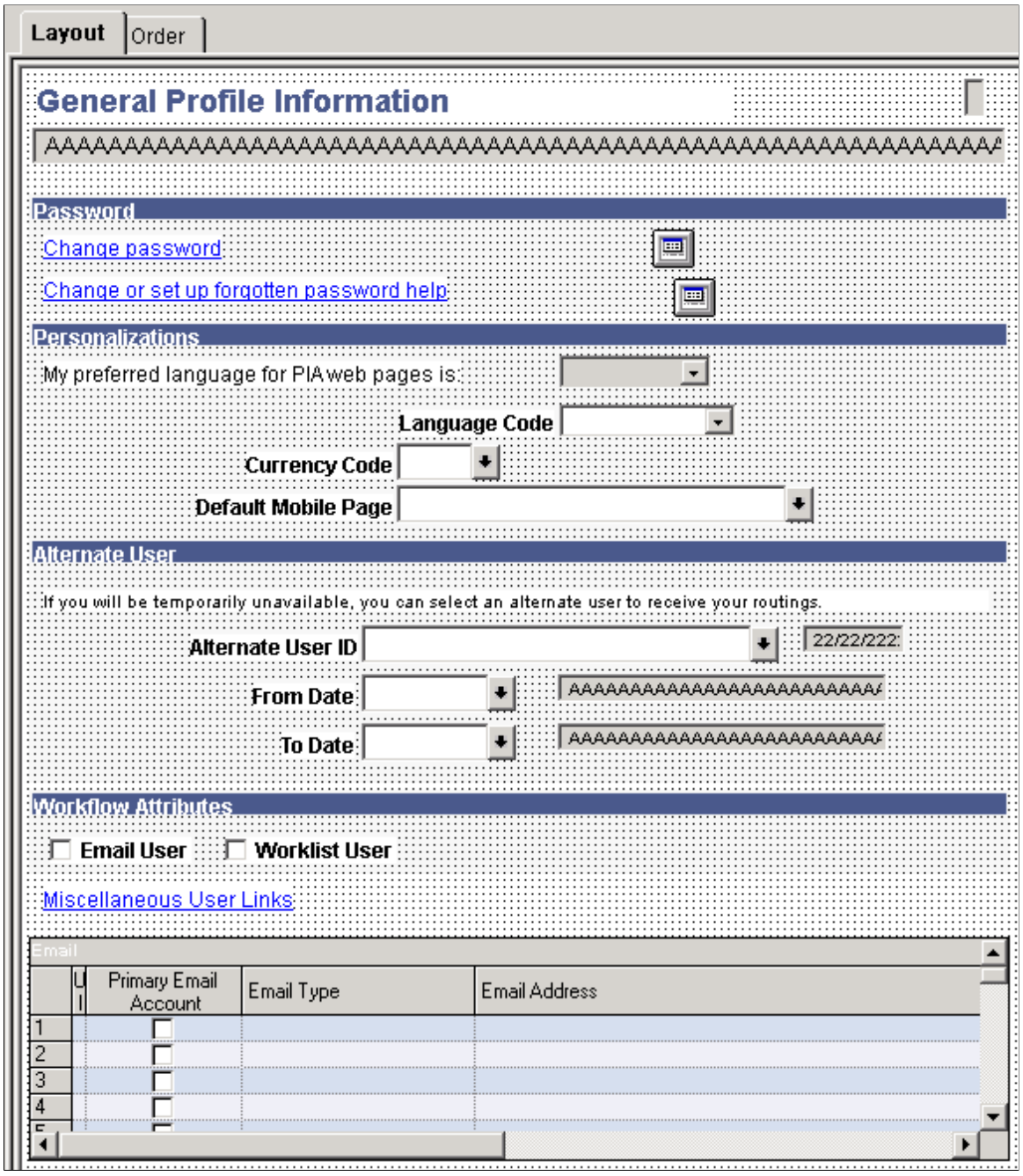
Defining the Layout of the Main Fluid Page

Designing and defining the layout of a fluid page is one of the major differences between developing fluid applications and developing classic PIA applications. In a classic PIA application, you perform layout of a

page in a WYSIWYG format, where page elements and controls are placed exactly where you want them within a pixilated invisible table.

Image: Classic page definition at design time

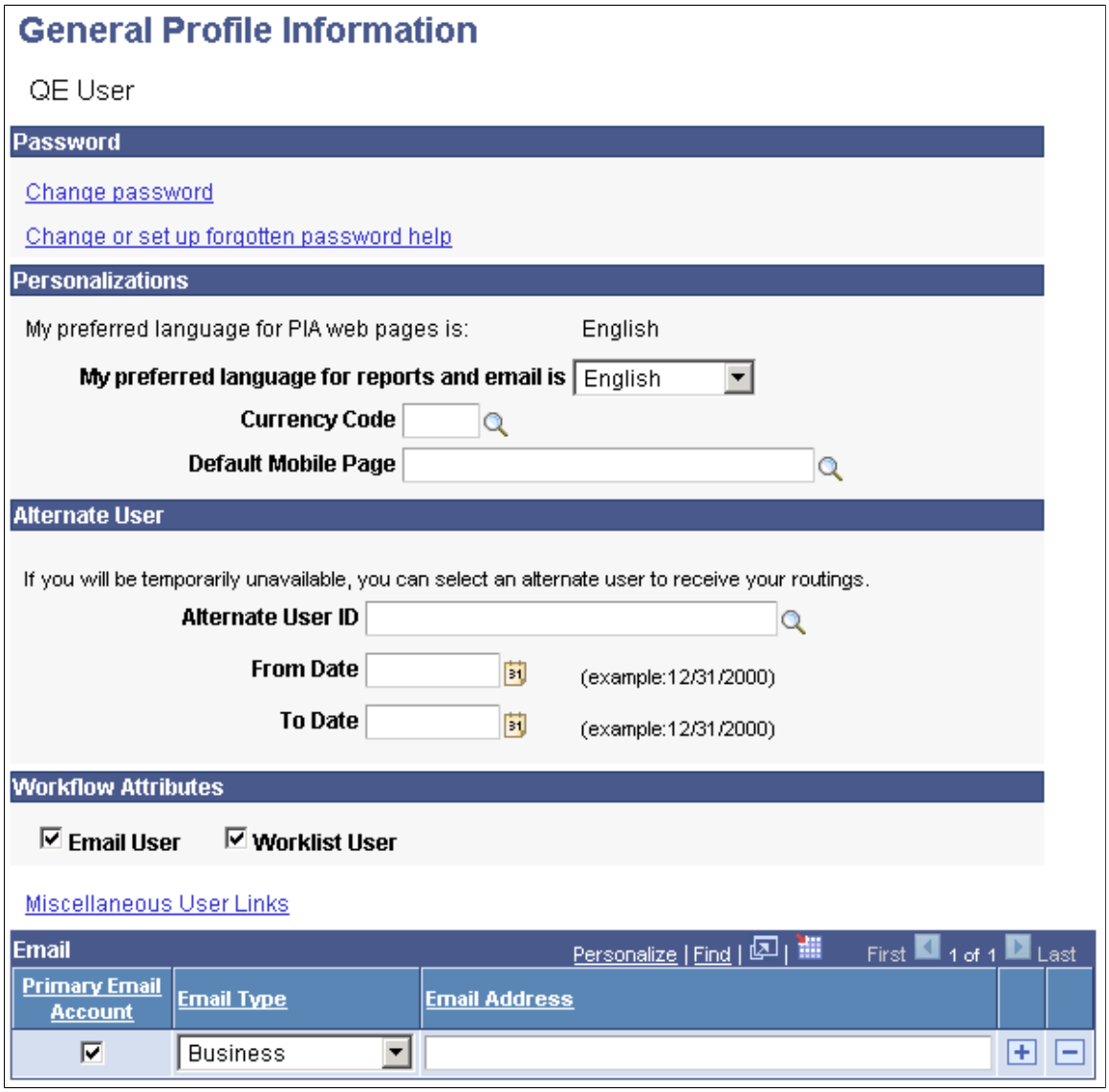
This example illustrates a classic PeopleSoft page definition in Application Designer, where each page element is meticulously placed on the Layout tab.



And, despite the obviously more polished presentation in the generated HTML page that displays in the browser interface for the end user, the layout, spacing, order, color, and so on is for the most part identical between the design time definition and the runtime presentation.

Image: Classic page at runtime

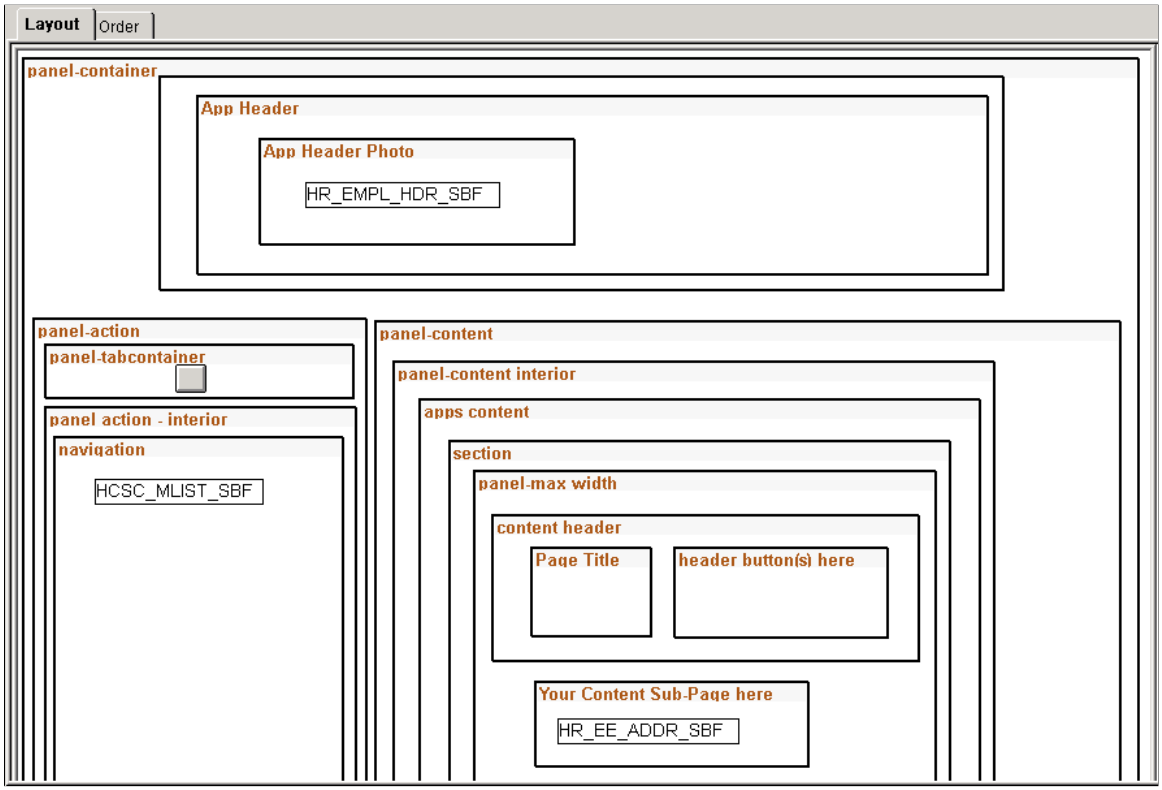
This example illustrates the same page as depicted in the previous example at runtime. The design time and runtime appearance is practically identical in terms of layout, positioning, sizing, color, font, and so on.



When performing layout for a fluid page, you will notice that design representation in no way resembles the design representation for the classic PIA page.

Image: Fluid page at design time

This example shows a fluid page layout at design time in Application Designer, with various elements contained in nested group boxes and little if any indication regarding how the page might appear at run time to the end user.

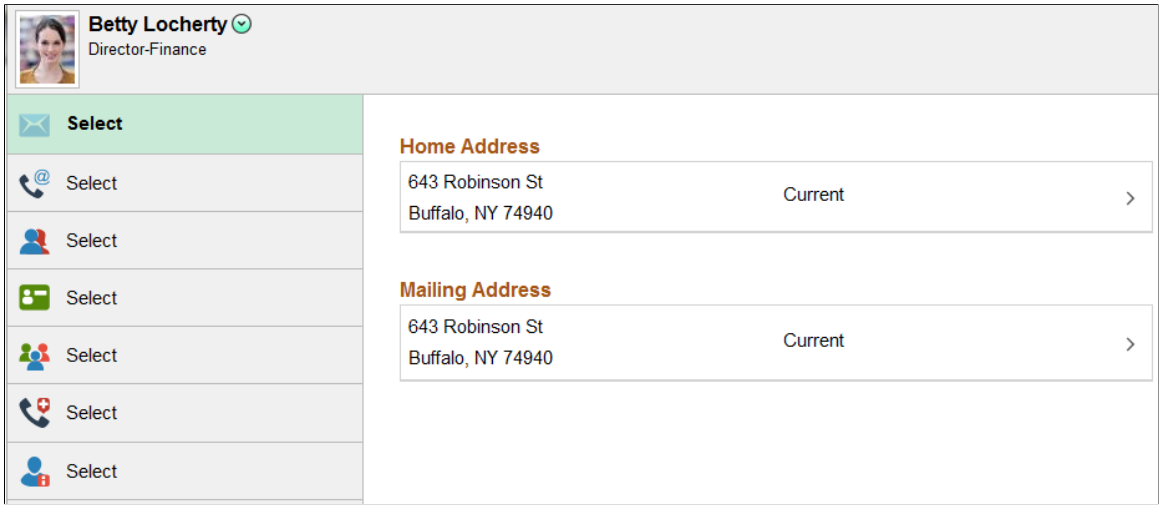


The layout of the fluid application is not what you would call a WYSIWYG approach. If an ender user were to access a page in this state, it would be impossible to complete a transaction. Notice the boxes that are used to indicate levels of containment and to provide visual cues, making each element from

high-level to low-level a self contained and discrete. Group boxes are used to achieve this effect. The rectangular boxes within some of the group boxes represent subpages.

Image: Fluid page at run time

This example illustrates the fluid rendering of this very same page definition (above) in a browser.



The strip running across the top displays the default PeopleTools, or system, header page definition. Because it is a separate page definition entirely, it is not depicted in the layout view of the main page (the content below it) in Application Designer.

At a high-level, you can map the rendering in the browser to the layout as:

- The area showing the employee photo is represented in the layout as the App Header Photo group box.
- The area showing the navigation on the left is represented in the layout as the Left Side Navigation group box.
- The area showing the Name and Addresses fields is represented in the layout as the Content group box. When Names and Addresses is selected from the left hand navigation, it appears in the Content container. When Contact Details is selected, the contact information is displayed in the Content container, and so on.

While there are many factors contributing to the appearance of a fluid page, which will be discussed in this document, the main elements of fluid page layout are:

PeopleSoft Design Elements	***Description***
group boxes	Group boxes are the primary control, providing not only visual cues, but, more importantly, it acts as the container for *all* controls and page elements. For example, subpages must be enclosed in a group box, grids and scroll areas need to be enclosed in a group box, and so on. Each control is comprised of two parts: the label and the control – wrapped in the container of a group box. In fact, the entire page itself *must* be contained in one super group box: a requirement for fluid rendering.

PeopleSoft Design Elements	*Description*
style sheets	While group boxes take on the role of discretely containing each and every element on a page, the physical placement of the group boxes within the Layout tab in Application Designer is largely irrelevant. How and where a page element or control renders on the end user device is determined by style sheets. PeopleTools delivers the system-wide PSSTYLEDEF_ FMODE style sheet. You can make custom, free form style sheets to extend or override the system-wide style sheet.

Both group boxes and free form style sheets are discussed in detail in other sections of this document.

Note: Tab order index is determined only by the order in which the fields and controls appear on the page from top-to-bottom and left-to-right.

Chapter 4

Defining Components for Fluid Applications

Adding Fluid Pages to a Fluid Component

Like classic PIA applications, the fluid application uses components to contain sets of pages. With all the different page types that are included within a single fluid transaction (header pages, footer pages, primary pages, and so on) they must all be included within the same component so they run in the same panel buffer. Having these pages all in the same panel buffer means you can write PeopleCode to work on all page types at the same time.

Note: When using the default system pages, as in the default header page provided by PeopleTools, the system automatically loads those pages into the component. If you intend to use those pages, they do not need to be added manually to the component. However, if you have built custom header pages, search pages, and so on, those pages need to be inserted manually into the component to associate them with that component so the system loads them at runtime.

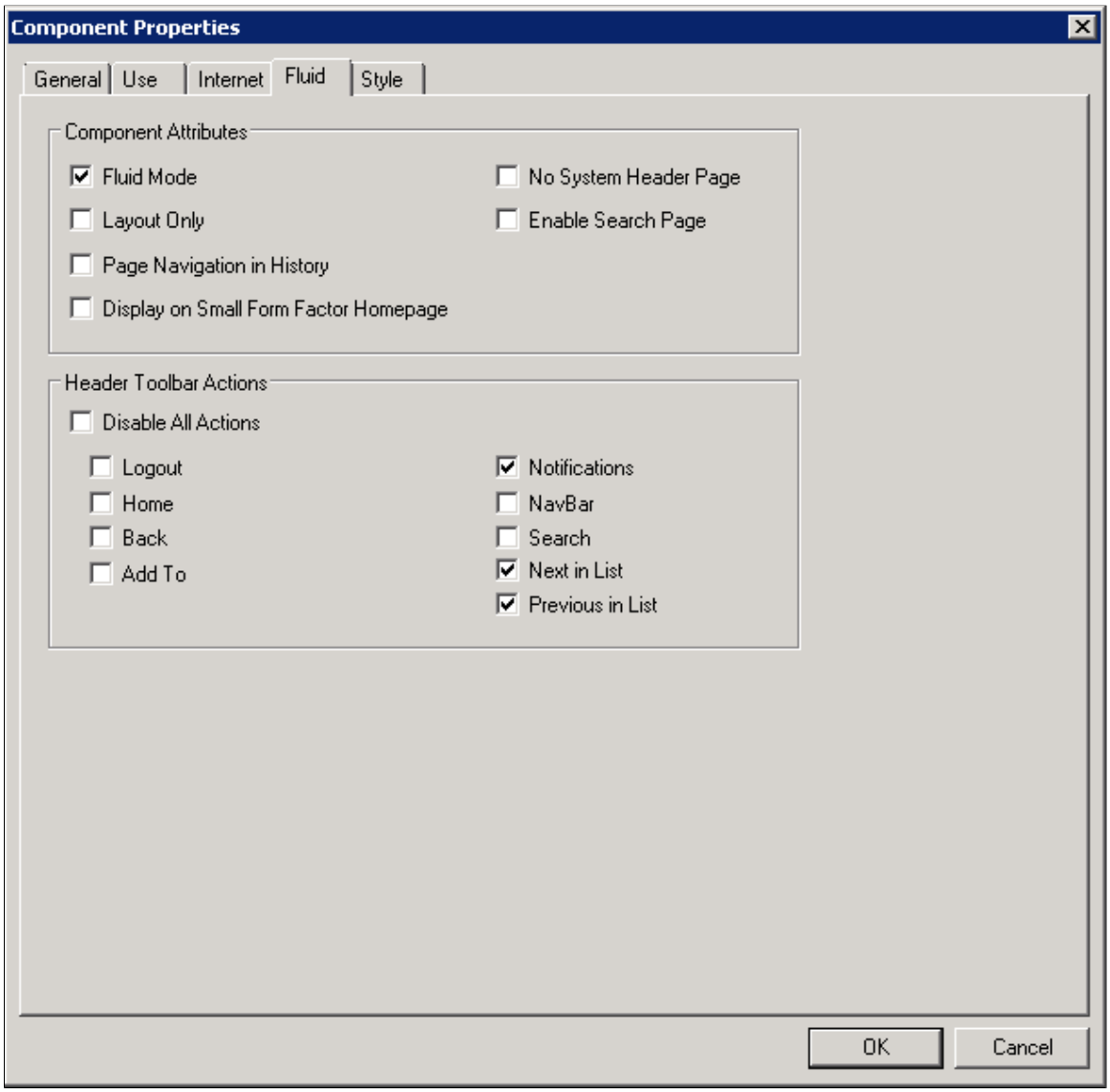
Setting Component Properties for Fluid Components

Setting Fluid Component Properties

Use the Component Properties, Fluid tab to set fluid component properties.

Image: Component Properties dialog box: Fluid tab

This example illustrates the options on the Fluid tab. You can find descriptions for the options after the example.



Component Attributes

Fluid Mode Indicates that the pages in this component are to be accessed in fluid mode. This is *required* for fluid rendering.

Layout Only Select when testing the layout of the pages within the component. When selected the borders of all the group boxes on

the page appear in different colors to illustrate the scope of each group box container.

Page Navigation in History Select to record the navigation among the fluid pages for use with the Back button.

Display on Small Form Factor Homepage Select to enable the tile for that component for display on a fluid homepage for a small form factor device.

If Display on Small Form Factor Homepage *is not* selected, the tile for that component *will not appear* on a fluid homepage when accessed by a small form factor device.

If a fluid homepage does not contain *any* tiles for components with Display on Small Form Factor selected, that fluid homepage *will not be displayed* on the small form factor device. The system does not display empty fluid homepages on the small form factor device.

If *none* of the fluid homepages available to a user contain tiles for components with Display on Small Form Factor selected, the system routes the small form factor device for that user to the classic portal homepage.

If you intend to provide access to your PeopleSoft fluid system for small form factor devices, at least one of the fluid homepages to which an end user has access must contain at least one tile for a component with Display on Small Form Factor Homepage selected.

No System Header Select if you have a custom header page that you want to display instead of the default, system header (the fluid banner).

Enable Search Page Select to enable search pages for the component, which, just like classic PIA applications, include the Find an Existing Value page, Add a New Value page, and Keyword Search page.

Header Toolbar Actions

Disable All Actions Select to deselect all header toolbar actions. At runtime the system displays a header page toolbar containing no buttons.

Logout Select for the Logout button to appear on the header toolbar.

Use to logout, or sign off, of the application, returning the user to the fluid login page.

Home Select for the Logout button to appear on the header toolbar.

Returns the user to the default fluid homepage, as if they've just signed on.

Back Select for the Back button to appear on the header toolbar.

	Returns the user to the previous page.
Add To	Select for the Add To button to appear on the header toolbar. Enables a user to add a link to the component to the homepage or the Navbar or Favorites.
Notifications	Select for the Notifications button to appear on the header toolbar. Enabled by default. The Notification button shows the list of notifications for the logged in user, such as events and notifications from AWE. Use this option to disable notifications while within a specific component (exception to rule and also allows debugging). The banner uses this option to determine whether to hide or show the notification button and to enable the appropriate JavaScript.
NavBar	Select for the NavBar button to appear on the header toolbar. **Note:** For the current release, the NavBar is not supported for right-to-left languages, such as Arabic. **Note:** For small form factors, personalizing the NavBar is not supported.
Search	Do not select the Search checkbox. It should not be used, and it will be removed in a future update.
Next in List	Select for the Next in List button to appear on the header toolbar. Takes the user to the next search result on a search page.
Previous in List	Select for the Previous in List button to appear on the header toolbar. Takes the user to the next search result on a search page.

Setting Internet Properties for Fluid Components

If a component is a fluid component, then most of the properties on the Internet tab of the Component Properties dialog box do not apply. For example, none of the Toolbar actions apply.

However, these properties on the Internet tab do apply to Fluid components:

- Primary Action (Search Pages).
- Display Folder Tabs.
- Process Mode options (except for Allow Expert Entry).

For more information on the Internet tab on the Component Properties dialog box, see "Setting Internet Properties" (PeopleTools 8.54: Application Designer Developer's Guide).

Note: In the current release, spell check is not optimized for fluid mode. It is not recommended to enable spell check at the component level or at the field level for fluid applications.

Working with Search Pages

This topic discusses:

- Working with Search Pages for Real Time Component Search
- Working with Search Pages for Oracle SES-based Search
- Comparing Fluid Search Features to Classic Search Features

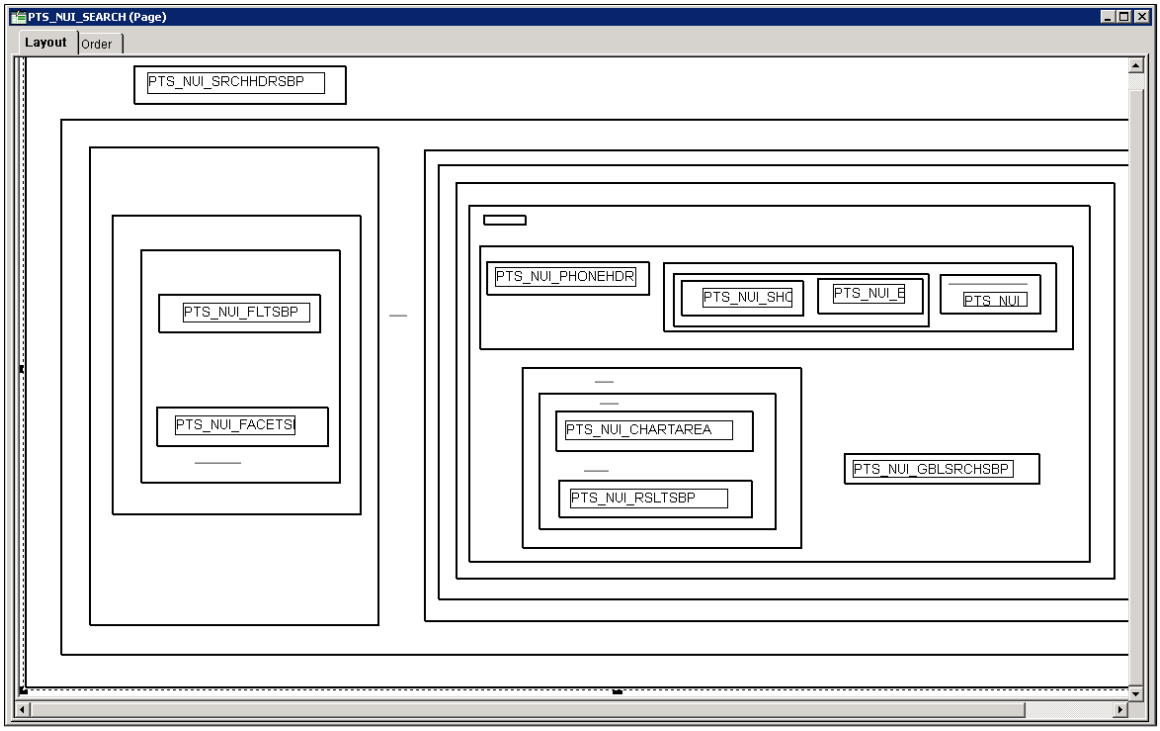
Understanding Search Pages for Fluid Applications

The search pages the system displays for classic PIA applications are not displayed for fluid applications. For each type of search, component search or SES-based search, PeopleTools delivers a default search page. The delivered search page definitions must be the base for any custom or system search page as they contain the basic framework and structures to use as a reference. If you decide to customize the

search page for a specific fluid component, you can clone the delivered search page and add or remove elements as needed.

Image: PTS_NUI_SEARCH

This example illustrates the page design of the default search page, PTS_NUI_SEARCH. It uses the two-panel layout template, and contains numerous group box containers and subpages. If creating a custom seach page based on this template, you can remove the subpage for the feature you don't want to include on your search page. For example, if you don’t want to include charts or facets, remove the PTS_NUI_FACETS and PTS_NUI_CHARTAREA subpages.



Note: The style sheet, PTS_NUI_FFSTYLE, contains the CSS styling used to manage the layout for the delivered search pages.

For any custom search page you create, make sure to select Search Page as the page type on the Page Properties, Use tab.

Search Page Definition	*Description*
PTS_NUI_SEARCH	The fluid equivalent of the classic component search page. Provides full-featured set of search results for using SES-based Search Framework searches as well as a pivot grid-based component search, which resembles the SES based search results. When Enable Search Page is selected, the system automatically loads this search page definition for end users to use for retrieving information.
PTS_NUI_GBLSRCH	Page delivered to display global search results.

For any custom fluid component that you create, Oracle recommends that as part of the normal page and component development, you create:

- a component-based pivot grid (for component search).
- a search definition in the Search Framework for indexing the underlying application data (for SES-based keyword search).

You can set the default search mode for the page using the Primary Action property in the Search Page group box on the Component Properties, Internet tab:

- New: Add mode.
- Search: Real time component search.
- Keyword Search: SES-based search on the search category associated with the component.

Note: If a fluid component is designed to provide both realtime component search and keyword search, then realtime component search should be used as the default search option. That is, the Primary Action setting on Component Properties, Internet tab: should set to *Search.*

Note: In the current version, if you have a component search page used for accessing a component, the Back button does not take the user to the search results page, but rather to the previous component accessed.

Working with Search Pages for Real Time Component Search

In classic PIA, the system automatically builds the component search page displayed to a user, with no additional effort required by the application developer. The system uses built-in logic to create the search page on the fly based on the search keys defined for the component.

In the fluid user interface, the component search page is not built automatically by the system. For any custom fluid component that requires real-time, component search capabilities, the application developer must create a pivot grid, using the component as the data source. The pivot grid generates the underlying query using the search keys defined on the records in the component and populates the search page.

The system can retrieve any default search criteria by using:

- record field defaults set in Application Designer.
- SearchInit PeopleCode.
- prompt defaults set in the pivot grid.

Application Designer Definition	***Definition***
Page	PTS_NUI_SEARCH

If you are creating a new search page or customizing a delivered search page, complete the following task.

To enable real-time component search for a fluid component:

1. On the Component Properties > Fluid tab, enable these properties:

   - Fluid Mode
   - Enable Search Page
   - Next in List
   - Previous in List

   For more information on fluid component properties, see Setting Component Properties for Fluid Components.

2. Clone PTS_NUI_SEARCH, and make any required modifications for your implementation, and make sure you manually insert your search page into the component so the system loads it along with the component.

3. Create a component-based pivot grid for the current component.

   For more information on creating a component-based pivot grid for fluid search, see "Understanding Component Real Time Search Using Pivot Grid in the Fluid Mode" (PeopleTools 8.54: Pivot Grid) and "Creating a Component Pivot Grid Model Using the Pivot Grid Wizard" (PeopleTools 8.54: Pivot Grid).

Working with Search Pages for Oracle SES-based Search

With the SES-based search option, PeopleTools provides an interface tailored to fluid applications, however, the system displays it only if these requirements are met:

- The minimum required browser for fluid access is being used.
- Fluid Mode is enabled for the system.

Working with Global Search

Global search is available if at least one search group has been deployed and indexed. The global search results are displayed in the default search page listed below.

End users invoke a global search by pressing the Global Search icon and entering the search string.

Application Designer Definition	***Definition Name***
Component	PTSF_GBLSRCH_FLUID
Page	PTS_NUI_GBLSRCH

Working with Keyword Search

Each fluid component should have a Search Framework search category mapped to it, deployed to SES and indexed.

When accessing a component with keyword search enabled, the search category mapped to that component appears selected in the Global Search box. Running a search on that category returns results only from the indexes in that category and displays them in PTS_NUI_SEARCH. If the user selects All or

another search category and runs a search, the system treats the search as a global search, and the global search page (PTS_NUI_GBLSRCH) is used to display results.

Application Designer Definition	*Definition Name*
Page	PTS_NUI_SEARCH

Comparing the Availability of Fluid Search Features to Classic Search Features

Due to the differences between classic PIA applications and fluid user interface applications, not every search feature that is available for classic PIA is available for a fluid application, and vice versa. Likewise, within a fluid application, not every search feature is available on every device type. For example, a search option used on a tablet or laptop, may not be available for use on a smartphone.

Feature	*Classic PIA*	*Fluid User Interface*
Persistent Search	Yes	No
Images in results	No	Yes
Related Actions in component search results	No	Yes
Related Actions in keyword search results	Only in List View	Yes
Bulk Related Actions	No	Yes
Conditional Global Search	Yes	Yes
Facets in component search results	No	Yes
List view in component search results	No	Yes
Chart view in component search results	No	Yes
Add a New Value	Yes	No

Feature	*Desktop/Laptop*	*Tablet*	*Phone*
Related Actions	Yes	Yes	No
Bulk Related Actions	Yes	Yes	No
Next in List, Previous in List (Actions Menu)	Yes	Yes	No
Chart interaction in component search	Yes	Yes	No

Note: The features by device table contains only those features where differences occur on one or more devices. For any feature not appearing, it can be assumed that feature is available on all supported devices.

Chapter 5

Adding Page Controls

Managing Common Control Concepts and Settings for Fluid Pages

This topic provides an overview and discusses common page control properties for fluid pages.

Understanding the Makeup of a Control on a Fluid Page

A basic input page control consists of two items:

- The control.
- The label.

The system considers the control and the label pairing as a single unit and presents them within a single container in the HTML—ps_box-<control_type>.

For example, in the case of an edit box, the system generated this structure in the HTML.

Image: HTML for fluid page controls

This example illustrates the appearance of the HTML used to display a page control.

```
<div id="win0divWM_WR_HDR_CONTACT_DETAIL" class="ps_box-edit" >
  <div id="win0divWM_WR_HDR_CONTACT_DETAILlbl" class="ps_box-label" >
    <label id="WM_WR_HDR_CONTACT_DETAIL_LBL" class="ps-label"
    for="WM_WR_HDR_CONTACT_DETAIL" >
      Alternate E-mail address or Phone number for this request only
    </label>
  </div>
  <div id="win0divWM_WR_HDR_CONTACT_DETAILctrl" class="ps_box-control" >
    <input id="WM_WR_HDR_CONTACT_DETAIL" class="ps-edit" type="text"
    title="Alternate E-mail address or Phone number for this request only"
    onclick="javascript:cancelBubble(event);" maxlength="70" value=""
    ></input>
  </div>
</div>
```

The system generates default styles, which are styled through the style sheets. Applications can override or extend style by adding additional styles or replacing the top-level system default style. In the case of the example above, the application change chose to override the PeopleTools-provided style and it would remove the ps_box-edit class *only* in favor of the specified class from the application definition.

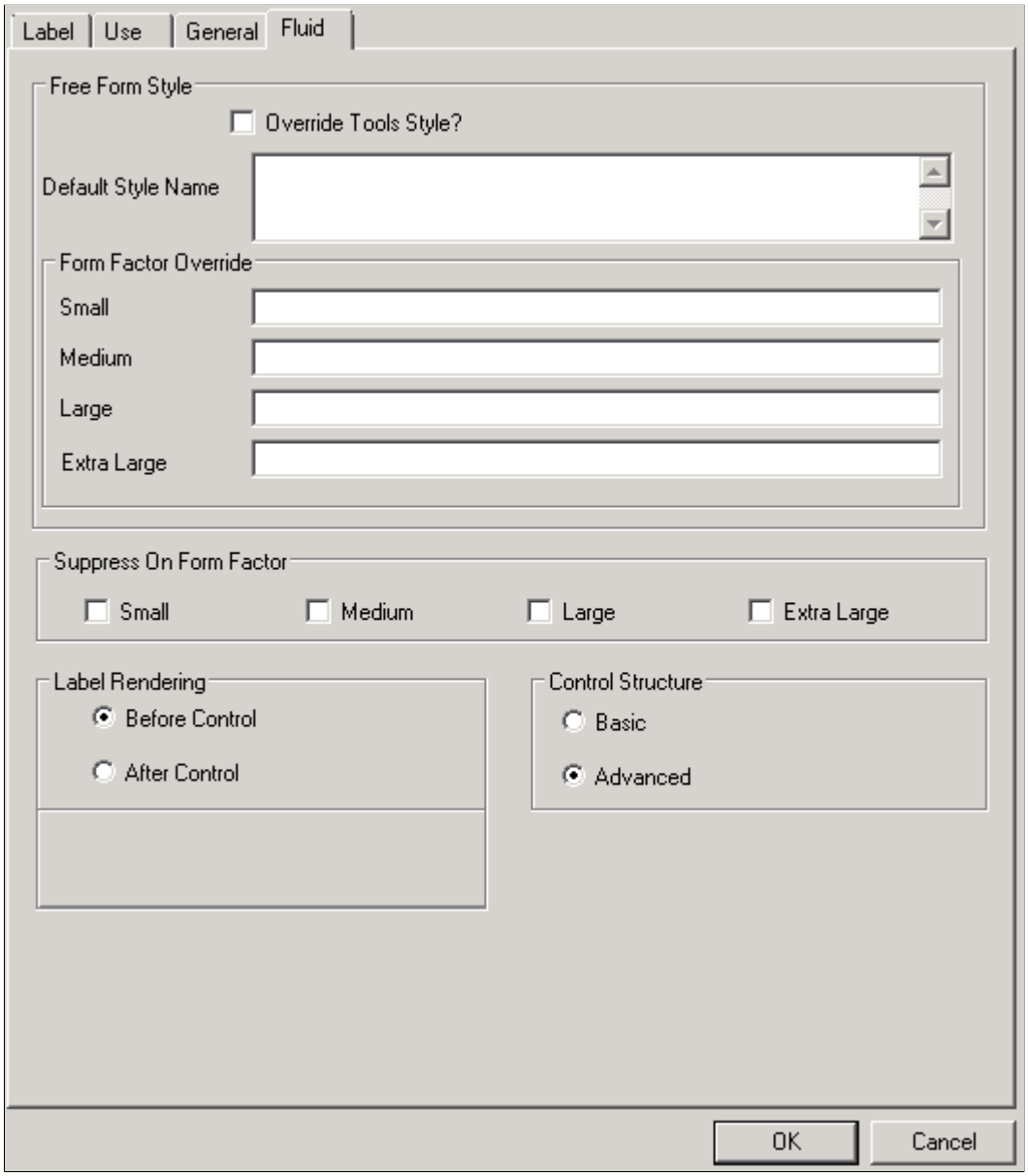
Setting Common Page Control Fluid Properties

For any control that you can add to a page, such as a group box, an image, a grid, and so on, there is a Fluid tab on the properties dialog box for that control. All of the controls share some properties, while others have additional properties that apply only to the function of that control.

Note: The options documented below apply to all fluid controls. Depending on the control selected for the page, additional properties may appear. Properties specific to an individual control type are documented within a section devoted to that control type.

Image: Common Fluid tab Options

This example illustrates the common page control options available for on the Fluid tab of the Properties dialog box for the control. Descriptions of the options follow the example.



Override Tools Style

Indicates to the system *not* to render the default class for that specific control type (as in, the ps_box-edit in the previous example). If this is applied, it applies to *all form factors*.

No PeopleTools default classes for the control are populated (nor does the system load state information).

Default Style Name

Enter additional class specifications used to adjust the display or behavior of a specific control. This setting applies to all form factors unless specifically overridden by subsequent form factor overrides.

Form Factor Override

Applies classes specifically for adjusting the display (and possibly behavior) of an element on a specific form factor. If no value is specified at the form factor override, the default Style Name will be applied.

Suppress on Form Factor

To prevent a page element from being displayed on specific form factor (device size), select the form factor. If selected, the page element is never rendered on the client device, similar to setting a field as invisible for a classic page. Suppression can improve throughput as the size of the HTML document is reduced.

Note: If you select Suppress on Form Factor for a group box, all contents contained within that group box will be suppressed, regardless if the option Hide all Fields when Group Box Hidden is selected on the Grid Properties Use tab.

Label Rendering

The system processes the display of a control as a combined unit of both the display and its label. You can control the position of the label.

Before Control: Select for the label to appear to the right of the control.

After Control: Select for the label to appear to the left of the control.

Control Structure

Basic: Select to have the control appear on the fluid page just as it would appear on a classic PIA page.

Advanced: Select to have the control appear on the fluid page having the look and feel of a typical touch-screen application for a tablet or cell phone. For example, a check box would appear as a sliding switch, rather than an empty square box.

Working with Group Boxes

This topic discusses:

- Using Group Boxes for Fluid Pages
- Setting Group Box Properties
- Working with Layout Group Boxes
- Working with Container Group Boxes
- Working with Accordion Group Boxes
- Creating Custom Headers
- Working with Related Field Group Boxes
- Using a Toggle Container
- Testing Group Box Layout
- Working with Grouplets

Using Group Boxes for Fluid Pages

Group boxes are essential containers used for the layout of the fluid page as well as containers for encapsulating and managing control and widget behavior, positioning, and appearance.

Any fluid page must have at least one over-arching group box to contain all other elements of the page. Depending on the number of controls and complexity of the page, additional group boxes can be added. The group box can be used to indicate visual cues but it can also act as a logical grouping of related controls. Using the fluid properties for a group box, you can override or customize the style at each group box layer.

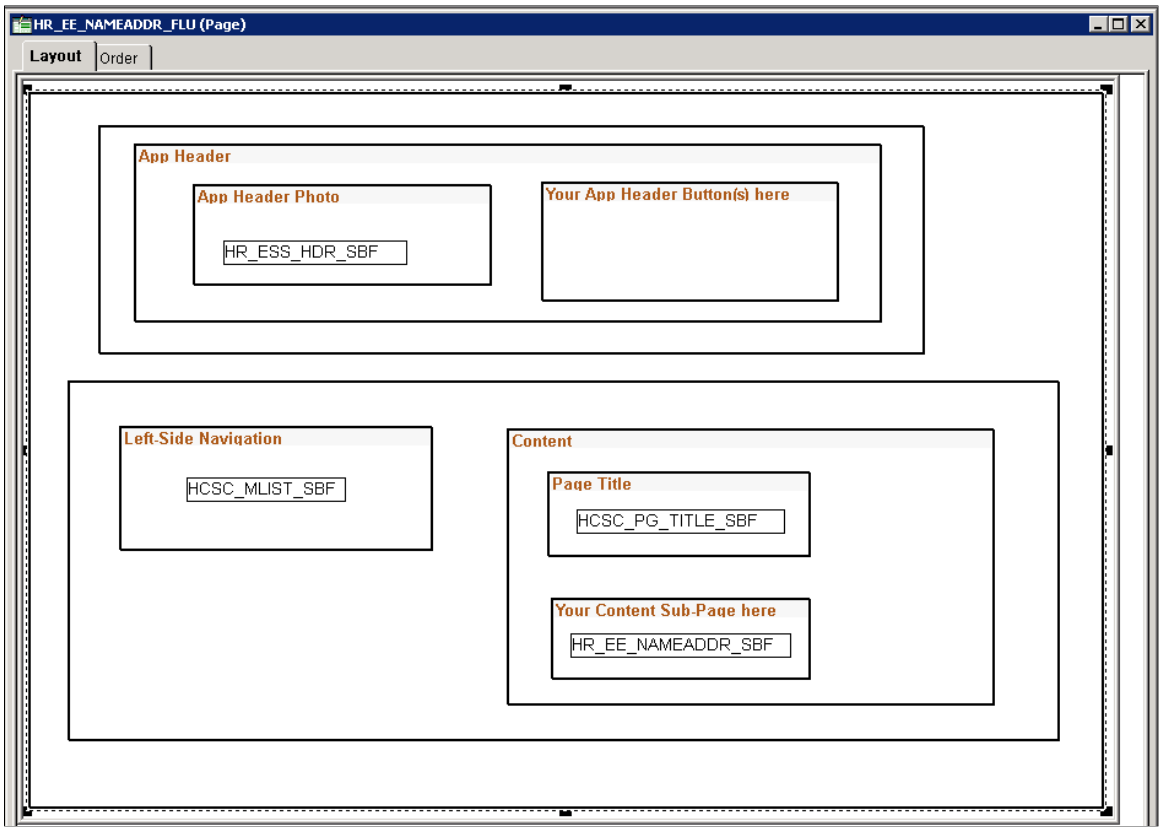
The following example illustrates the use of multiple group boxes in the layout of the fluid page, clearly separating the page elements into distinguishable parts, enabling individual control if needed. For example, the Page Title group box and the Your Content Sub-Page here group box can have individual styles applied if needed.

Every Fluid page should start with a group box. Typically, the outer group box for primary or secondary pages should have the style of ps_apps_content class, though there are exceptions as with 2 panel

display). While the ps_apps_content style would not typically be used in subpages, subpages should start off with a containing group box as well.

Image: Group Boxes define fluid page layout

This example illustrates how group boxes are used in the layout of the fluid page, clearly separating the page elements into distinguishable parts.



In general, it is recommended to be generous in the use of group boxes to layout and organize your page elements. However, group boxes are not just used for layout purposes. For some controls and widgets, it is required that they be encapsulated within a group box in order to apply behavior characteristics.

In addition to using group boxes as a container with a header (the default type of group box), a group box can have additional functions and behaviors. The most common is the usage of a layout type of group box. Layout group boxes are used to enclose elements for specific layout purposes. Some layouts are defined by styles only.

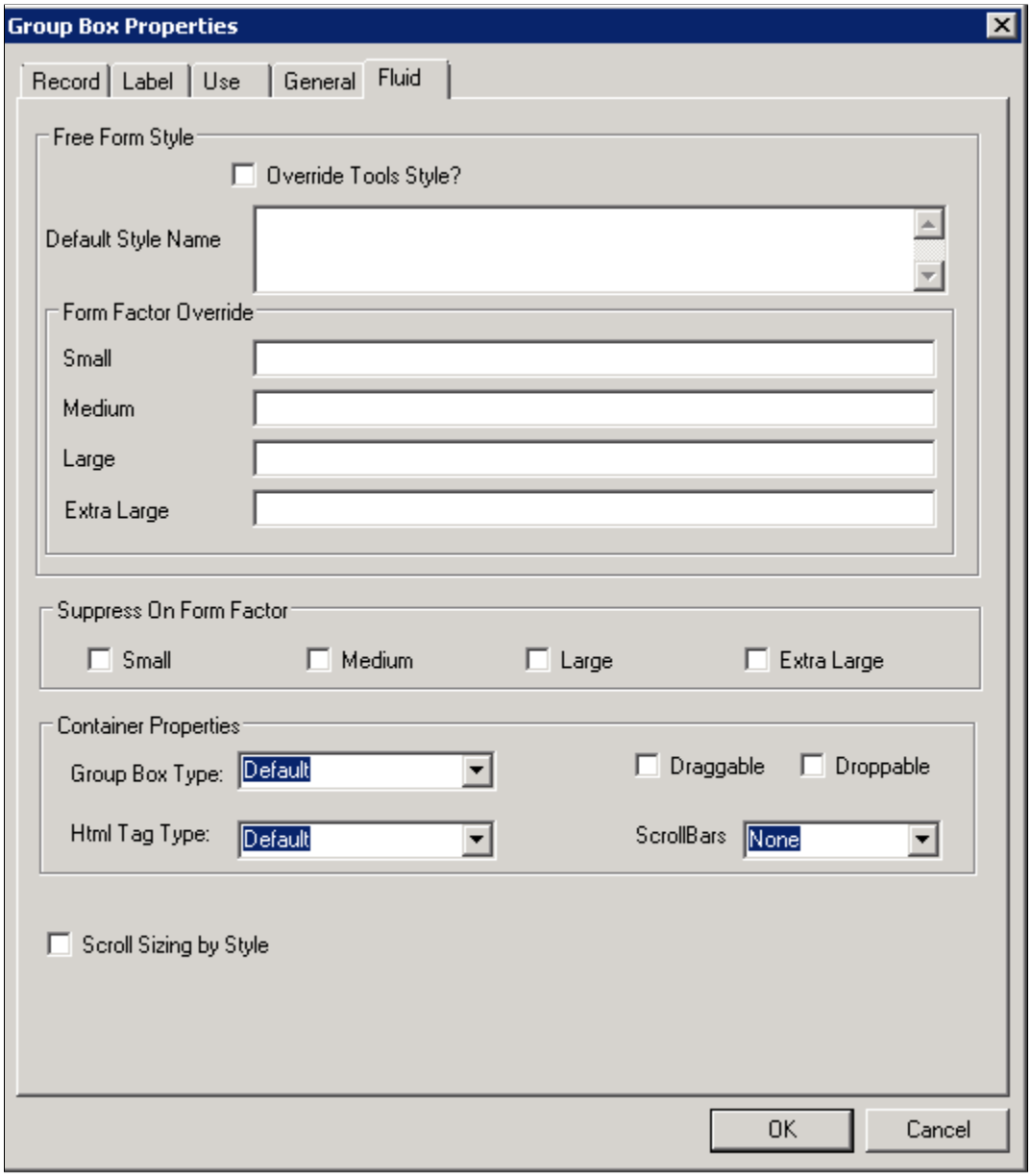
Some group boxes have additional types, like the Accordion Vertical group box which allows for multiple embedded Accordion group boxes to provide the behavior of a mutually exclusive display of group box, with only one of the Accordion Group boxes expanded and visible at any one time. Group boxes can even be put inside of grids to create new virtual columns containing multiple stacked fields.

Group boxes are the most critical element with which you should become familiar for fluid page development.

Setting Group Box Properties

Image: Group Box Properties Dialog Box: Fluid Tab

This example illustrates to properties to set for group boxes uses on fluid pages. Descriptions of the properties follow the example.



Free Form Style and Suppress On Form Factor are common properties to all fluid controls. See Setting Common Page Control Fluid Properties.

Group Box Type

Default: Same as *Container.* It includes both a header structure and a content structure.

Container: Use as a standard container, just as in classic PIA applications, to illustrate groupings of fields.

Accordion Vertical/Horizontal/Group Box: Used to implement expandable sections.

Accordions are a construct where you have a mutually exclusive set of group boxes, only one of which can be expanded at a time (all others collapsed). Accordion Vertical is a container of Accordion Group boxes. The Accordion Group boxes are individual, expandable group boxes within the Accordion Vertical.

Note: Currently, do not use Accordion Horizontal/Vertical option. These options are for internal use only until a future release.

Layout Only: Use for a group box only to contain and position child elements.

Tab Container: Use for creating a side tab.

Note: Side tabs should not be used in the current release.

Note: Currently, do not use the Tab Container option. This option is for internal use only until a future release.

Custom Header Left/Right/Middle: Use to add application controls to header page sections.

Page Title: This is a container for the page title. The header description is generated in a H1 tag (<H1>..<\H1>). It is used by the header page. Related Field Group: Use to group a set of fields.

Related Field Group: Used to create a FIELDSET collection where the legend (the group box header) is repeated to a screen reader when input type elements are read within the FIELDSET. Typical usage would be when you have a collection of radio buttons based on the same field.

Toggle Container: Use to switch between two group boxes and changes the label of the button controlling the toggle.

Note: Currently, do not use the Toggle Container option. This option is for internal use only until a future release.

Custom Grid Header: Similar to the Custom Header sections, the Custom Grid Header group box type allows you to define a custom area to be available for association with a grid. This is used when defining a grid to look for the available Custom Grid Header defined group boxes to associate with a grid. A custom grid header can be associated only with one grid.

Custom Action Menu: Enables the transaction to add to the Action Menu displayed in the banner.

Popup: A type of group box, where the group box label is rendered as a button (a collapsible image can be used to identify an image as well to be displayed with the text). The button is rendered initially and when selected, the content appears as a Popup area associated with the button.

Custom Header Search: Use to create a custom header for component search.

HTML Tag Type

Selecting an HTML tag type enables you to create proper HTML structure as defined by HTML specifications.

Default: Same as DIV.

DIV: Set as an HTML 5.0 DIV tag.

Section: Set as an HTML 5.0 Section tag. If creating a section containing an article, you may consider using this group box type.

UL: Set as an HTML 5.0 UL (unordered list).

OL: Set as an HTML 5.0 UL (ordered list).

LI: Set as an HTML 5.0 UL (list item). Required to be structurally correct. The LI should always be within a UL or OL tag, and nothing else should be directly beneath a UL or OL except for LI.

Draggable

Indicates that content can be dragged out of the group box area. Users initiate drag and drop by using touch on touch screen devices and by using the mouse for desktops and laptops.

Droppable

Indicates that content can be dragged into the group box area.

Scroll Bars

None: Display group box with no scroll bar.

Vertical: Display the group box with a scroll bar on the right side, running top-to-bottom.

Horizontal: Display the group box with a scroll bar on the bottom, running left-to-right.

Both: Display both vertical and horizontal scroll bars.

Carousel: Display the scroll bar in carousel mode (slider mode), where records can be viewed in defined sets by spinning or swiping.

Scroll Sizing by Style

Indicates that the size of the scroll will be controlled by style.

By default, PeopleTools attempts to control the sizing of the group box to accommodate the size of the current page size. If you want to restrict the size of the group box stylistically (using

classes and CSS), you can do this so that the system does not attempt to change the size of the group box as needed.

Working with Layout Group Boxes

The layout group box is used only for layout for positioning its child elements. When using layout group boxes, keep these items in mind:

- No default look and feel style is applied, such as no border.
- No header will be generated.
- The label set for the group box will be ignored at runtime, but can be used at design time for reference.

Working with Container Group Boxes

Container group boxes can be used as a standard container, similar to classic PIA application usage. Use group boxes to associate relevant page field controls and to create a page that is intuitive to the user.

When using container group boxes, keep these items in mind:

- A default, system style provides a standard look and feel.
- A header will be generated if a label is defined.
- Custom styles can be used to override default styles.
- The default HTML tag used for group box is “div,” which you can override with the HTML Tag Type option.
- You can set the group box as Section and Article elements.
- Can be used as a collapsible data area.

Working with Accordion Group Boxes

The Accordion is a web control that allows you to provide multiple, collapsible panes and display them one at a time. With an accordion group box,

When working with the accordion feature:

- Set the parent group box container as Accordion Vertical for vertical accordion behavior (up and down).
- Set the children group boxes (the group boxes that will be the recipient of the expanding and collapsing behavior) as Accordion Group Boxes.

Note: Horizontal Accordion group boxes should not be used in the current release.

Creating Custom Headers

If you do not want to use the provided default system header page, you can use the custom header group boxes to create your own. Before doing so, become familiar with the default system header page definition, PT_HEADERPAGE.

To create a custom header page, create a new page with a Page Type of Header page. You will notice that the header page contains a left box, middle box, right box and a bottom. These correspond directly to the Custom Header Left/Middle/Right/Bottom group box types. Use these group boxes to inject the application controls into the header page sections. At runtime, the content of the group box will replace the content of the custom section in the header page.

Working with Related Field Group Boxes

Use related field group boxes to group or associate a set of fields. The fields will be enclosed in

```
<fieldSet>…</FieldSet>
```

The group header will be within <legend> instead <h..>.

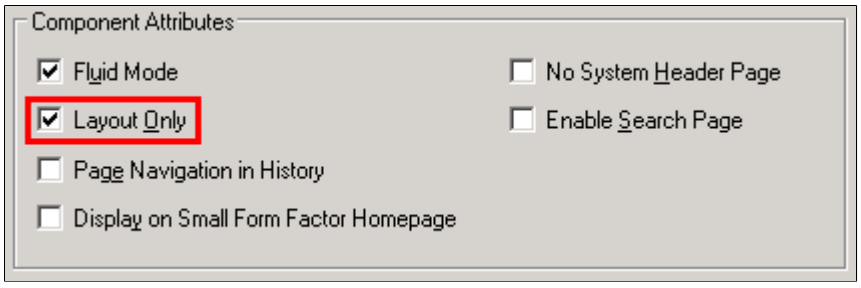
The Collapsible Data Area option should not be used for related field group boxes.

Testing Group Box Layout

Because group boxes are essential to the layout of the fluid page and all of its elements, being able to determine exactly the contents of a group box container is also essential. Select Layout Only on the General tab of the Component Properties dialog box.

Image: Enabling Group Box Layout Testing

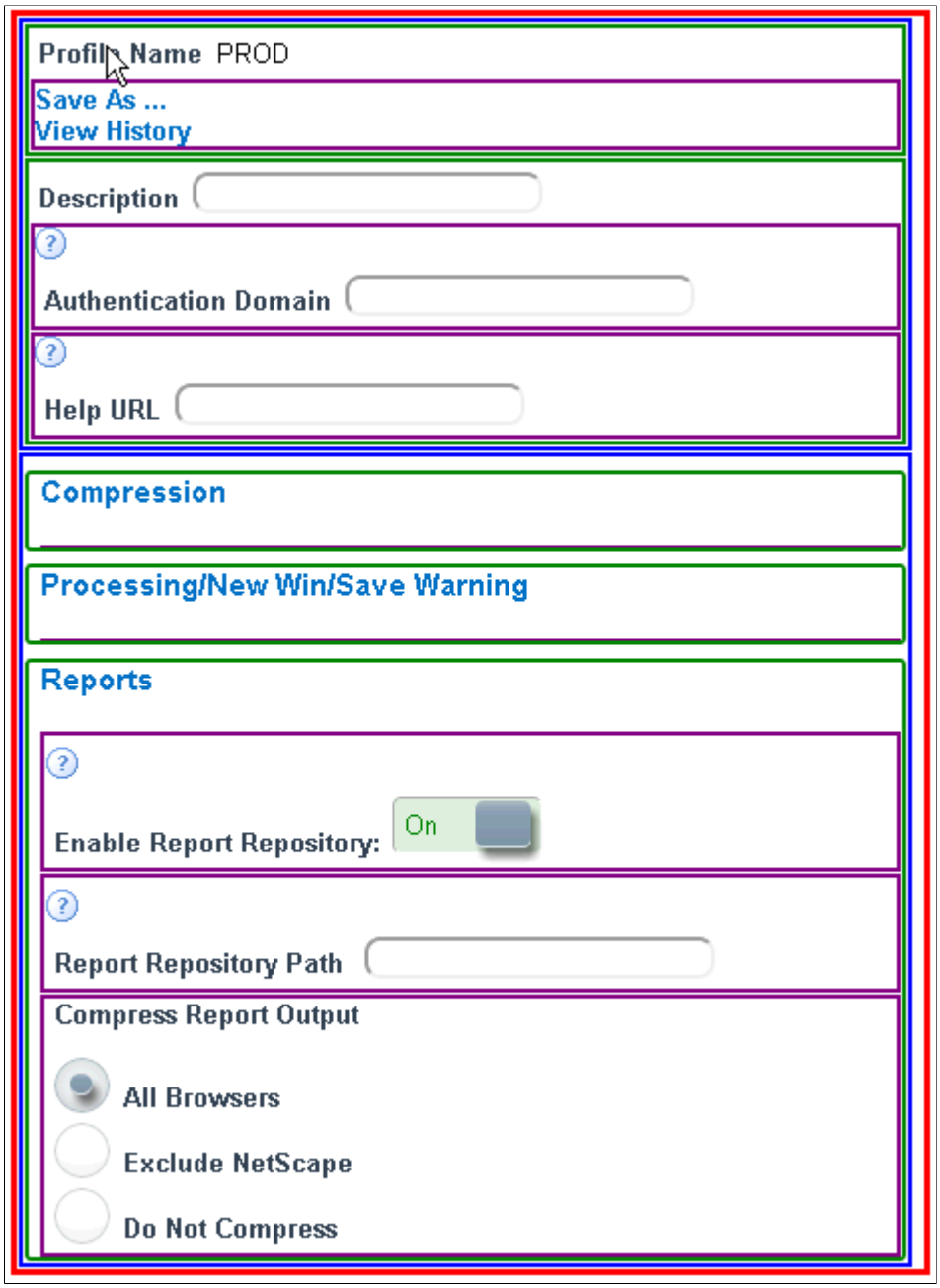
This example illustrates where you enable group box layout testing on the Component Properties dialog box, by selecting *Layout Only.*



At run time, the system displays the fluid page in the browser using different colored boxes to depict the scope of each group box on the page, providing visual indication for you to verify the scope and contents of each group box container.

Image: Group Box Testing Display

This example illustrates the different colored boxes indicating the scope and coverage of each group box.



Working with Grouplets

A grouplet is a group box on a fluid page that is comparable to a pagelet.

These are the grouplet types:

- Consumer Grouplet: The tile is a super container that will consume content from another local/remote component, iScript, or third party content.
- Producer Grouplet: Produces display-only content with no form fields. External URLs, email links and related content actions are supported for hot spots.
- Grouplet/Tile: Can use an event to display more details in a modal, a new window session, or replace the current window to a target group box from an action URL.

Use the Pagelet Wizard to publish the grouplets, similar to how you publish pagelets. Publishing as a grouplet is supported for the HTML data type.

On step 6 of the Pagelet Wizard, you select the Embeddable Pagelet option, and the system displays the Publish as Grouplet link. Click the link to open the Publish Grouplet Definition.

Image: Pagelet Wizard: Publish Grouplet Definition

This example illustrates the fields and controls on the Publish Grouplet Definition page.

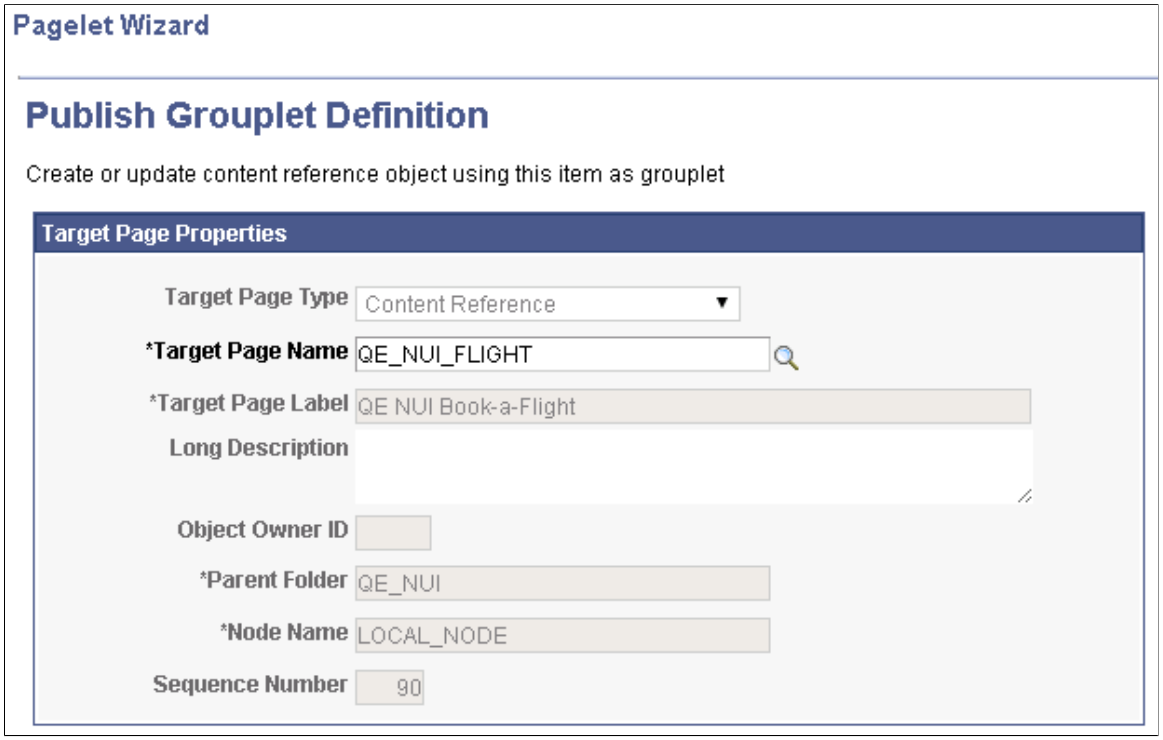


Image: Pagelet Wizard: Publish Grouplet Definition

This example illustrates the fields and controls on the Publish Grouplet Definition page.

Note: Grouplets can be used also in classic PIA to share information between components. If implemented in a classic PIA application, it is recommended to share only lightweight, well-defined information in a grouplet.

Working with Other Page Controls

This section contains an overview and discusses aspects specific to fluid page development with regard to adding page controls.

In general, adding page controls to fluid pages is comparable to adding page controls to a classic PIA page, except for using CSS styles to position the control and the obvious look and feel differences at runtime. The content in the this topic call out key differences with regard to fluid development. You should be well versed in classic PIA page development and all of the various page control options provided by Application Designer prior to beginning a fluid development or customization project.

Working with Grids

Setting Grid Properties

When working with grids, the relevant, common fluid properties apply, but also you can set these properties:

Scroll Sizing by Style Select to override the system default, automatic scroll container sizing. If you use this option, you *must* control the scroll container sizing through a custom style. If you select this option, be aware of these items:

- If the custom style does not control the scroll container size successfully, the scroll may not appear or work as desired.
- Your custom style must be able to handle window size and orientation change.
- If there is no overflow amount (data rows exceeding occurs limit) lazy scrolling may not be triggered.

Custom Grid Header Select to specify a custom grid header.

Note: If you want to add the scrolling feature to a grid (or scroll area), the grid must be placed within a group box.

Include Labels in Grid Cells If using a Tab Separator in your grid, you are allowed to generate field labels within a grid.

Typically, these are suppressed but because we have the ability to put group boxes within grids, the labels are appropriate to display what the individual fields are (depending on context).

Selecting Grid Layout

There are various grid layout options to select.

For more information on grid layout options, see "Setting Grid Use Properties" (PeopleTools 8.54: Application Designer Developer's Guide).

Enabling Lazy Scrolling for Scrollable Grids and Scroll Areas

Lazy scrolling provides the ability to load grid and scroll content as the end user scrolls downward. Rather than loading all content up front, which would degrade performance, the system loads the content as needed. The system loads the next set of scrollable content automatically as the user reaches the end of the currently displayed content.

The system shows the current set of rows displayed in the grid (row count) on the same line as the grid title. As additional rows display due to scrolling, the row count updates automatically.

To enable lazy scrolling:

1. On the Grid Properties > General tab or Scroll Area Properties > General tab, set the Occurs Count to the number rows to be displayed when the grid appears initially.

   Note: Make sure Unlimited Occurs Count (rows) is not selected.

2. In the Header Navigation Bar Properties for that grid, make sure the Invisible option is clear for these property tabs:

   - Previous
   - Next
   - View All

Note: In some situations, such as when displaying a page on a small form factor device, you may want grid scrolling disabled. Apply the style psc_noscroll at the grid level and assign it to Small in the Form Factor Override group box on the Fluid tab in the Grid Properties dialog box to disable scrolling for that grid for small form factors.

Working with Edit Boxes and Long Edit Boxes

When working with edit boxes, the relevant, common fluid properties apply, but also you can set these properties:

Place Holder Text

Display default, place holder, text in the edit box. For example, add a default value or sample format of the field data to assist users.

None: Disables the display of place holder text.

Static: Select to enable the Static Text box, and enter the desired text.

Message Catalog: Select to enable the Message Set/Number boxes. Enter the message set and message number of the message catalog entry that contains the place holder text.

HTML example:

```
<input type="text" id="PS_APP_END_DATE"
class="ps-edit"
placeholder="Specify the date you expect to
finish the assignment."
value="" maxlength="10" ...>
```

Input Properties

For Input Type, select the expected data type for users to input. This sets the HTML input type parameter, to take advantage of device or browser features, such as the numeric keyboard for touch devices for number fields.

Setting input types for forms allow for better input control and validation for various data types, including:

- text
- date
- datetime
- email
- number
- range
- search
- time
- url

Note: The Input Properties setting does not apply to long edit boxes.

Note: In the current release, spell check is not optimized for fluid. It is not recommended to enable spell check at the component level or at the field level for fluid applications.

Working with Number Fields

When you select *number* as the Input Type, the system uses an HTML 5 number input which renders in a fashion specific to the browser/device accessing the page. For example, on a touch device the system displays the numeric keyboard when the number field is in focus, and a browser may display a spinner (spin-button) for a number field.

Note: Browser rendering and HTML 5 rendering may change after the publication of this document.

When setting Input Type to number, consider that HTML 5 restricts the data that can be used as input or displayed for number fields to the following set of characters:

- digits 0 through 9
- "-" for negative numbers
- "." for a decimal point

Adding additional characters (such as a currency symbol, a thousand's separator, and so on) or additional field formatting is not supported for a field with the Input Type set to *number.* For example, if you set Input Type to *number,* the numeric keyboard will be displayed by default for that field on touch devices, however you also introduce the HTML 5 number field character restrictions. In this case, any special characters or formatting applied in Edit Box Properties, field definition formatting, or PeopleCode (such as currency symbols, thousands separators, and so on) will not be displayed or accepted as input for that field.

On the other hand, if you want to allow additional formatting or characters for numerical values, then set the Input Type to *text,* keeping in mind that with the Input Type set to *text* you lose the device/browser-specific representation (such as the numeric keyboard for example).

A general recommendation would be that you should only set a field's Input Type to number if the underlying field definition Field Type is set to *Number* in Application Designer, or if the Field Type is set to *Character,* you have also set the Format Type to *Numbers Only.*

Working with Subpages

Using subpages can help to reduce clutter in the layout page, and it clearly separates related fields into separate entities. In addition, it enables you to logically separate business logic.

When using subpages, you will notice that the footprint is small in terms of the space it consumes in the layout. The width of the subpage displayed on the primary page is based on the width of the label when you insert into the page (fluid page).

Subpages must be enclosed in a group box of the type layout.

There can be multiple levels of subpages.

Working with HTML Areas

In general, HTML Areas should be used sparingly, if at all, with fluid applications. If used, keep these guidelines in mind:

- Select the HTML Tags Only option on the HTML tab of the HTML Area Properties dialog box for performance reasons. This limits the processing and parsing tasks that the system needs to perform, especially in the case of large grids containing numerous input fields.
- Do not use/reference external JavaScript style sheets in the HTML Area.

Working with Check Boxes

With check boxes on a fluid page, there are differences based on the Control Structure settings on the Check Box Properties > Fluid tab.

The default (Advanced) shows the actual translate values (On/Off, Active/Inactive, whatever they happen to be). With Basic selected, the system shows just a check box, similar to standard PIA rendering.

Label positioning can be important with these options, as you typically want labels to appear after a standard check box but before the control with the Advanced option selected.

Image: Advanced and Basic Check Box Rendering

This example illustrates the differences in rendering and label placement of the Advanced and Basic Control Structure settings for check boxes. The Advanced selection shows the toggle switch, while the Basic selection displays a more standard check box.

Working with Push Button/Hyperlink

In addition to the existing push buttons used in classic PIA applications, these push buttons are available for use with fluid user interface pages.

- Home
- Sign Out
- NavBar
- Add To
- Search

You can create toolbars containing push buttons that appear in custom locations, such as sidebars for example, using your own actions, if needed. Push buttons are not required to appear only in the toolbar at the top of the page. Options that can be found on the Use tab that may also be of interest for fluid pages include:

- Execute PC on Row\Group click: Runs PeopleCode as soon as user clicks on row\group).
- Execute SearchSave Event: When a user runs a search, the system saves the page data prior to running the search so it is not lost. This option applies only to the search button, not every button on the page.

Note: For a button to display an image and text (a combination button), on the Label tab, select Both for the Label Type, then specify the Alt tag for image and Label Image.

Working with Static Images and Static Text

Static controls can be used for fluid pages. They have a different structure than what you see in classic PIA, and you can apply style classes to these page elements (to the outer wrapper as with all other controls).

Image: Static Text

This example illustrates the static text control represented in the generated HTML for a fluid page.

```
<div class="ps_box-text" id="win0div$ICField3">
  <span class="ps-text">This is Static Text</span>
</div>
```

Image: Static Image

This example illustrates the static image control represented in the generated HTML for a fluid page.

```
▼<div class="ps_box-staticimg" id="win0div$ICField4">
    <img src="/cs/ps/cache/A_DOWN_1.gif" class="ps-staticimg" alt="Alternate Text for Image" title="Alternate Text for Image">
  </div>
```

Working with Frames

Similar to classic PIA pages, using the frame control is not necessarily recommended, but it can be used a container. Rather than using the frame control, it is recommended to use a group box of type layout instead.

Working with Horizontal Rules

The horizontal rule control can be used and you can apply a style on the Horizontal Rule Properties > Fluid tab to be applied to the horizontal rule. Styles are applied at the ps_box-hr level (as with all controls, styles applied in IDE are found in the outer wrapper of the control).

Applying Styles

Understanding Fluid Styling

The layout of all page elements in a fluid application is controlled by free from style sheets using CSS 3.0 standards. This is one of the most critical differences to consider when comparing classic PIA applications to fluid applications. With each control that you add to a page, PeopleTools *automatically* applies a style class to that control structure, and that style class enables that control to behave and render successfully on a fluid page.

In some cases, a single style class creates a change in behavior, color, or look and feel of a page element. In other cases, depending on the control, a structure of multiple style classes are applied to create the behavior and look and feel of a control. It is critical that before you attempt to override any delivered default styles that you observe the structure of the control in the HTML source of the rendered page so that you can gain insight into the elements involved in the control's structure and the styles that are currently being applied, not to mention the order in which they are applied.

When the system applies a style, by default or override, the style is applied to the container in which the control resides, and those styles apply to all elements in that container, which includes the structure of the control, the label, the field, and so on. In a fluid application, you do not apply a style to discreet elements, such as just to the label for instance.

Identifying PeopleTools Style Sheets

PeopleTools provides a main style sheet for fluid pages and controls, while also supplying additional style sheets for various fluid features, such as the NavBar.

Style Sheet	***Description***
PSSTYLEDEF_FMODE	General style sheet for fluid applications (included on every page), containing the majority of fluid style classes.
PTNUI_NAVBAR_CSS	Contains the styling used for the Navigation Bar (NavBar).
PTPN_NOTIFYWIN_CSS	Notification/Alerts styling. Included on every page where notifications are enabled.
PTNUI_PINTO_CSS	Controls the styling for the “Add to…” functionality. Included on pages with the Add to feature enabled.
PTS_NUI_FFSTYLE	Used by both global search and component search features.
PTPG_MENUSTYLE	Used by pivot grid search features.
PTPG_NUI_GRID_CSS	Used by pivot grid display.

Overriding Delivered Style Classes

The system default style sheet is PSSTYLEDEF_FMODE, which contains all the default, delivered styles for all page elements and controls.

You can create custom free form style sheets in Application Designer, and you can choose to include them and override or extend the default styles at varying levels for page elements.

The system-wide style sheet can be overridden by way of the PeopleTools options page, by selecting the custom style sheet and assigning it to the Default StyleSheet Name option. This is a global, system-wide setting. If set, the system completely ignores any style contained in the delivered style sheet, PSSTYLEDEF_FMODE. Otherwise, if not set, the system applies PSSTYLEDEF_FMODE by default.

Note: Overriding the delivered system-wide style sheet is *not* recommended.

Note: Use PeopleTools, Portal, Branding, Branding System Options to set fluid branding options in the Fluid Component group box.

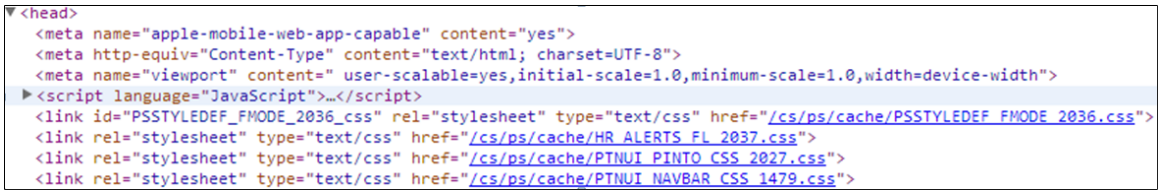
To incorporate additional style sheets at either the component or page level, use AddStyleSheet PeopleCode:

```
AddStyleSheet(StyleSheet.stylesheetname);
```


AddStyleSheet adds the reference to the style sheet to the header section of the HTML page.

Image: Style Sheets in HTML head section

This example illustrates style sheets added into the HTML <head> section.



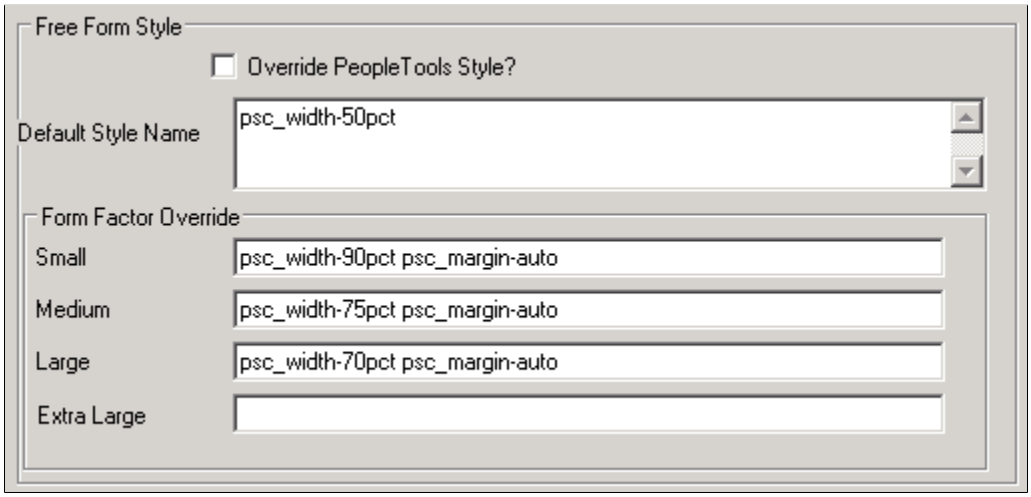
In Application Designer, you can also elect to override or extend the default styles, using the Fluid tab for any page or control use the Free Form Style group box to call out styles to be modified. To completely ignore the default style, select Override PeopleTools Style, otherwise you extend or add to the default style. For situations where you have adjustments to make depending on the form factor of the device, you can set those in the Form Factor Override fields for the various form factors.

When working with the order in which style sheets are applied, consider:

- When you call AddStyleSheet, the system checks if there is a theme with a mapped, skinned style sheet for *that* style sheet. If so, the system loads that style sheet after the style sheet passed in the argument.
- The order in which you call AddStyleSheet determines the order in which the style sheets appear in the stylesheet list.
- The system determines the order in which the PeopleTools style sheets are applied.
- Regardless of custom changes you may make, standard CSS rules *always* apply. It is very important for the developers to understand how CSS and style sheets work.

Image: Form Factor Override

This example illustrates an approach to adjusting the layout of controls based on the form factor of the device accessing the fluid page.



For details on the override options, see Setting Common Page Control Fluid Properties.

Free form styles set in Application Designer are static and can't be changed, however, styles set using PeopleCode are dynamic, meaning they can be switched programmatically at runtime. Styles set in Application Designer can't be changed or queried using PeopleCode at runtime. Dynamic styles are applied *only* with PeopleCode; they can be added, changed, deleted or queried at runtime.

It is not recommended to add style to the system containers. Add your own group boxes, and add style to that.

Example:

```
&fld1.AddFFClass("addffclass");
  &fld1.ReplaceFFClass("addffclass", "replaceffclass");
  &fld1.FreeFormStyleName = "pc_ffclass";
```

In your implementation, Oracle strongly recommends that you only override style classes with respect to coloring, theme, and branding; a practice often referred to as "skinning." Oracle strongly discourages overriding delivered style classes with regard to page layout (positioning, padding, filling, and so on).

You only skin complete style sheets, not the sub style sheets contained within a style sheet. To extend (or skin) a delivered style sheet, you want the delivered style sheet to appear first in the style sheet list, followed by the override or "skinned" style sheet. For example, PSSTYLEDEF_FMODE should appear before PSSTYLEDEF_FMODE_skinned, allowing the original to be applied, and the "skin," containing the supplemental changes, to be applied after.

Note: With CSS style sheet application, the last style sheet in the sequence to be applied takes precedence.

It is recommended to skin or extend only one style sheet at a time. As in, if you have multiple style sheets, have the skinned style sheet appear just after (under) the style sheet it extends or overrides. If the extended or override style sheet appears beneath other style sheets in the list (assuming there are multiple style sheets), any override in the skinned style sheet will apply to any style sheet specified before it, which may result with unintended style overrides.

If you intend on extending or overriding the delivered styles, consider these items:

- You need to be aware of all the styles applied in any previously applied style sheet(s).
- If overriding styles, it is recommended to change only color characteristics.
- It is not recommended to override any styles related to structural (layout or positioning) characteristics or behaviors applied to page elements.
- Do not modify a delivered style sheet. Any changes will be overridden in any future upgrade.
- Styles are associated directly to structure (container, label, control, and so on), and any change PeopleSoft makes to structure in future releases can affect any custom styles you've implemented.

Setting Min/Max Width on Content

Not all content displays as desired when stretched to a maximum size of a device or reduced (shrunk) too much to fit a smaller space, or device. There are functional minimum and maximums, which should be defined on the content so that scrolling occurs or the size of the area no longer grows in terms of min/max values for width. When creating custom fluid pages, or customizing delivered pages, it is your responsibility to manage the styling for these min/max values.

It is recommended that you *do not* set the min/max values directly on the group box with the style ps_apps_content, as this can cause browser issues.

The ps_apps_content style implements the flex model. Pages are laid out as a column based on flex model with horizontal stretching. This can cause varying results per browser type when attempting to stretch the width fully while restricting the width based on a max-width value.

It is recommended to include a layout group box within the ps_apps_content group box containing your page content, and apply the max/min widths on the interior group box.

Layout Element	***Example***

Structure and Style Class	`GROUPBOX – class=ps_apps_content`
	`  GROUPBOX – class=psa_my_content`
	`  <page content goes here>`

Style	`.psa_my_content {`
	`  min-width:320px;`
	`  max-width: 1440px;`
	`  margin:0 auto;`
	`/* center content horizontally */`
	`}`

Hiding Columns in a Flex Grid Using Style

You can style an entire column in a flex grid by using the page field name of the field (or group box) within a grid. The page field name will be applied directly to the column (both the table header [TH tag] and the cells [TD tag]) as a style. At that point, you can affect styling an entire column (centering, font weight, and so on).

You can hide an entire column stylistically, using the page field name (as opposed to hiding the column using PeopleCode, which can prohibit complete rendering of the column).

If you are hiding a column using style, you must consider the following items.

Consideration	***Description***

Accessibility	If you stylistically hide the trigger column used for the row action, you must make sure that you un-hide the trigger column in Accessibility Mode.
	
	This is because in accessibility mode, the row action is disabled and the column used as the trigger field now becomes actionable and can receive focus. In accessibility mode, the row action trigger should have the following characteristics:
	
	• It must be visible.
	
	• It must have text (alt text or image) that explains the action.
	
	For example, if you have a field on a grid that has a page field name of “ABC_XYZ”, the style applied on all references to that column is “ABC_XYZ” (TD and TH tags in the table). To properly hide the entire column, use the following:
	
	`:root:not(.psc_mode-access) .ABC_XYZ`
	`{ display:none; }`

Consideration	*Description*
Grid Header Alignment	If you stylistically hide a column, *do not* stylistically hide the last column in a grid. Doing so can adversely affect grid header layout. PeopleTools uses the last column of the grid to adjust the grid header if it needs to have an overhang for the scroll bar, enabling the grid scroll bar to appear as part of the grid. If you hide the last column, the overhang is assigned to the last column and if it is hidden, the overhang can be lost, resulting in a misaligned grid header.

Working with Flex Grid Column Alignment

A flex grid consists of two table structures that support both the fixed grid header and a scrollable grid body. The column styes are applied to both table constructs that make up this page element. However, these styles can have adverse effects on the grid header, causing alignment issues if you attempt to adjust column dimensions.

If you are attempting to set any dimensions on columns, make sure you are familiar with the limitations of using min/max values on grid columns.

If you attempt to set a dimension of any type (width, min-width, max-width, and so on), make sure you affect only the real table and not the header-only table, which the system uses for header alignment.

For example, assume a column named XYZ_ABC as a page field name in the grid. That class is assigned to all “TD” and “TH” tags both in the real table (ps_grid-flex) as well as the table used internally for fixed header implementation (ps_grid-flex-head). To style a dimension and change the text color, see the following:

```
.ps_grid-flex .XYZ_ABC { min-width:10em; } /* set dimension only on real table */
.XYZ_ABC {color: #A00;} /* set color both real and header-only table */
```

Working with JavaScript

This topic discusses:

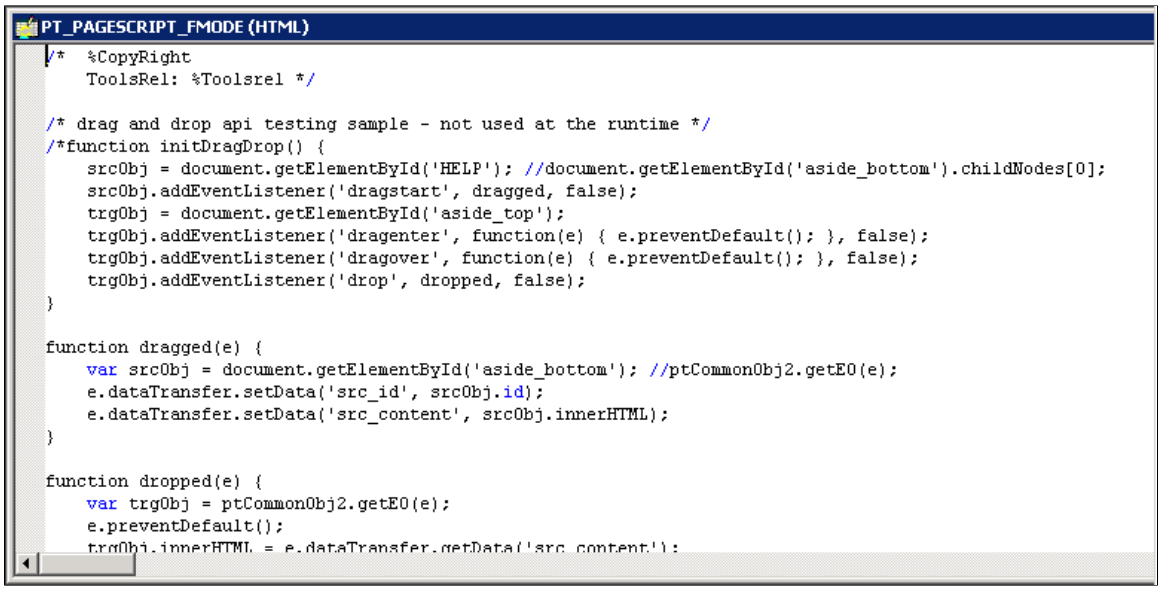
- Delivered JavaScript.
- Custom JavaScript guidelines.

Delivered JavaScript.

PeopleTools delivers an HTML definition, PT_PAGESCRIPT_FMODE, containing all the provided JavaScript functions.

Image: PT_PAGESCRIPT_FMODE

This example illustrates the delivered JavaScript in PT_PAGESCRIPT_FMODE.



These JavaScript functions control a wide range of page behavior, including (but not limited to):

- Scrolling for group boxes
- Accordion widget
- Toggle widget
- Grouplet (like a pagelet – smaller)
- Drag and drop group boxes
- Grid lazy scroll
- Carousel – (used only for scroll areas)
- Transfer animation
- File attachment
- Search page and nav bar
- Related action menu
- Side page tabs
- New action buttons – Back, home, logout
- Search side page

Custom JavaScript Guidelines

Incorporating custom JavaScript programs is supported, but application developers and implementation teams should use custom JavaScript programs judiciously, and they should be well aware of any implications introduced by the custom code.

With the architecture of the PeopleSoft Fluid User Interface, PeopleTools provides increased flexibility and opportunity to customize. However, with that flexibility comes increased responsibility for developers to test and manage any custom code. PeopleTools can only impose logical restrictions and processing constraints on the code it delivers and tests.

When considering adding custom JavaScript programs, consider the following items:

- Include JavaScript files in component PeopleCode, such as PostBuild, for better performance. Otherwise, the system needs to perform extra processing, such as inserting references into the header and checking to make sure all of the styles have been downloaded and applied, and before completing the page load\activate process.
- Avoid using custom JavaScript events on a control.
- If there are any JavaScript errors from app onload, the page will also encounter problems loading. • Do not use JavaScript to change the structure of the document.
- Don’t use JavaScript to add/change the free form styles on the controls. Use PeopleCode to change/add free form style.
- Only add JavaScript events on the controls that do not have any system generated script.

Considering Accessibility for Fluid Pages

When building any application, developers need to consider accessibility requirements always. This topic describes some key concepts to keep in mind with regard to accessibility while developing fluid pages.

For more information on accessibility for PeopleSoft applications in general, see"Accessibility in PeopleSoft Applications" (PeopleTools 8.54: Accessibility Guide).

Accessibility Consideration	***Description***
Creating Custom Controls	If you create a custom control using HTML Area, which is discouraged, the "Name, Role, Value" rule has to be in place. For example: • Custom controls need a meaningful programmatically exposed name. • Custom controls have appropriate roles specified using the role attribute. • Custom controls have an appropriate value or state and property information conveyed using the relevant WAI-ARIA or HTML 5 attributes.

Accessibility Consideration	*Description*
Providing Indication of Focus	An end user must be able to see where focus is for the control, for example a box surrounding the widget. • if the widget is in HTML Area, provide your own focus indicator. • if you style a delivered widget in a unique way so it no longer looks like others of its kind, you need to make sure to style a clear focus indicator.
Navigating HTML Area	The controls in an HTML area must be able to be operated using the keyboard only, with no reliance on the mouse. Use `tabindex="0"` for controls that are not accessible using TAB, but should be (those that can receive user input).
ARIA-live region	Use are-live regions sparingly.
Invisible fields	When fields are not visible (for example, within closed accordion content) they should not be tab-able.
Lists	If you have a list, use list grid layout to render it, because it is a list, not a table, you are presenting. When you have only one column, it is not a table, it is a list. If you want to show how many items are in the list, do not use the word "rows" for the counter. *Row* implies tables, which is confusing for screen reader users.
Table Summaries	Provide table summaries for table. Use Application Designer to populate them. Use the Summary attribute for Accessibility property on the Grid Properties > Use tab.
Links versus Buttons	If a link doesn't transfer users to a different location when they press or tap it, it is strongly recommended that you use a button, rather than a link.
Foreground/Background colors	If you specify a foreground color, it is strongly recommended that you also specify a background color and vice-versa. Otherwise a user changing their system colors may result in invisible text. Also, be cautious when using any of the following CSS techniques: • Background Images • Gradients • RGBA (Transparent or semi-transparent) colors These all make it difficult to test for color contrast, if not applied appropriately.
Background images	Do not use background images for "information" icons. They are not visible in high contrast mode. Only do this for "decorative" images.

Accessibility Consideration	*Description*
Dynamic information	User should be able to stop/pause/hide information that is moving, blinking, or auto-updated if: • it starts automatically • it lasts more than five seconds
Instructions displayed on pages	Avoid instructions with sensory characteristics/descriptions of page elements, such as: • color (red, blue, and so on) • visual location (above, below, and so on) • orientation (right, left) • size and shape (small, larger, round, arrow, and so on). Use field identification instead to point users to page elements.
Image/Control consistency	Images and controls are used and identified consistently throughout. For example, it is not recommended to include a button labeled "Search" on one page and one button labeled "Find" on another page of the same application.
Plug-ins	If you use plug-ins, verify content rendered with those plug-ins is accessible.
Tab order	Pay attention to focus order by tabbing to make sure tab order is in logical sequence. Pay close attention to: Focus location after refreshing part of a page Focus location in and out of popups
Accessibility mode	If your fluid component and classic PIA component are exactly the same in functionality (only styling are different), you may want to transfer to the classic PIA component when a user is in fluid accessibility mode. This can eliminate the need to design fluid pages for regular and accessibility mode. Also, in accessibility mode, add the word "Selected" for the selected list. `:root:not(.psc_mode-access) .GRID1_EDIT, :root:not(.psc_mode-access) .GRID1_EDIT { display:none !important }` If you hide it always, there will be no action available on the row in accessibility mode
MENU	Do not use the word "MENU" in a title for a groupbox-popup. Otherwise, a screen reader user will expect a keyboard that works with MENU to be available (UP/DOWN/LEFT/RIGHT arrow keys), which is not available yet.

Accessibility Consideration	*Description*
Button/Hyperlink Row Action	If you have a button/hyperlink and the control is used for Row Action, you must supply a translatable label for the label whether it's visible/hidden or not. This label is needed for screen reader users.
Row Header	Fluid pages should specify a row header for tables, specified in Application Designer. Use the Row Header property on the Grid Properties > Use tab.
Testing	Contrast: Test your page in Windows High Contrast Mode. Screen Reader: If you build your application for tablets or phone, test using a native screen reader. For example, use VoiceOver (OSX native screen reader).
Toggle buttons	If you have a toggle button with different visual style when it is ON/OFF, you should do one of the following: • change the label of the button when ON/OFF. • code wai-aria states aria-pressed="true/false".
Read-only group box content	Groupboxes that contain read only elements and have row action, it is recommended to add an aria-label and set the role using PeopleCode `GROUPBOXFIELD.JavascriptEvents = "aria-label=""Partial Days None""role=""button""";`
Tab order	Unlike classic PIA pages, with fluid UI pages the tab order cannot be set by the developer. The tab order is based strictly on top-to-bottom, left-to-right sequence. For fluid pages, design in such a way that logical tab order is maintained as much as possible, leveraging group boxes to group page elements as necessary.
Drag and Drop	If you are leveraging drag and drop capabilities on your page, you must provide an accessible mechanism that sighted users can use. A similar mechanism must be available. For example, with homepage personalization, users can drag and drop tiles into their own specific layout and order. For sighted users, who cannot use drag and drop, a Personalize Homepage interface is available where users can also use the keyboard to arrange pagelets.

Chapter 6

Working with PeopleCode for the Fluid User Interface

Working with Fluid Page and Component PeopleCode

This topic discusses PeopleCode constructs and usage specific to working with fluid pages and fluid components.

PeopleCode Processing Order

The PeopleCode processing order for Page Activate is:

1. Header
2. Side
3. Footer
4. Main

Page Type PeopleCode

PeopleCode for page types of Header, Side and Footer:

`Page.Visible`	Dynamically control page visibility.
`GetPageType`	See "GetPageType" (PeopleTools 8.54: PeopleCode Language Reference)..

Using Fluid Component Level PeopleCode

The following table contains common component level PeopleCode used in fluid applications.

`IsSaveEnabled()`	See "IsSaveEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
`IsLogoutEnabled()`	See "IsLogoutEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
`IsHomeEnabled()`	See "IsHomeEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
`IsBackEnabled()`	See "IsBackEnabled" (PeopleTools 8.54: PeopleCode Language Reference).

IsPinEnabled()	See "IsPinEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsNotifyEnabled()	See "IsNotifyEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsNavBarEnabled()	See "IsNavBarEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsNextInListEnabled()	See "IsNextInListEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsPrevInListEnabled()	See "IsPrevInListEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsSearchEnabled()	See "IsSearchEnabled" (PeopleTools 8.54: PeopleCode Language Reference).
IsGroupletRequest()	See "IsGroupletRequest" (PeopleTools 8.54: PeopleCode Language Reference).
IsFluidMode()	See "IsFluidMode" (PeopleTools 8.54: PeopleCode Language Reference).
IsFluidStartSearch()	See "IsFluidSearchStart" (PeopleTools 8.54: PeopleCode Language Reference).
GetPageTitle()	See "GetPageTitle" (PeopleTools 8.54: PeopleCode Language Reference).

Working with Grids

When working with girds you can use these constructs:

See "Layout" (PeopleTools 8.54: PeopleCode API Reference).

See "Grid Class Properties" (PeopleTools 8.54: PeopleCode API Reference).

Meta HTML

Additional HTML you can use with fluid development include these constructs.

See "%ExplainMessage" (PeopleTools 8.54: PeopleCode Language Reference).

See "%FORMFACTOREXTRALARGE" (PeopleTools 8.54: PeopleCode Language Reference).

See "%FORMFACTORLARGE" (PeopleTools 8.54: PeopleCode Language Reference).

See "%FORMFACTORMEDIUM" (PeopleTools 8.54: PeopleCode Language Reference).

See "%FORMFACTORSMALL" (PeopleTools 8.54: PeopleCode Language Reference).

See "%BB" (PeopleTools 8.54: PeopleCode Language Reference).

See "%BP" (PeopleTools 8.54: PeopleCode Language Reference).

See "%BV" (PeopleTools 8.54: PeopleCode Language Reference).

Additional PeopleCode Constructs

This section lists various other PeopleCode constructs used in fluid development.

See "AddJavaScript" (PeopleTools 8.54: PeopleCode Language Reference).

See "AddMetaTag" (PeopleTools 8.54: PeopleCode Language Reference).

See "AddOnLoadScript " (PeopleTools 8.54: PeopleCode Language Reference).

See "CreateSearchRowset" (PeopleTools 8.54: PeopleCode Language Reference).

See "ExecuteSearchSavePC" (PeopleTools 8.54: PeopleCode Language Reference).

See "GetAddSearchRecName" (PeopleTools 8.54: PeopleCode Language Reference).

See "GetPageType" (PeopleTools 8.54: PeopleCode Language Reference).

See "GetSearchRecName" (PeopleTools 8.54: PeopleCode Language Reference).

See "IsAddMode" (PeopleTools 8.54: PeopleCode Language Reference).

See "SetAddMode" (PeopleTools 8.54: PeopleCode Language Reference).

See ."SetThemeId" (PeopleTools 8.54: PeopleCode Language Reference).

See "HtmlAttributes" (PeopleTools 8.54: PeopleCode API Reference).

See "HtmlInputType" (PeopleTools 8.54: PeopleCode API Reference).

See "IsHyperlink" (PeopleTools 8.54: PeopleCode API Reference).

See "JavaScriptEvents" (PeopleTools 8.54: PeopleCode API Reference).

See "PlaceholderText" (PeopleTools 8.54: PeopleCode API Reference).

Determining Browser and Device Type

Depending on the device or browser accessing the application, you may want to set various options or styles catered to that device type. At runtime, the HTML page-level class is determined by the device type, browser type, and browser platform (operating system).

See ."Request Class Methods" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDeviceInfo" (PeopleTools 8.54: PeopleCode API Reference).

See "BrowserDeviceType" (PeopleTools 8.54: PeopleCode API Reference).

See "BrowserDeviceFormFactor" (PeopleTools 8.54: PeopleCode API Reference).

See "BrowserFluidCapable" (PeopleTools 8.54: PeopleCode API Reference).

See "BrowserTypeClass" (PeopleTools 8.54: PeopleCode API Reference).

See "BrowserPlatformClass" (PeopleTools 8.54: PeopleCode API Reference).

See "ExtraLarge" (PeopleTools 8.54: PeopleCode API Reference).

See "Large" (PeopleTools 8.54: PeopleCode API Reference).

See "Medium" (PeopleTools 8.54: PeopleCode API Reference).

See "Small" (PeopleTools 8.54: PeopleCode API Reference).

Using Drag and Drop PeopleCode

The PeopleCode used to handle drag and drop functionality needs to be component-level PeopleCode and to be added to the PT_WORK.PT_BUTTON_DND field. With drag and drop, you drag data from a field, row, or rowset in a source group box and drop it onto a field, row, or rowset in a target group box.

See "GetDNDField" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDNDRow" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDNDRowset" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDNDTargetField" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDNDTargetRow" (PeopleTools 8.54: PeopleCode API Reference).

See "GetDNDTargetRowset" (PeopleTools 8.54: PeopleCode API Reference).

See "Draggable" (PeopleTools 8.54: PeopleCode API Reference).

See "Droppable" (PeopleTools 8.54: PeopleCode API Reference).

Working with Grouplet PeopleCode

This topic presents PeopleCode used with grouplets.

See "SetGroupletActionUrl" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletDisplayIn" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletID" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletImage" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletMOptions" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletTargetID" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletTimer" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletType" (PeopleTools 8.54: PeopleCode API Reference).

See "SetGroupletUrl" (PeopleTools 8.54: PeopleCode API Reference).

Working With DoModal Popup PeopleCode

When working with modal windows you can use these PeopleCode constructs.

See "DoModalComponentPopup" (PeopleTools 8.54: PeopleCode Language Reference).

See "DoModalPopup" (PeopleTools 8.54: PeopleCode Language Reference).

Working with the Back Button

The Back button enables you to configure the system to track user history and enable the user to retrace steps as needed. Including the Back button is an option for the system header page.

See "SetTransferAttributes" (PeopleTools 8.54: PeopleCode Language Reference).

Managing Styles

When managing styles with PeopleCode use these constructs.

See "AddFFClass" (PeopleTools 8.54: PeopleCode API Reference).

See"ReplaceFFClass" (PeopleTools 8.54: PeopleCode API Reference).

See "FreeFormStyleName" (PeopleTools 8.54: PeopleCode API Reference).

See "AddStyleSheet" (PeopleTools 8.54: PeopleCode Language Reference).

Chapter 7

Managing Fluid User Interface Deployments

Configuring Fluid Mode Web Profile Settings

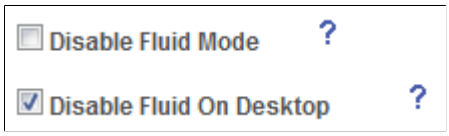
Web Profile Configuration - General Page

Navigation

Access the Web Profile Configuration page. (Select PeopleTools, Web Profile, Web Profile Configuration.)

Image: Web Profile Configuration - General page, fluid mode properties

The following example illustrates the web profile properties you can set for fluid applications.



Disable Fluid Mode

Setting this option will disable all fluid pages and content references. Only the classic equivalent content references and links will be available throughout the entire system. This option allows system administrators to force classic-only mode for all users.

This option is not selected (disabled) by default.

Note: This option will disable all fluid content references. Applications delivered only as fluid components will not be accessible with this option enabled.

Disable Fluid on Desktop

Setting this option will display only classic pages on desktops and laptops.

By default this option is selected (enabled), which means that by default, browsers running on desktop computers display classic PIA pages, not the fluid equivalents.

Note: This option is effective only when the Disable Fluid Mode is not set.

PeopleSoft Fluid User Interface is enabled for the entire system by default for your initial PeopleTools installation, but it is disabled for desktop computers. Specifically, when you install your PeopleTools System Database, by default, the Disable Fluid Mode web profile setting is not selected, but the Disable Fluid On Desktop setting is selected.

The intention is for mobile devices, such as smartphones and tablets, to access the fluid user experience by default, while conventional desktop computers access the classic user experience by default. The classic user experience is what previous PeopleSoft application users on desktops are accustomed to seeing when they access a PeopleSoft system.

You can enable fluid mode for desktop computers if desired, depending on your user base and the fluid pages and applications provided by your PeopleSoft application.

For information on all of the fields on the Web Profile Configuration - General page, see "Configuring General Portal Properties" (PeopleTools 8.54: Portal Technology).

Web Profile Configuration - Look and Feel Page

Navigation

Access the Web Profile Configuration - Look and Feel page. (Select PeopleTools, Web Profile, Web Profile Configuration. Select the Look and Feel page.)

Image: Web Profile Configuration - Look and Feel page, fluid mode properties

The following example illustrates the Fluid Homepage property that you can set for fluid applications.

Fluid Homepage Displays the name of the component used to render fluid homepage tabs.

The default value is *NUI_FRAMEWORK.PT_LANDINGPAGE.GBL*.

Override Click to change the default fluid homepage. The Override Fluid Homepage Component page is displayed with these fields:

- Menu Name: The default value is *NUI_FRAMEWORK*.
- Component: The default value is *PT_LANDINGPAGE*.
- Market: The default value is *GBL*.
- Fluid Homepage: This field is blank by default.

  Important! When this value is set, it appears as `?LP=LB_PAGE_ID` and it overrides any values set on the Assign Branding Themes page. See "Assigning Branding Themes" (PeopleTools 8.54: Portal Technology) for more information on defining the default fluid homepage for a portal and by user attribute.

The values of these four fields are concatenated to produce the value of the Fluid Homepage field on the Look and Feel page.

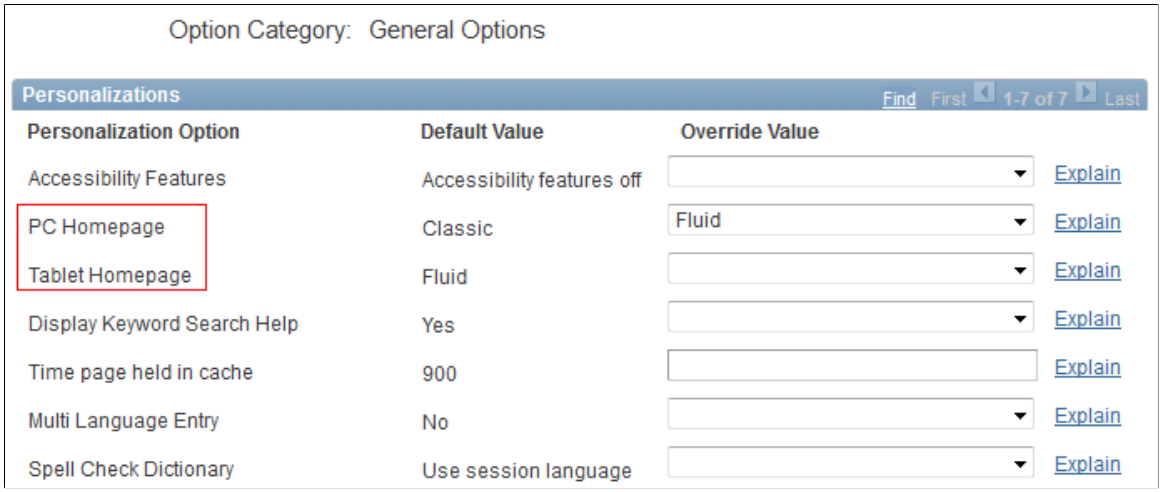
For information on all of the fields on the Web Profile Configuration - Look and Feel page, see "Configuring Appearance and Character" (PeopleTools 8.54: Portal Technology).

Setting Fluid Homepage Personalizations

When deploying fluid user interface applications to your user base, consider the devices being used to access your system, and if any default personalization options need to be overridden.

Image: Fluid Homepage Personalizations

This example illustrates the personalization options provided for users to set their default homepage when signing on to the system.



PC Homepage For users signing on from a PC or laptop, the default value is set to *Classic,* meaning the system displays the classic portal homepage. If fluid user interface is enabled for your system, and you want PC and laptop users to access the fluid homepage, set the Override Value to *Fluid.*

Tablet Homepage For users signing on from a tablet or smartphone, the default value is set to *Fluid,* meaning the system displays the fluid homepage to users on these devices.

Note: Regardless of the device being used for access, the system does not display the fluid homepage if the minimum browser version for fluid access is not being used on the device. Refer to the PeopleTools browser support information on My Oracle Support > Certifications.

Configuring the NavBar to Appear in the Classic Portal Header

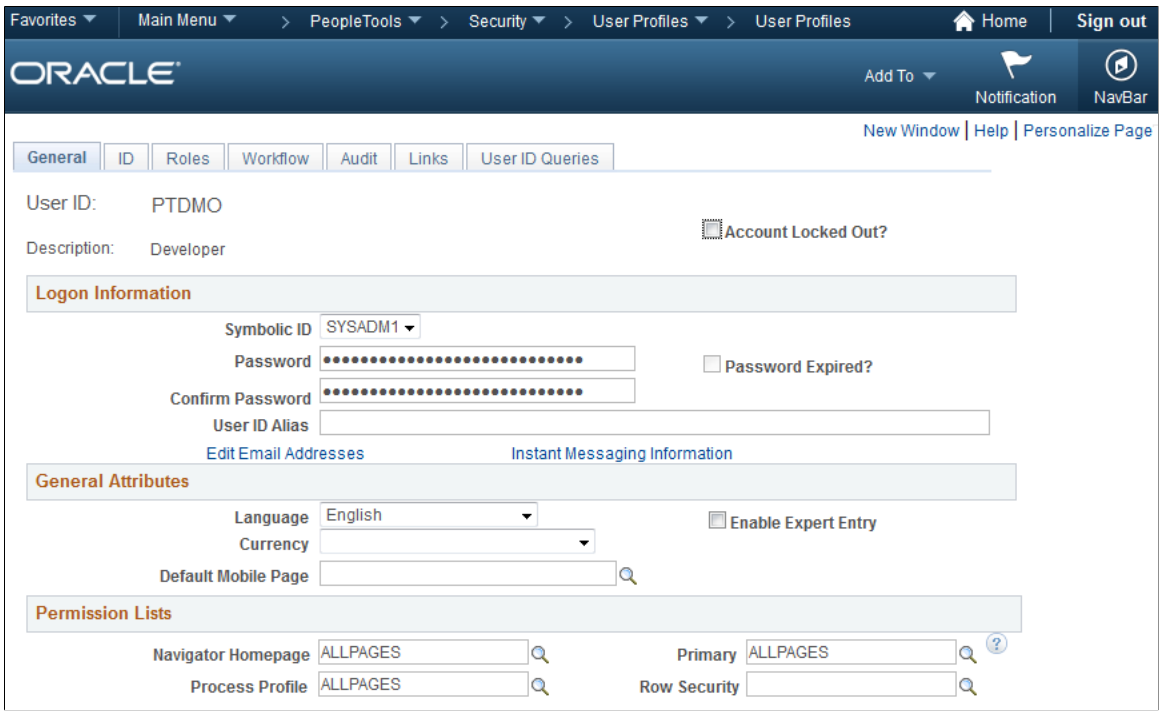
If you are configuring fluid pages to be accessible from desktop computers, which will also be accessing classic pages, it is recommended to update the system branding options so that both classic and fluid pages share a common header, making the transition between classic pages and fluid pages more seamless, from the end user perspective.

To configure the NavBar to appear in the classic portal header:

1. Select PeopleTools, Portal, Branding, Branding System Options.
2. On the Branding System Options page, set the following:
    - Default Branding Theme: *DEFAULT_THEME_TANGERINE_ALT*
    - Theme Style Type: *Tangerine (PT8.53 default)*
    - Default Style Sheet Name: *PSSTYLEDEF_TANGERINE*
3. Click Save.
4. Sign out and sign on to the system.

Image: Fluid Banner Displayed in Classic Mode

This example illustrates how the fluid user interface banner can be configured to display when accessing classic PIA pages.

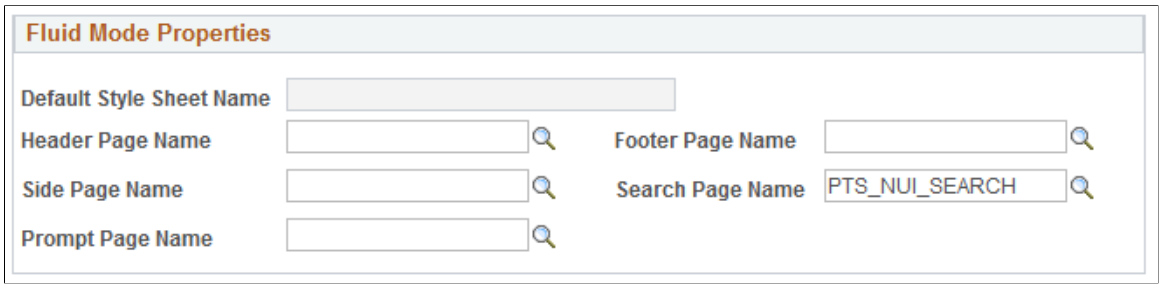


Configuring Fluid Mode PeopleTools Options Settings

The system-wide options for fluid mode applications appear on PeopleTools Options in the Fluid Mode Properties group box.

Image: PeopleTools Options: Fluid Mode Properties

This example illustrates the Fluid Mode Properties on the PeopleTools Options page. Descriptions of the properties appear after the example.



Default Style Sheet Name

Displays the default, system-wide style sheet for fluid components, which is defined on the Branding System Options page.

Note: If no style sheet is specified, then PSSTYLEDEF _FMODE is used as the default style sheet for all fluid components.

See "Administering System Branding" (PeopleTools 8.54: Portal Technology)

Header Page Name

The default header page is PT_HEADERPAGE.

If you want to override this page, enter your custom page.

Side Page Name

The default side page is PT_SIDEPAGE.

If you want to override this page, enter your custom page.

Prompt Page Name

The default prompt page is PT_PROMPTPAGE.

If you want to override this page, enter your custom page.

Footer Page Name

The default footer page is PT_FOOTER_SAVEONLY.

If you want to override this page, enter your custom page.

Search Page Name

The default search page is PTS_NUI_SEARCH.

If you want to override this page, enter your custom page.

Managing Fluid Homepages

PeopleTools provides a default homepage, named DEFAULT_LP. The underlying fluid page and component are both named PT_LANDINGPAGE. (The term landing refers to where the user “lands” after signing on to the system).

All fluid homepages are content references under the Fluid Homepage - Hidden folder. Set the content reference permission to control the access.

Default PeopleSoft User (PTPT1000) permissions are granted to the component. Managing and administering the system delivered fluid homepages requires the Portal Administrator permission list (PTPT1300).

Renaming a homepage is controlled by the All Rename option on the Homepage tab attributes.

Managing fluid Homepages on small form factors (smartphones):

- Personalizing fluid homepages *is not* supported from smartphones.
- For small form factor devices, the system only displays fluid homepages that contain tiles optimized for small form factors. If a fluid homepage contains tiles optimized for small form factors, the system displays only those tiles optimized for small form factors on small form factor devices. (Has the Display on Small Form Factor Homepage option selected for the component properties). See Setting Component Properties for Fluid Components.

For example, assume a fluid homepage contains a mixture of "regular” tiles and tiles optimized for small form factors. If an end user accesses that fluid homepage using a smartphone, that user will see only the tiles optimized for the small form factor device.

Note: On fluid homepages, the Actions list does not display a Help option, even when online help is configured for the system.

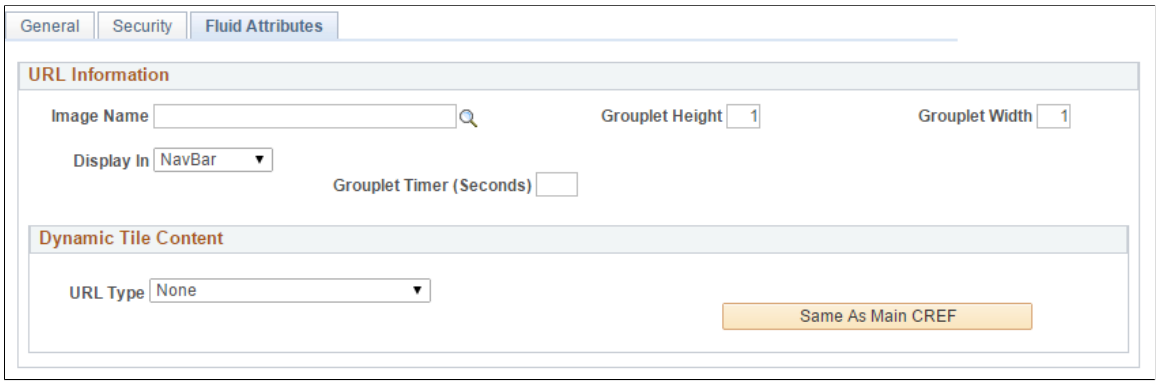
Note: Users signing in to the system using laptops with "touch" capability, will be routed by default to the fluid homepage. For example, this would be the case with users signing in with Windows 8 touch screen laptops. This is a result of the way the system interprets some of the capabilities sent from the browser's user agent running on the touch screen laptop. Currently, the system is interpreting the device type as a tablet, not a laptop PC. Assuming the users of the touch screen laptop do not need to access fluid homepages upon signin, they can update their user preferences for the Tablet Homepage to Classic. But this will force mobile devices used by that user, such as smartphones and tablets (if they use them to access the system), to be routed to the classic portal homepage as well. Also, you may consider conditional navigation. For more information, see "Understanding Conditional Navigation" (PeopleTools 8.54: Portal Technology).

Configuring Tile Options

To configure the behavior and appearance of tiles use the Fluid Attributes page of the target content reference. (PeopleTools, Portal, Structure and Content, and drill into the folder hierarchy as needed, then click Edit on the appropriate content reference.)

Image: Fluid Attributes

This example illustrates the fluid attributes for content references.



URL Information

Image Name

To display a custom image on the tile, select the image from the drop-down list. Otherwise the default image will appear.

Grouplet Height

Use to adjust the height of the tile.

The default dimensions of a tile are 1x1, with a maximum of 2 x 2.

Note: While you can set the value to higher than 2, at runtime, the system enforces a limit of 2.

Grouplet Width

Use to adjust the width of the tile.

Note: While you can set the value to higher than 2, at runtime, the system enforces a limit of 2.

Display In

Control how the target transaction or content appears once a user presses the tile. Options are:

- Cur Window: target content appears in the current browser window.
- NavBar: applies only when the tile is added to the NavBar. If this is selected when the tile is on the homepage, the tile displays in a modal, when pressed.
- Modal: target content appears in a modal window.

- New Window: target content appears in a new tab in the browser.

Grouplet Timer (Seconds)

To set an automatic refresh period for dynamic content on a tile, enter the time in seconds. When the timer limit has been reached, the system re-draws the tile so that it displays the current data, such as in the case with chart.

The system enforces a 10 second minimum limit. Any value entered less than 10 seconds will be ignored and treated as 10 seconds. When setting this value, be sure to monitor performance of the page refreshes and adjust accordingly.

The default value of 0, disables any automatic refresh.

URL Type

If the tile is a “live” tile, meaning it displays dynamic content, you have these options:

- PeopleSoft Component.
- PeopleSoft Generic URL.
- PeopleSoft Script.

See the following sections for information on each option.

Selecting None results in the system displaying the static image or a default static image if no image is defined on the content reference.

Same as Main CREF

If the desired parameters for the dynamic URL type for a component, a generic URL, or a script match those already entered on the Content Reference, General page.

Note: If end users pin (add) a link to a fluid application on the home screen of their device (such as a smartphone), PeopleTools provides a standard icon for this purpose. If end users modify the icon by adding a custom image, for example, that modification will be overwritten after applying a PeopleTools update or upgrade.

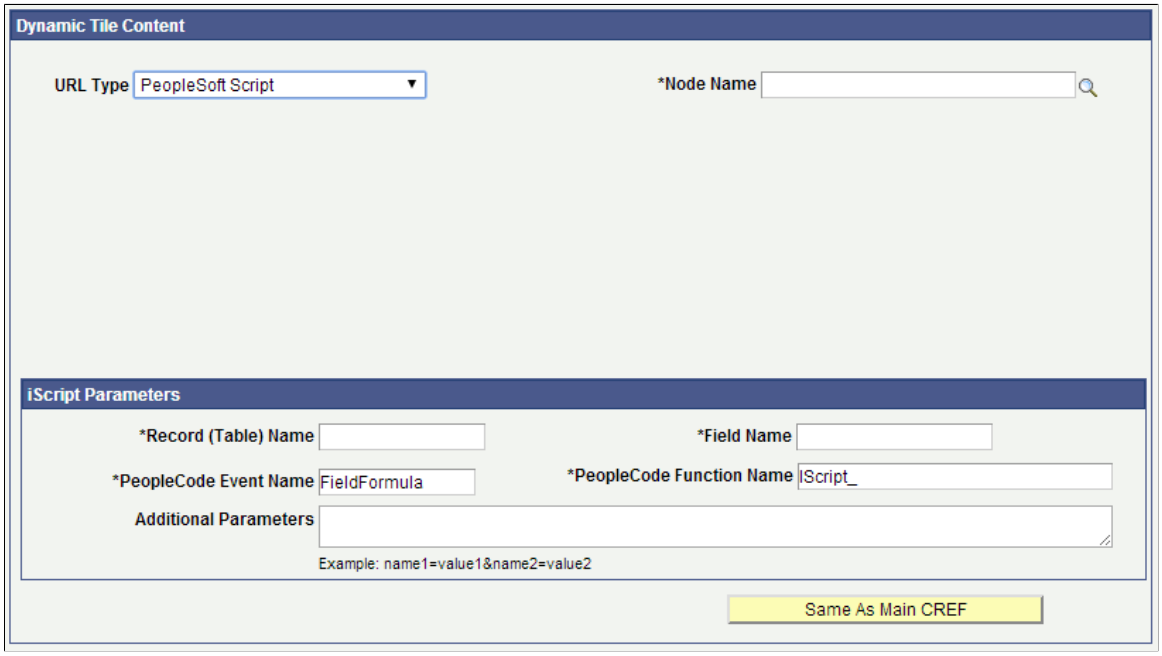
Dynamic Tile Content: PeopleSoft Component

After selecting a URL Type of PeopleSoft Component, the Component Parameters group box appears. Select the menu name, market, and component name, and so on to use for this content reference.

If the parameters required do not differ from those specified for the main content reference on the General page, press Same As Main CREF.

Image: Dynamic Tile Content

This example illustrates the options for configuring dynamic tile content for components.



Node Name	Select the node for the target component.
Menu Name	Enter the menu for the target component.
Market	Select the market for the target component.
Component Name	Enter the menu for the component name.
Page Name	Enter the fluid-enabled page name.
Grouplet ID	Reserved for future use.
Additional Parameters	Enter any additional parameters.

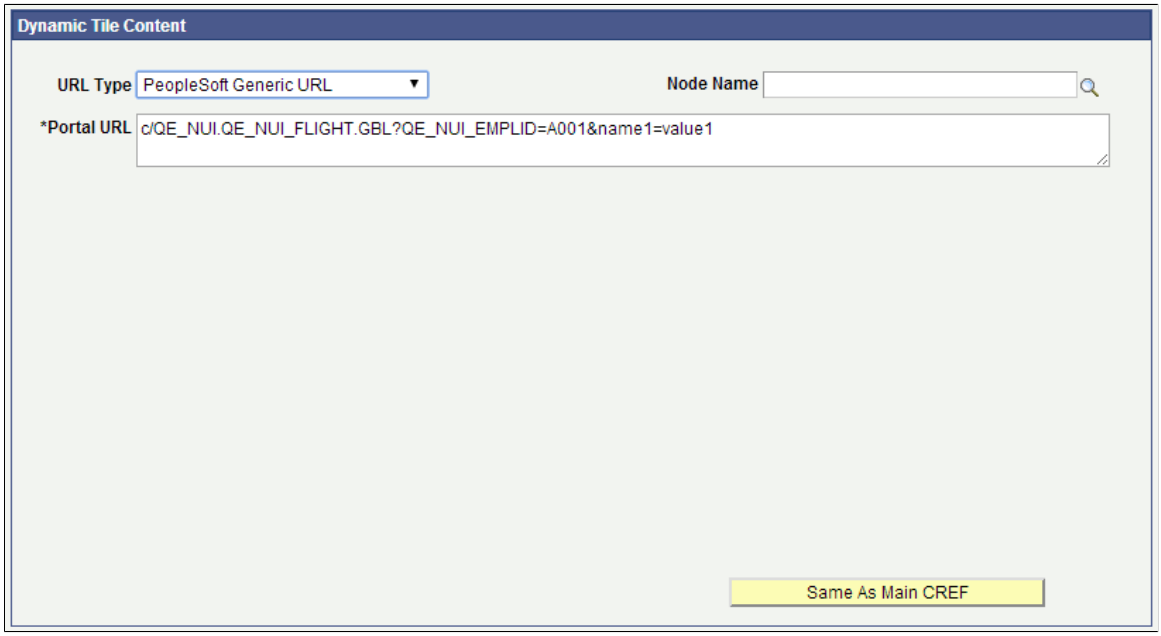
Dynamic Tile Content: PeopleSoft Generic URL

After selecting a URL Type of PeopleSoft Generic URL, options related to that URL appear. This usage type can be used in conjunction with a defined node. The result is the concatenation of the selected node's URI text and the portal URL that you enter.

If the parameters required do not differ from those specified for the main content reference on the General page, press Same As Main CREF.

Image: Dynamic Tile Content: PeopleSoft Generic URL

This example illustrates the options for configuring dynamic tile content for URLs.



Dynamic Tile Content: PeopleSoft Script

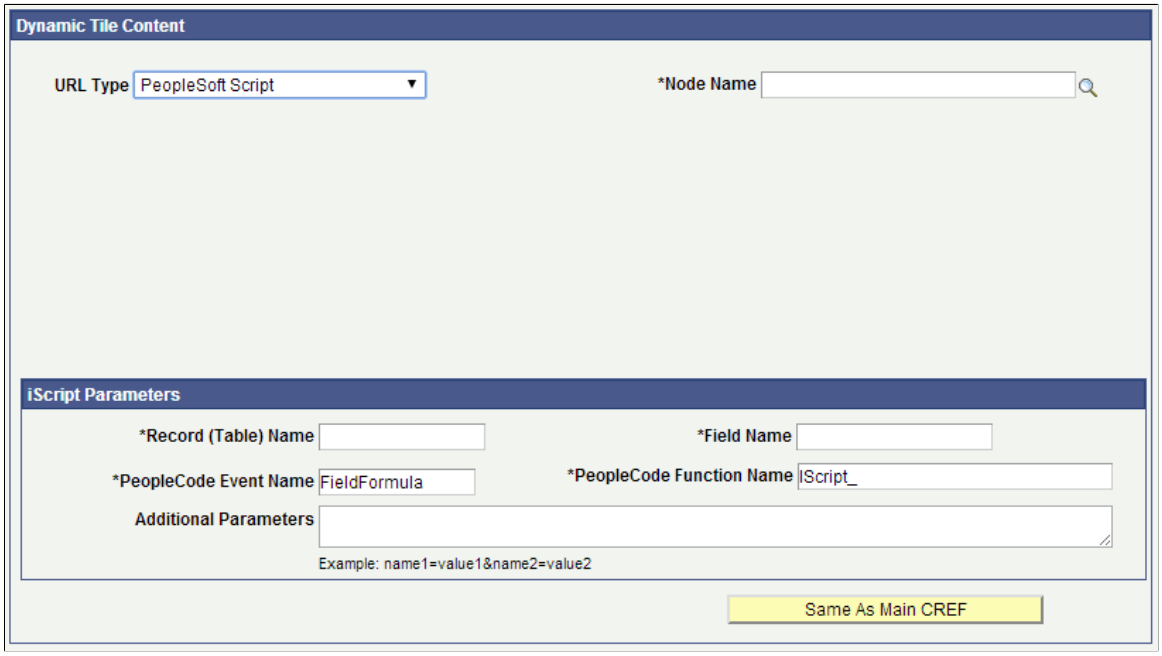
After selecting a URL Type of PeopleSoft Script, the iScript Parameters group box appears.

Select the record (table) and field names, PeopleCode event name, and PeopleCode function name that specifies the iScript to use, plus any additional parameters.

If the parameters required do not differ from those specified for the main content reference on the General page, press Same As Main CREF.

Image: Dynamic Tile Content: PeopleSoft Script

This example illustrates the options for configuring dynamic tile content for URLs.



Enabling Fluid User Interface for PC's and Laptops

If you want fluid pages and applications to be accessible from desktop computers, you will need to complete these tasks:

- Enable the web profile settings for fluid applications on the desktop.
- Set the homepage personalization fluid options.
- Configure the NavBar to appear in the classic portal header.

Enabling the Web Profile Settings for Fluid Applications on the Desktop

If you intend your end users on PC’s and laptop computers to access fluid applications, you need to set the fluid web profile settings appropriately.

For details on setting fluid web profile settings, see Configuring Fluid Mode Web Profile Settings.

Setting the Homepage Personalization Options for Fluid Applications

When deploying fluid user interface applications to your user base, consider the devices being used to access your system, and if any default personalization options need to be overridden.

For details on setting homepage personalization options for fluid user interface, see Setting Fluid Homepage Personalizations.

Configuring the NavBar to Appear in the Classic Portal Header

If you are configuring fluid pages to be accessible from desktop computers, which will also be accessing classic pages, it is recommended to update the system branding options so that both classic and fluid pages share a common header, making the transition between classic pages and fluid pages more seamless, from the end user perspective.

For details on configuring the NavBar to appear in classic mode, see Configuring the NavBar to Appear in the Classic Portal Header.

Managing Mobile Device Access

PeopleTools delivers a default icon (LOGO_FAVICON) that will be used when a user adds (pins) a link to any PeopleSoft URL on the home screen of their mobile device. If you replace the PeopleTools default icon with a custom image, that customization will be overwritten after applying a PeopleTools update or upgrade.