ORACLE®
PeopleSoft

PeopleTools 8.57: Mobile Application Platform

March 2020

ORACLE®

PeopleTools 8.57: Mobile Application Platform
Copyright © 1988, 2020, Oracle and/or its affiliates. All rights reserved.

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

vii

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 Online Help

You access the PeopleSoft Online Help on Oracle's PeopleSoft Hosted Online Help website, which enables you to access the full help website and context-sensitive help directly from an Oracle hosted server. The hosted online help 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 Online Help website is available in English only.

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 After a new release is posted, previous releases remain available but are no longer updated.

Locally Installed Help

If you are setting up an on-premises PeopleSoft environment, and your organization has firewall restrictions that prevent you from using the Hosted Online Help 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 Elasticsearch for full-text searching. See *PeopleSoft 9.2 Application Installation* for your database platform, "Installing PeopleSoft Online Help." If you do not use Elasticsearch, see the documentation for your chosen search engine.

Note: See Oracle Support Document 2205540.2 (PeopleTools Elasticsearch Home Page) for more information on using Elasticsearch with PeopleSoft.

Note: Before users can access the search engine on a locally installed help website, you must enable the Search field. For instructions, go to your locally installed PeopleSoft Online Help site and select About This Help >Managing Locally Installed PeopleSoft Online Help >Enabling the Search Button and Field in the Contents sidebar.

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)

Translations and Embedded Help

PeopleSoft 9.2 software applications include translated embedded help. With the 9.2 release, PeopleSoft aligns with the other Oracle applications by focusing our translation efforts on embedded help. We are not planning to translate our traditional online help and PeopleBooks documentation. Instead we offer very direct translated help at crucial spots within our application through our embedded help widgets. Additionally, we have a one-to-one mapping of application and help translations, meaning that the software and embedded help translation footprint is identical—something we were never able to accomplish in the past.

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 accessibility.

- Accessing, navigating, and searching the PeopleSoft Online Help.

- Managing a locally installed PeopleSoft Online Help website.

PeopleTools Related Links

PeopleTools 8.57 Home Page

PeopleTools Elasticsearch Home Page

"PeopleTools Product/Feature PeopleBook Index" (PeopleTools 8.57: Getting Started with PeopleTools)

PeopleSoft Hosted Online Help

PeopleSoft Information Portal

PeopleSoft Spotlight Series

PeopleSoft Training and Certification | Oracle University

My Oracle Support

Oracle Help Center

Contact Us

Send your suggestions to PSOFT-INFODEV_US@ORACLE.COM. Please include release numbers for the PeopleTools and applications that you are using.

Follow Us

Facebook.

YouTube

Twitter@PeopleSoft_Info.

PeopleSoft Blogs

LinkedIn

Chapter 1

Getting Started with the PeopleSoft Mobile Application Platform

Mobile Application Platform Overview

The PeopleSoft Mobile Application Platform (MAP) is a framework for building mobile applications.

The framework includes a Template Designer that enables you to define styles, formatting and other display options, and then quickly and easily apply the template to one or more application pages. The Layout Designer is the tool you use to develop the mobile application pages, including the user-interface, processing logic, and so on. Several utilities are included for uploading and managing images for your mobile applications. Several administrative tools are also included for managing metadata and security for your mobile applications. And a MAP WorkCenter is provided to provide easy access to the tools and features of the framework.

Developing Applications with the Mobile Application Platform

This topic contains information to consider before you develop mobile applications using PeopleSoft's Mobile Application Platform.

Determining Security Requirements

A security analyst should evaluate your security requirements for applications you develop using the Mobile Application Platform.

Assessing Staff Skills

Assess the skills of the people who will perform development and administrative functions of the mobile applications in your organization. Developers should have familiarity, training, or experience in the following areas:

- PeopleTools.

- PeopleSoft Documents technology.

- PeopleCode.

In addition, developers should understand and have research capabilities in HyperText Markup Language 5 (HTML5).

PeopleSoft's Mobile Application Platform also supports the following technologies. Knowledge of them is not required to use the framework, but may be helpful depending on your business requirements:

- Cascading Style Sheets level 3 (CSS3).

- JavaScript.

- jQuery.

- jQuery Mobile.

Other Sources of Information

In addition to the considerations presented in this topic, take advantage of all PeopleSoft sources of information, including installation guides, release notes, product documentation, curriculum, and red papers.

Chapter 2

Understanding the PeopleSoft Mobile Application Platform

Mobile Application Platform

The PeopleSoft Mobile Application Platform enables you to easily build mobile applications using mobile-first strategy and design principles. A "mobile-first" strategy focuses on developing and optimizing applications for mobile devices first, and provides options to scale up for larger devices, such as desktops and laptops.

The framework uses a template-layout approach for building mobile applications.

Templates use the concept of containers that enable you to create reusable grid-based layouts, with pre-defined display properties. You can define media queries on templates that enable your application pages to adapt to the constraints of the devices that render them. Templates automatically generate cascading style sheets (CSS) from the metadata you define in them, and in turn the system uses the CSS to define how the content and mobile application pages are rendered.

The layout is where you create the mobile application by selecting a template and adding elements to the containers defined in the template. Each layout can contain one or more application pages.

Just as rowsets supply the data structure for PIA-based applications, PeopleSoft documents supply the data structure for mobile applications that you build in the framework. Application classes are the primary objects to use to add business logic and to dynamically change the layouts of mobile applications.

REST services provide the communication layer between mobile applications and PeopleSoft. When you save a layout, behind the scenes the system creates a REST service that the mobile application uses to communicate with the PeopleSoft database.

Mobile Application Platform Designers, Tools and Utilities

You can access Mobile Application Platform designers, tools and utilities in the PeopleSoft Pure Internet Architecture (PIA) by selecting *PeopleTools, Mobile Application Platform.*

Template Designer	Use the Template Designer to build templates that you can apply to layouts to define how application pages display.
Layout Designer	Use the Layout Designer to build your application pages, including building the user interface, adding business and processing logic, and so on.

Administration Tools	PeopleSoft delivers a number of tools for managing and synchronizing Mobile Application Platform metadata, administering security, and more.
Utilities	Mobile Application Platform utilities include:

- Layout Image Upload — Use this utility to upload images for the mobile application.

- Dynamic Image Registration — Use this utility to store images on a web server for use by an application at runtime.

MAP WorkCenter	A MAP WorkCenter is delivered that provides centralized access to Mobile Application Platform designers, tools, and utilities.

Mobile Application Platform Security

There are several types of security that you can implement for mobile application that you build with the Mobile Application Platform.

Restricted Access to Templates and Layouts

You can control developer access to Mobile Application Platform application pages by restricting templates and layouts.

When you restrict a layout or template, it cannot be changed or deleted from the database.

Permission List Access to Mobile Application Pages

You can use PeopleSoft permission lists to grant user-profile access to invoke specific pages in your applications.

Authentication

You can secure the mobile applications that you develop with Mobile Application Platform using the PeopleSoft authentication framework.

You can secure your mobile application pages using:

- Basic authentication.

- Basic authentication and SSL.

- PeopleSoft token.

- People Soft token and SSL.

- SSL.

PeopleSoft delivers a configurable sign-on page for use if you use PeopleSoft token or PeopleSoft token and SSL authentication.

Mobile Application Platform Development Steps

In general the steps for building a mobile application using the PeopleSoft Mobile Application Platform are:

1. Design the application.

2. Create the application data structure using PeopleSoft Documents.

3. Design layout templates in the Template Designer.

4. Build the application in the Layout Designer.

5. Test the application.

Chapter 3

Understanding Mobile Application Platform Metadata

Layouts

In the Mobile Application Platform framework, a layout is the actual mobile application, and it can contain one or more application pages.

The end result of building a layout is a URL that's used to invoke the application.

PeopleSoft documents define the data structure for Mobile Application Platform applications. To add a Mobile Application Platform application to the database, you must specify a PeopleSoft document that contains the data structure for the application.

A sample of items that you define for each page in a layout includes:

- Elements that appear on the page. For example, text, check boxes, buttons, and other page controls.

- Main layout classes, style, and color themes.

- Content classes, styles, and color themes.

- Page transitions. For example, flip, slide, turn, and so on.

- For each page in a layout, you specify the elements to appear on the page. For example, text, check boxes, buttons, text fields, and other page controls.

PeopleSoft provides a Layout Designer for building a mobile application.

Layout Templates and Sub-Templates

This topic describes layout templates and sub-templates.

Layout Templates

Just as a word processing template contains default styles and formatting to apply to word processing documents, a layout template contains default styles, formatting, media queries and other options that impact the display of mobile application pages that you develop using the PeopleSoft Mobile Application Platform.

Layout templates are not required to build applications using the PeopleSoft Mobile Application Platform, however they provide the following benefits:

- Speed up development.

You can define styles, formatting and other display options in a layout template, and then quickly and easily apply the template to one or more application pages.

- Provide consistency among mobile applications and mobile application pages.

    Since styles, formatting, and other display options are stored in the template, you're assured that every mobile application page to which a template is applied features the same styling and display options.

    If you're building multiple mobile applications and want them to have the same look and feel, using templates can help you achieve that objective, especially if different development teams and developers are involved in developing the applications.

- Simplify customization and maintenance.

    Customizing and maintaining mobile application pages is simplified when you use templates, because you make changes in one place (the template) and they are then with the push of a button you can apply the changes to your mobile application pages to which the template is applied.

Layout templates are managed objects. The Mobile Application Platform framework provides a Template Designer that you use to create and manage layout templates. You apply layout templates to application pages in the Layout Designer.

Layout Sub-Templates

In the Mobile Application Platform framework you can define a template as a sub-template and then assign the sub-template to a template.

Creating and using sub-templates is way to re-use styles and other display options at a more granular level than with regular templates.

For example, you could have a requirement where you need to use differently styled footers for application pages. You could create several footer sub templates. And then when you create the main template, you could assign a sub-template that featured the appropriate footer styling for the main template.

You cannot use sub-templates as stand-alone templates. You must attach them to a regular template.

Related Links
Understanding Using the Template Designer

PeopleSoft Documents

To create a mobile application in the Mobile Application Platform framework, you must link a PeopleSoft document to the application.

PeopleSoft Documents provide the data structure for mobile applications you build with the Mobile Application Platform framework.

If you are familiar with developing applications for PIA, the following comparisons may be helpful.

- A document in the Mobile Application Platform framework is similar to a rowset in PIA.

In PIA, a rowset provides the data structure for a PIA page. In the Mobile Application Platform, a PeopleSoft document provides the data structure for your mobile application pages contained in a layout.

- A parent PeopleSoft document roughly equates to a Level 0 record in the PIA framework. And document compound and collection elements roughly equate to Level 1 records in the PIA framework, however unlike a rowset, there does not have to be a parent-child relationship with documents.

When developing a PeopleSoft document for a Mobile Application Platform application keep in mind that you are always mapping to a logical document. PeopleSoft provides a Document Builder and several utilities for creating and managing documents.

For more information see "PeopleSoft Documents Technology Overview" (PeopleTools 8.57: Documents Technology)

Chapter 4

Understanding Mobile Application Platform Programming Technologies

Cascading Style Sheets (CSS)

The Mobile Application Platform framework supports the Cascading Style Sheets level 3 (CSS3) specification of the World Wide Web Consortium (W3C). According to W3C at the time of this writing:

"CSS (Cascading Style Sheets) is a language for describing the rendering of HTML and XML documents on screen, on paper, in speech, etc. It uses color-related properties and values to color the text, backgrounds, borders, and other parts of elements in a document."

See www.w3.org for more information.

As a result, CSS enables you to separate the presentation style of a web or mobile application page from the information that appears on a page and the logic that runs on the page.

The description of the styling information is contained in a CSS document or file with a .CSS extension.

Note: In this documentation the term *CSS* refers to CSS3 style constructs and/or the file or files that contain the defined constructs.

CSS Types and Sources in the Mobile Application Platform

In the Mobile Application Platform the CSS styles you use in a mobile application can be:

System-generated   When you create a layout template or sub-template in the Template Designer, the system generates a CSS file from the container metadata upon save.

Manually-coded   You can manually code and add CSS to templates and sub-templates in the Template Designer. You can also manually code and add CSS to layouts in the Layout Designer.

Uploaded   You can add CSS to application pages in the Layout Designer. You can define CSS you add in the Layout Designer as a file or inline within the generated HTML. When you add CSS to a layout using the file option, it is pushed to the web server as a file and referenced in the generated HTML as a link. CSS added with the file option, the CSS will be pushed to the web server as a file and referenced in the generated HTML as a link.

Order of Precedence for Applying CSS

The order of precedence for how the system applies CSS to application pages at runtime is as follows:

Order	Source	Component	Type
1	Layout	Layout Designer	Manually coded.
2	Layout	Layout Designer	Manually coded defined as a file.
3	Layout template	Template Designer	System generated in the Template Designer from template container metadata
4	Layout template	Template Designer	Manually added to a template in the Template Designer.
5	Layout sub-template	Template Designer	System generated in the Template Designer from sub-template container metadata.
6	Layout sub-template	Template Designer	Manually coded.

Hypertext Markup Language (HTML)

Hypertext markup language (HTML) is a markup language for structuring and displaying content in web pages and web-based applications.

The mobile applications that you develop using the Mobile Application Platform framework are structured in HTML version 5, or HTML5.

JavaScript

JavaScript is a client-side and browser-based scripting language.

The Mobile Application Platform supports JavaScript. You can use JavaScript and JavaScript plug-ins while building templates in the Template Designer.

You can also use JavaScript, JavaScript events, and Ajax events in the Layout Designer when building your mobile applications using the Mobile Application Platform.

jQuery

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. The Mobile Application Platform supports jQuery scripts.

In PeopleTools 8.56, JQuery was updated to Version 2.1.4, as part of this upgrade these files are no longer part of the Web Server install for the Gateway. The old files will remain under the directory (map/defaults) in order for the Gateway to support older tools releases. For 8.56 and above these files are part

of the PT_MAP_FRAMEWORK Template. When a File Sync is performed these files are propagated to the Gateway. This allows different versions of JQuery/ JQuery Mobile to coexist on a single Gateway, as well simplifying the update process for these files.

jQuery Mobile

jQuery Mobile is an HTML5-based framework that uses CSS3 and themes for building applications. The jQuery Mobile API supports Touch events as well as Mouse events usually associated with jQuery. As a result you can use jQuery Mobile to build applications that are accessible on mobile and desktop devices.

The Mobile Application Platform supports jQuery Mobile scripts.

In PeopleTools 8.56, JQuery Mobile was updated to Version 1.4.5, as part of this upgrade these files are no longer part of the Web Server install for the Gateway. The old files will remain under the directory (map/defaults) in order for the Gateway to support older tools releases. For 8.56 and above these files are part of the PT_MAP_FRAMEWORK Template. When a File Sync is performed these files are propagated to the Gateway. This allows different versions of JQuery/ JQuery Mobile to coexist on a single Gateway, as well simplifying the update process for these files.

PeopleCode

Application classes are the primary means to initialize, add business logic, and dynamically change layouts for Mobile Application Platform applications.

The Mobile Application Platform framework completely separates the business logic from the presentation layer. All Mobile Application Platform metadata is available in the application class PeopleCode layer and is not tied to application pages or other metadata, allowing you to override the defaults defined on the actual layout. This means that you have complete control of the presentation layer for any transaction within the application class PeopleCode.

PeopleSoft delivers two Mobile Application Platform application interface class methods that enable you to manipulate pages, elements, and other aspects of your applications via PeopleCode you define in layouts.

31

Chapter 5

Using the MAP WorkCenter

Using the Mobile Application Platform WorkCenter

Use the MAP WorkCenter page (IB_MAP_WRKCNTR) to perform the tasks necessary to create a MAP application..

Navigation

PeopleTools > Mobile Application Platform > MAP WorkCenter

Image: Mobile Application Platform WorkCenter page

This example illustrates the fields and controls on the Mobile Application Platform WorkCenter page. You can find definitions for the fields and controls later on this page.
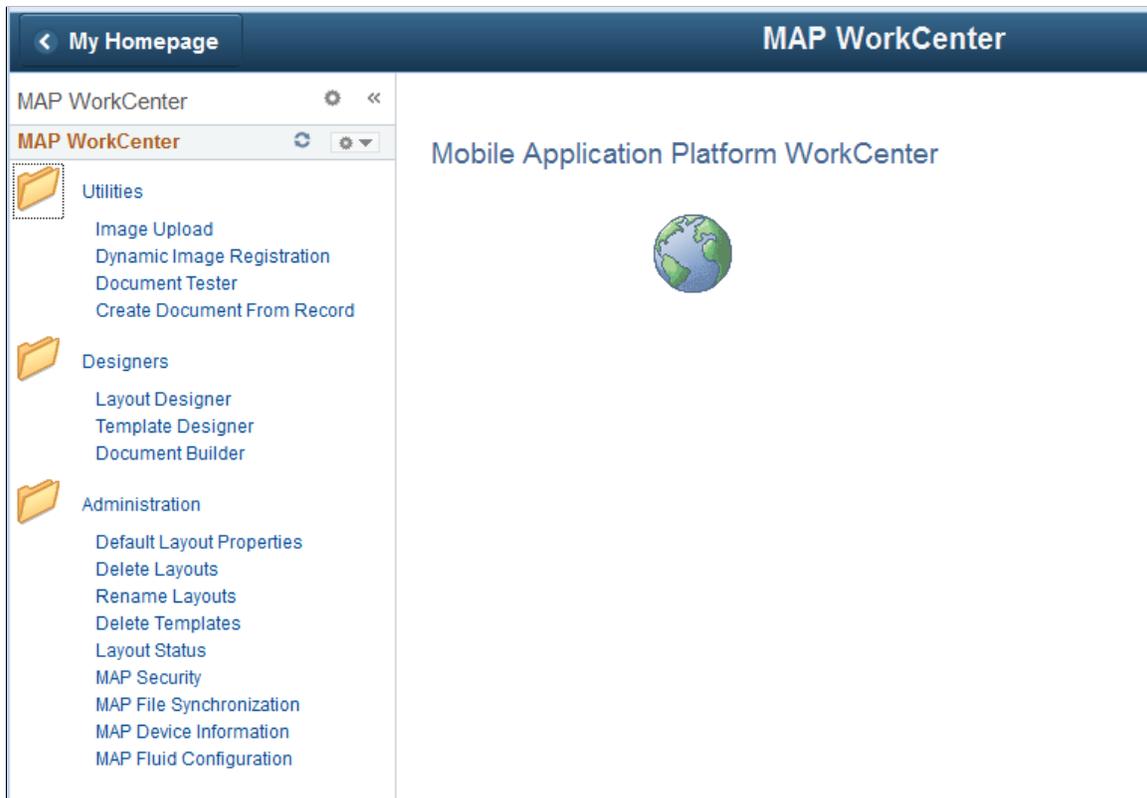
The workcenter contains 3 folders related to the type of task to be performed:

Utilities

Image Upload	See Using the Upload Image Page
Dynamic Image Registration	See Using the MAP Dynamic Image Registration Page

Document Tester	See "Understanding Testing Documents" (PeopleTools 8.57: Documents Technology)
Create Document From Record	See "Creating Documents from Records" (PeopleTools 8.57: Documents Technology)

Designers

Layout Designer	See <u>Navigating the Layout Designer</u>
Template Designer	See <u>Understanding Navigating the Template Designer</u>
Document Builder	See "Accessing the Document Builder" (PeopleTools 8.57: Documents Technology)

Administration

Default Layout properties	See <u>Defining Default Layout Options</u>
Delete Layouts	See <u>Deleting Layouts</u>
Rename Layouts	See <u>Renaming Layouts</u>
Delete Templates	See <u>Deleting Layout TemplatesDeleting Layout Templates</u>
Layout Status	See <u>Using the Layout Status Page</u>
MAP Security	See <u>Administering Security</u>
MAP File Synchronization	See<u>Synchronizing MAP Files</u>
MAP Device Information	See <u>Reviewing Map Device Information</u>
MAP Fluid Configuration	See <u>Configuring MAP for Fluid</u>

Chapter 6

Using the Template Designer

Understanding Using the Template Designer

The Mobile Application Platform features a Template Designer for building layout templates.

Using the Template Designer you:

- Define the sections of an application page. For example, define the main content area, the header, the footer, and so on. In the Mobile Application Platform framework, these sections are called *containers*.

- Define CSS style rules for each container, such as selectors, selector properties, and selector values.

- (Optional.) Define media queries that specify how application pages render on different mobile devices.

- (Optional.) Add CSS and CSS plug-ins to run on application pages.

- (Optional.) Add JavaScript events and JavaScript plug-ins to run on application pages.

The end result of building a template in the Template Designer is a template that you can apply to one or more application pages or applications in the Layout Designer.

Layout templates that you create in the PeopleSoft system are managed objects.

When you save a layout template the system automatically generates a CSS3-compliant style sheet that becomes part of the template. (As described previously in this section, you can also manually add CSS, CSS plug-ins and JavaScript to templates.)

Understanding Navigating the Template Designer

The Template Designer is located in the IB_DOCTPLDEFN component in the PeopleSoft Pure Internet Architecture.

To access the Template Designer, select PeopleTools >Mobile Application Platform >Template Designer.

Image: Template Designer

This example illustrates the default view of the Template Designer.

The Template Designer features six pages. The following table lists and briefly describes the pages and how to access them:

Page	Object ID	Description	Navigation
Template Designer	IB_DOCTPLDEFN	Use the page to: • Add a layout template or sub-template to the system. • Attach sub templates to a template. • View a template layout.	PeopleTools > Mobile Application Platform, >Template Designer.
Containers	IB_DOCTPLDEFN3	Use the page to: • Add containers to a template. • Define container selectors and selector properties. • Define container elements.	PeopleTools >Mobile Application Platform >Template Designer and click the Containers tab.
Media Queries	IB_DOCTPLDEFN5	Use the page to define how a mobile application page renders on specific devices.	PeopleTools >Mobile Application Platform >Template Designer and click the Media Queries tab.

Page	Object ID	Description	Navigation
Viewer	IB_DOCTPLDEFN4	Use the page to document and view information about the template or sub-template.	PeopleTools >Mobile Application Platform >Template Designer and click the Viewer tab.
CSS	IB_DOCTPLDEFN2	Use the page to: • View system-generated CSS code. • Enter custom CSS code and plug-ins.	PeopleTools >Mobile Application Platform >Template Designer and click the CSS tab.
JavaScript	IB_DOCTPLDEFN6	Use the page to stage JavaScript code, libraries, and plug-ins.	PeopleTools >Mobile Application Platform >Template Designer and click the JavaScript tab.

Using the Template Designer Page

Use the Template Designer page to add a template or sub-template to the database. Use the page to access pages to add one or more sub-templates to a template and to view a textual representation of a template.

To access the Template Designer page (IB_DOCTPLDEFN) select PeopleTools, Mobile Application Platform, Template Designer.

Image: Template Designer – Template Designer page

This example illustrates the fields and controls on the Template Designer – Template Designer page. You can find definitions for the fields and controls later on this page.

Template Name	Displays the name of the layout template.
Use in Rapid App	Select the box to designate that this template can be used when creating layouts and applications using the Rapid App Designer.

Sub Template	Select the box to designate the template a sub-template.
Attach Sub Templates	Click the link to access the Add Sub Template page to add one or more sub-templates to the main template. This control does not appear on templates you define as sub-templates; you cannot nest sub-templates in sub-templates.
Description	Enter a description for the template.
Long Description	Enter a long description for the template.
Object Owner ID	From the drop-down list select the owner of the template.
View Template Layout	Click the link to access the Template page and to view a text-based representation of the template.
Save	Click the button to save any changes to the definition.

Related Links

Adding Layout Templates

Using the Add Sub Templates Page

Use the Add Sub Template page to add one or more sub-templates to a template.

To access the Add Sub Template page (IB_DOCTPLSUB_SEC), on the Template Designer page click the Attach Sub Templates link.

Image: Add Sub Templates page

This example illustrates the fields and controls on the Add Sub Templates page. You can find definitions for the fields and controls later on this page.

Template Name	Displays the name of the main template.
Name	Click the Lookup button to search the database for a sub-template to add.
Go to Sub Template	Click the link to access the definition for the sub- template.

Related Links

Adding Layout Sub-Templates

Using the Template Page

The Template page displays the structure of the template. The intent is to let developers quickly view the structure of a template to evaluate using it in a layout.

The information that appears on the page is defined in the Template Designer – Viewer page. Depending on how the template developer documented the template contents, the Template Designer – Viewer page may contain additional information to assist in understanding the template.

To access the Template page (IB_DOCLOTMPLVW_SEC) select PeopleTools >Mobile Application Platform >Template Designer and click the View Template Layout link.

Image: Template page

This example illustrates the Template page:
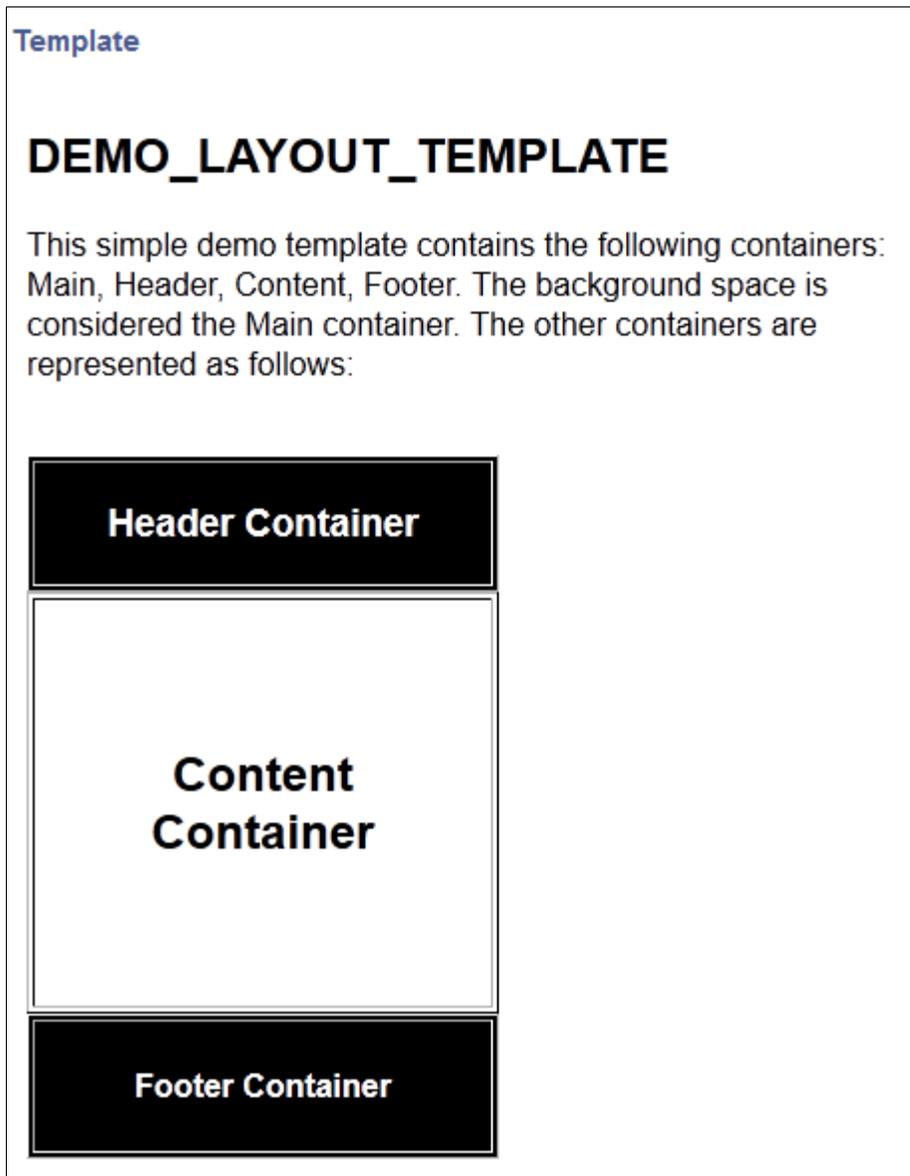
When you are done viewing the information on the page, click the Return button to go back to the Template Designer page.

Chapter 7

Adding Layout Templates

Adding Layout Templates

This topic discusses how to:

- Name layout templates.

- Add main containers to layout templates.

- Add HTML to layout templates.

Understanding Adding Layout Templates

To add a layout template to the system at a minimum you must:

- Name the template.

- Add a main container to the template.

- Add HTML to the template.

Note that you cannot save a template until you have provided the minimum information.

Note: This topic describes the minimum steps and information to enter to add a layout template to the PeopleSoft database.

Naming Layout Templates

To name a layout template:

1. Select PeopleTools >Mobile Application Platform >Template Designer.

2. Click the Add Template tab.

3. In the Template Name field enter a name for the template.

4. Click the Add button.

   The Template Designer page appears.

5. In the Description field enter a short description for the template.

6. In the Long Description field enter a more lengthy description for the template.

The next section describes how to add a main container to a layout template.

Adding Main Containers to Layout Templates

To add a main container to a template:

1. Click the Containers tab. The Template Designer - Containers page appears.

2. In the Container Class Name field, enter a class name for the container.

3. Select the Main Container box.

4. In the Comments box enter any comments about the container.

Proceed to the next section for information about adding HTML to the template.

Adding HTML to Layout Templates

To add a template to the database you must enter some HTML in the HTML long edit box in the Template Designer – Viewer page.

This page is where you document information and details about the template, so that other developers may understand the template contents and evaluate using it for their applications.

When you initially add a template to the database you can add as little as opening and closing HTML tags in the HTML box. For example:

```
<html>
</html>
```

To add HTML to a layout template:

1. Click the Viewer tab. The Template Designer – Viewer page appears.

2. In the HTML long edit box, enter some HTML related to your template.

3. Click the Save button.

Chapter 8

Managing Layout Template Containers

Understanding Layout Template Containers

Containers in the Mobile Application Platform framework define the sections of a mobile application page.

A mobile application page is typically divided into three parts: header, content, and footer. In the Mobile Application Platform framework, these sections are considered containers.

The only mandatory container in a layout template is a main container. The main container in a template contains the default style information for the entire application page. You can define additional containers for different content sections of a page, and define specific styles and formatting for each of them.

Note: The only mandatory container in a layout template is a main container.

Image: Layout Template Containers

Most mobile application pages feature four (4) sections, or containers: a main container, which represents the application page; a header container; a content container; and a footer container, as the following example shows:
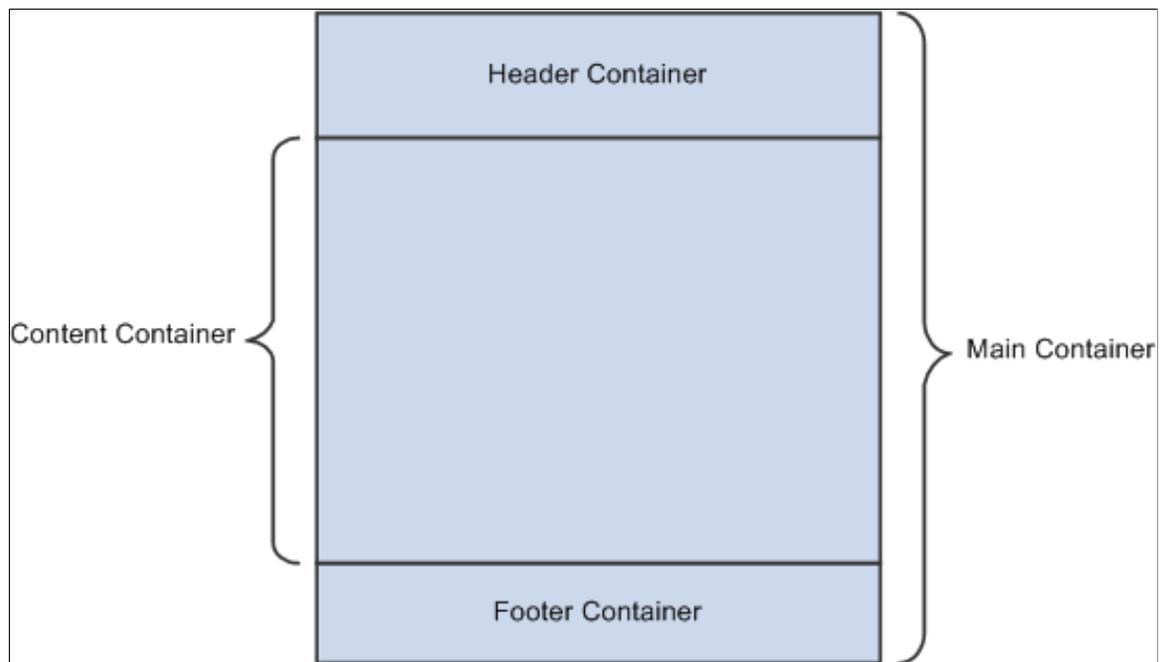
In this example, the content, header, and footer containers are separate containers that more or less sits on top of the main container. You can, however, build nested containers, where you build containers within containers.

43

When building your mobile application in the Layout Designer, you then program to each container to determine how elements are rendered and how they behave.

Understanding Layout Template Container Terms and Concepts

This topic describes terms and concepts associated with managing layout template containers.

Layout Template Comments

Many of the pages you use to manage layout templates contain Comments fields.

As you create and manage containers it is important for you to include as much information as possible in the Comments fields, because they are included in the generated CSS file. And more importantly, developers can use the information to determine if a given container or container selector should be used in his or her layout.

Containers

Containers define the sections of an application page. At a minimum, you must define a main container when you create a layout template.

Container Selectors and Selector Properties

Container selectors and selector properties comprise a style rule set.

Container selectors are equivalent to CSS3 selectors.

Selector properties are equivalent to CSS declaration blocks. A selector property consists of a declaration property and its defined value.

The use of container selectors and properties depends on your mobile application requirements and are optional.

This table lists container selectors supported by the Mobile Application Platform framework and shows the selector naming convention when defining the selector name.

Selector Type	*Naming Convention for Selector Name*
Attribute	The selector name is contained in brackets ([ ]). For example: `[selector_name]`
Class	The selector name is prefaced by a period (.). For example: `.<selector_name>`

Selector Type	Naming Convention for Selector Name

Contextual	The format for this selector is:
	
	`<context><space><element>`
	
	For example:
	
	`h1 em`
	
	Given this example the styles defined in the selector properties are applied to the *em* element found in the *h1* element.

Type	The element type to which the rule set applies is the selector name. For example: `span` In this example, the style properties defined for the selector with the name *span* apply to all content within the span element.
ID	The selector name is prefaced by a hash (#). For example: `#<selector_name>`
Psuedo-element	The selector name is prefaced by a colon (:). For example: `:<selector_name>`

Blob Properties

Blob properties enable you to define a block of CSS3 styles as a single selector property.

Container Elements

Container elements are the default page elements that you define for a container.

The page elements that you can define for a container are:

- Action sheet.
- Button.
- Check box.
- HTML area.
- Image.
- Static text.
- Input box.
- List view.

- Mobile grid.

- Radio button.

- Select box.

- Select menu.

- Sidebar.

- Static text.

- Table.

- Toggle.

- URL.

Themes

jQuery Mobile themes control specific visual elements such as fonts, colors, gradients, shadows, and corners.

The Mobile Application Platform framework enables you to apply the default JQuery Mobile theme and its default color swatches to user interface elements in a layout template.

A color swatch is defined by letters, *a* to *z.* The default theme used for layout templates include swatches *a* to *e.*

This table lists and describes the default color swatches available in the Template Designer:

Letter	*Description*	*Color*
a	Highest level of visual priority.	Black.
b	Secondary level of visual priority.	Blue.
c (Default.)	Baseline level.	Silver.
d	Alternate secondary level.	Gray.
e	Accent.	Yellow.

You can also create custom themes using jQuery ThemeRoller.

Using the Containers Page

To access the Containers page (IB_DOCTPLDEFN3) select PeopleTools >Mobile Application Platform >Template Designer and click the Containers tab.

Image: Template Designer – Containers page

This example illustrates the fields and controls on the Template Designer – Containers page. You can find definitions for the fields and controls later on this page.
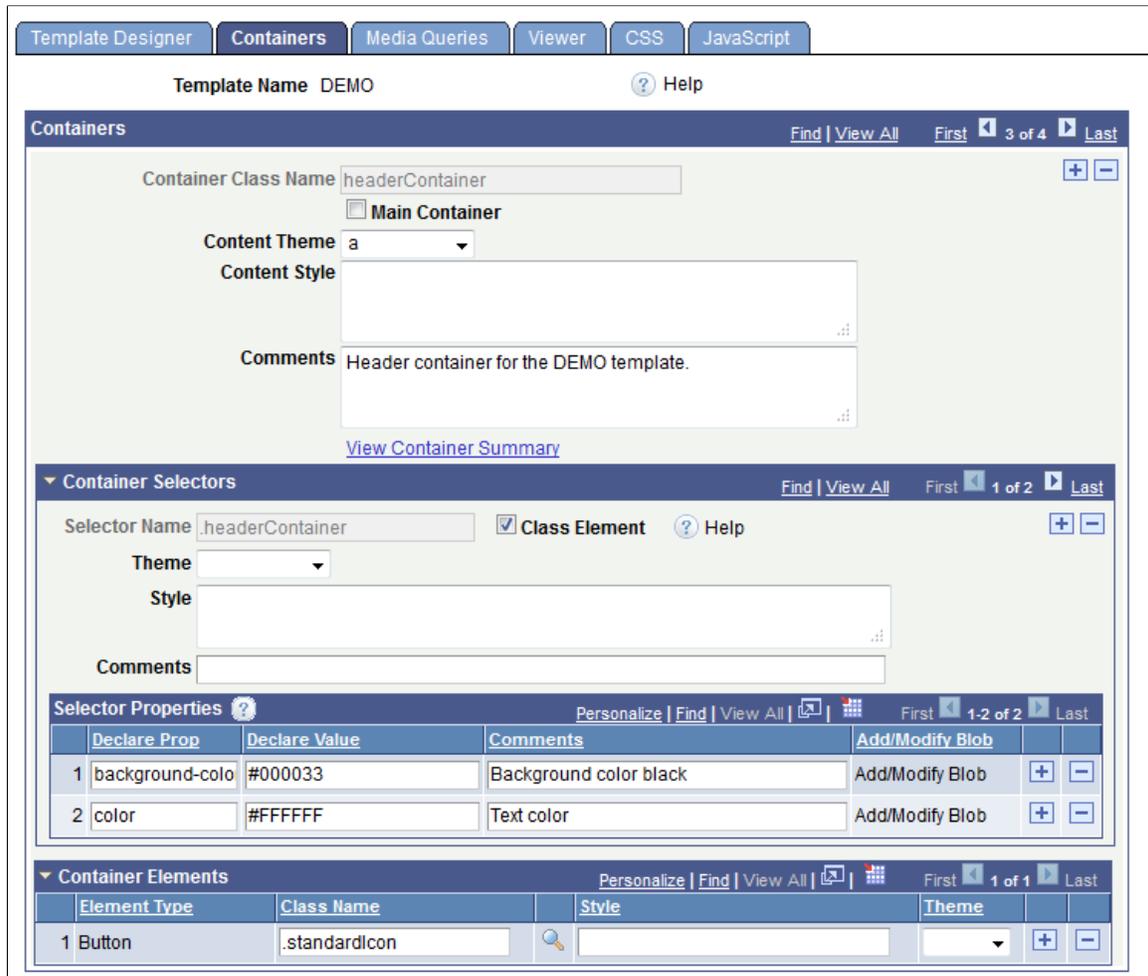
This table describes common elements used on the Template Designer – Containers page:

Content Style/Style	Enter style information in CSS3 format to apply.
	The styles you define in this field are individual styles.
	Use the Container Selector section to define and apply CSS selectors, such as class selectors, ID selectors, element selectors, and so on, for the container.
Content Theme/Theme	From the list choose a jQuery mobile color theme for the container or selector.
Comments	Enter comments about the container, selector, or selector properties.

This table describes the unique fields and controls that appear on the Template Designer – Containers page:

Template Name	Displays the name of the template.
Container Class Name	Enter the name of the container class.
Main Container	Select the box to define the current container as the main template container. There can be only one main container per template.
	This control is disabled when defining and working with sub-templates.
View Container Summary	Click the link to access the Container Summary page to view a text-based summary/description of the container.
Selector Name	Enter the name of the selector. If you do not specify a name, the first selector defined defaults to the Class type selector using the class name, prefaced with a period (.). See "Selector Types" for information on selector types supported.
Class Element	Select the box to define the selector as a Class element. By default the first selector defined for the main container defaults to a Class element type.
Declare Prop	Declare a CSS3 property for a selector.
	See <u>W3C – Cascading Style Sheets</u>
Declare Value	Declare a value for a CSS3 selector property.
Add/Modify Blob	Click the link to access the Selector Properties Blob page to define a block of CSS3 selector properties and values.
Element Type	Select an element to include in the container
Class Name	Enter a selector to apply to the element or use the Lookup button to search the database for a value.

Using the Container Summary Page

Use the Container Summary page (IB_CONTSUMMARY_SEC) to view a text-based summary of the containers defined for a layout.

To access the Container Summary page click the View Container Summary link on the Template Designer – Containers page.

Image: Container Summary page

This example illustrates the Container Summary page.

Container Summary

Container : headerContainer - Header container for the DEMO_LAYOUT_TEMPLATE.

Theme :
Style :

Selectors

.headerContainer (Class Element)

Theme :
Style :

Selector Properties

background-color : #000033 - Background color black
color : #FFFFFF - Text color white

==

Container Fields

Button
Class : .standardIcon
Theme :
Style :

Adding Containers

Use the Containers section of the Template Designer – Containers page (IB_DOCTPLDEFN3) to add containers to a template.

Image: Containers section of the Template Designer – Containers page

Use the Containers section of the Template Designer – Containers page to add containers to a template.

Before you can add a container to a layout template, you must first add a template to the system.

To add a container:

1. Access the Template Designer – Containers page.

   (Select PeopleTools >Mobile Application Platform >Template Designer and click the Containers tab.)

2. In the Container Class Name field enter a name for the container.

3. (Optional.) Select the Main Container box to designate the container a main container.

4. (Optional.) From the Content Theme drop-down list, select a theme to apply to the container.

5. (Optional.) In the Content Style field, enter CSS3 styles for the container.

6. Click the Save button.

   Important! Before you save a container definition, note that if you do not define a container selector the system creates one. The system creates a class element type selector and uses the container class name a as the selector name.

Use the Container Selector section to define CSS selectors, such as class selectors, ID selectors, element selectors, and so on, for the container.

Click the Add a new row button (+) to add and define another container.

Adding Selectors and Defining Selector Properties

Use the Container Selectors section of the Template Designer – Containers page (IB_DOCTPLDEFN3) to add container selectors and define selector properties.

Image: Template Designer – Containers page

Use the Container Selectors section of the Template Designer – Containers page (IB_DOCTPLDEFN3) to add container selectors and define selector properties.

To add selectors and define selector properties:

1. Access the Template Designer – Containers page.

    (Select PeopleTools >Mobile Application Platform >Template Designer and click the Containers tab.)

2. In the Container Selectors section of the Template Designer – Containers page, add a selector:

    a. In the Selector Name field enter a name for the selector.

        Take care to enter the selector name using the appropriate naming convention for the selector type that you are defining.

        If you do not enter a name, the system uses the container class name and defaults the selector to a Class type when you save the definition.

    b. (Optional.) Select the Class Element box to designate the selector a CSS3 class selector.

    c. (Optional.) From the Theme drop-down list, select a theme to apply to the content rendered by the selector.

    d. (Optional.) In the Style box, enter CSS3 styles for the selector.

    e. In the Comments box, enter information about the selector.

3. In the Selector Properties section of the Template Designer – Containers page, add properties for the selector.

    a. In the Declare Prop field enter a CSS3 property.

    b. In the Declare Value field enter a value for the property.

    c. In the Comments field enter comments about the property and value.

    d. Click the Add a Row button (+) at the end of a row to add a row and define additional properties for the container.

4. Click the Save button.

To add additional container selectors click the Add a Row button (+) in the upper-right corner of the Container Selectors section heading to add a row.

Adding and Modifying Blob Properties

This topic describes how to add and modify blob properties.

Understanding Adding and Modifying Blob Properties

Consider the following CSS3 properties that you could define for, say, a standard icon in a template:

```
display: inline-block;
overflow: visible;
position: relative;
```

You can define these as a single selector property, called a *blob property*.

Adding and Modifying Blobs as Selector Properties

Use the Selector Properties Blob page (IB_DOCTPLBLOB_SEC) to add and modify blob properties.

Image: Selector Properties Blob page

This example illustrates the fields and controls on the Selector Properties Blob page. You can find definitions for the fields and controls later on this page.

In the Blob Data long edit box, enter selector properties.

After you define and save a block of styles as a blob, the blob appears as a single property/value pair in the Selector Properties grid in the Template Designer – Containers page (IB_DOCTPLBLOB_SEC).

Image: Selector Properties grid

In this example, the Selector Properties grid shows a blob defined as a selector property. The Declare Value field shows the individual properties that comprise the blob.
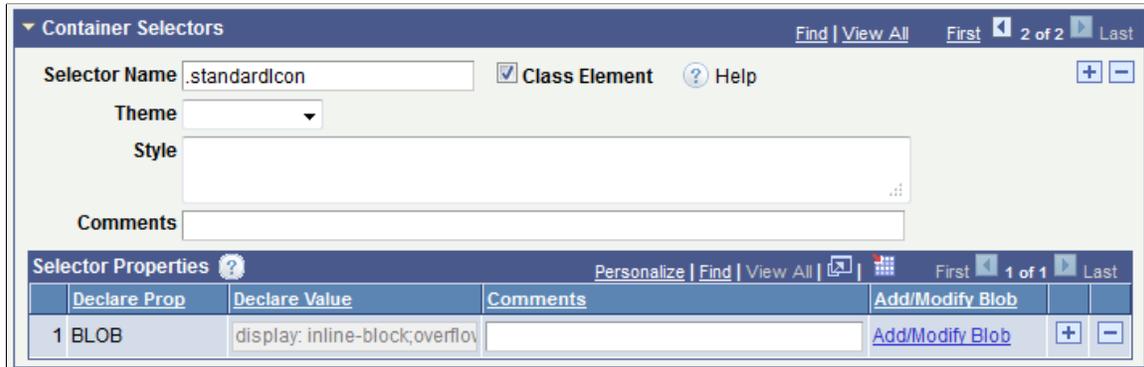
To add and modify blob properties:

1. Access the Template Designer – Containers page. (Select PeopleTools, Mobile Application Platform, Template Designer and click the Containers tab.)

2. In the Selector Properties grid click the Add/Modify Blob link.

   The Selector Properties Blob page appears.

3. In the Blob Data field enter and/or modify the properties in standard CSS3 format.

4. Click the OK button.

   The Containers page appears.

5. In the Selector Properties grid, enter comments about the blob property and its values in the Comments field.

6. Click the Save button.

Click the Add a Row button (+) to add a row and define additional properties.

Adding Container Elements

Use the Container Elements grid on the Template Designer – Containers page (IB_DOCTPLBLOB_SEC) to defined container elements, as shown in the following example:

Image: Container Elements grid on the Template Designer – Containers page

This example illustrates the Container Elements grid on the Template Designer – Containers page.
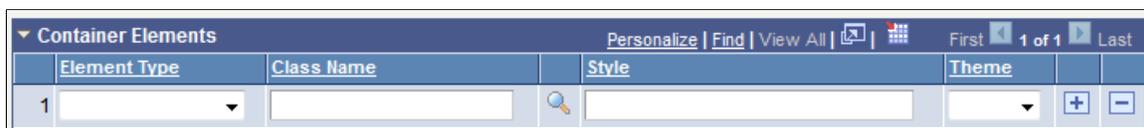
To define container elements:

1. Access the Template Designer – Containers page. (Select PeopleTools, Mobile Application Platform, Template Designer and click the Containers tab.)

2. Expand the Container Elements section if not already expanded.

3. From the Element Type drop-down list, select an element to add to the container.

4. (Optional.) Define a class for the element.

   a. In the Class Name field click the Look Up button. The Class Selection page appears and displays all of the classes defined for the container.

   b. Select the box next to the class to define for the element.

   c. Click the OK button to return to the Template Designer – Containers page.

5. (Optional.) In the Style field enter a CSS3 style to define for the element.

6. (Optional.) From the Theme drop-down list select a jQuery theme to define for the element.

7. Click the Save button.

Click the Add a Row button (+) to add a row and define additional elements for the container.

Viewing Container Metadata

To view container metadata:

1. Access the Template Designer – Containers page. (Select PeopleTools, Mobile Application Platform, Template Designer and click the Containers tab.)

2. In the Containers section of the page, click the View Container Summary link. The Container Summary page appears and displays summary information about the container.

Chapter 9

Adding Media Queries

Understanding Media Queries

Media queries are part of the CSS3 specification and define how styles are applied and how content renders on devices based on the media type and features that you define.

See <u>W3C Media Query Recommendations</u>

Used in conjunction with a media type, a media query is a logical expression, evaluating to true or false, that tests one or more features of an output device to which the CSS is to be applied. Therefore to render a mobile application on a certain device, you can simply use the containers that are currently defined for the template and which are applicable for specific devices, and modify the containers accordingly to get the desired rendering.

The Template Designer enables you to specify and build media queries for layout templates. To build a media query:

1. Build a media query string.

2. Assign containers to the media query.

3. Modify media query containers (as necessary) to achieve the preferred rendering on the device.

You can also code queries into CSS files using the Template Media Query page.

Using the Media Queries Page

To access the Media Queries page (IB_DOCTPLDEFN5) select PeopleTools >Mobile Application Platform >Template Designer and click the Media Queries tab. This example illustrates the fields and controls on the Media Queries page when no media query classes defined. You can find definitions for the fields and controls later on this page:

Image: Template Designer – Media Queries page

This example illustrates the fields and controls on the Template Designer – Media Queries page. You can find definitions for the fields and controls later on this page.
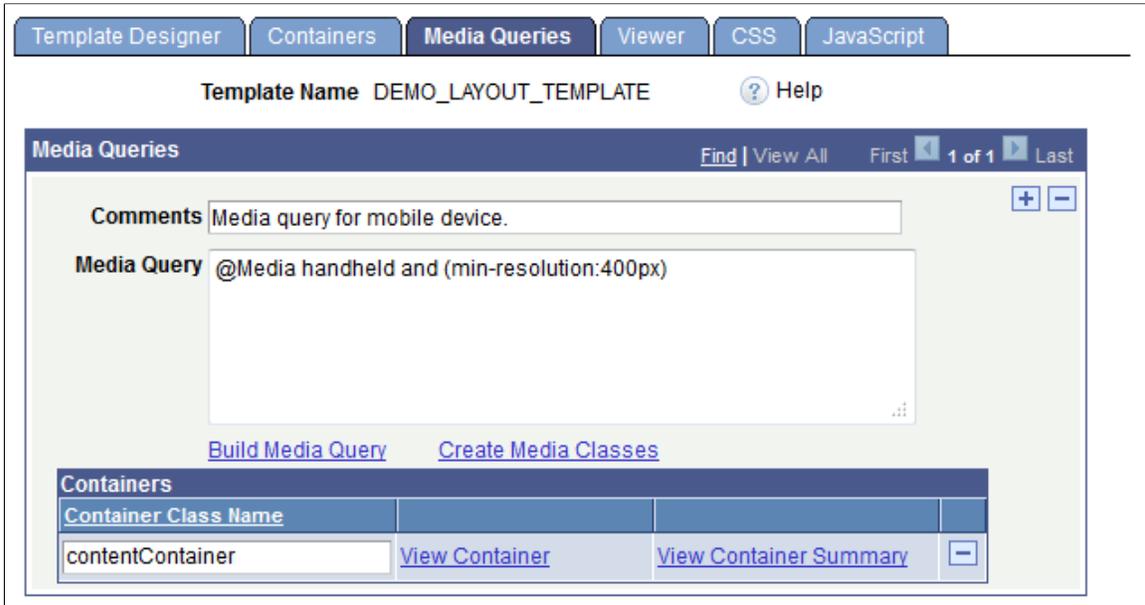
Note that the Containers grid appears on the page only when media query classes have been created for the template.

Comments	Enter comments to describe the media query.
Media Query	Displays the media query string. You can manually enter a media string in this field or click the Build Media Query link to build one.
Build Media Query	Click the link to access the Build Media Query String page to build a media query string.
Create Media Classes	Click the link to access the Select Classes for Media Query page to create media query classes for the template.
Container Class Name	Name of the container class created for the media query. This field appears only if one or more media classes have been created for the template.
View Container	Click the link to access the Media Query Class page where you can define aspects of the media class, including themes, styles, selectors, and so on. This link appears only if one or more media classes have been created for the template.
View Container Summary	Click the link to access the Container Summary page to view a textual representation of the container.

Using the Build Media Query String Page

Use the Build Media Query String page (IB_DOCLOTMPL_SEC) to build a media query. When you click the Generate media query text button, the system populates the Media Query field on the Media Query page with the string.

Image: Build Media Query String page

This example illustrates the fields and controls on the Build Media Query String page. You can find definitions for the fields and controls later on this page.

Logical Prefix	The valid media query keyword values are:
	• *Only.* Select this keyword to ensure that older browsers don't try to read the rest of the media query and ignore the linked style sheet altogether
	• *Not.* Select this prefix to negate the result of the media query.
Type	Indicates the media type to which the query applies. The valid values are:
	• *Braille.*
	• *all.* (Default.)
	• *aural.*
	• *handheld.*
	• *print.*
	• *projection.*
	• *screen.*
	• *tty.*

	• *tv.*
Conditional Separator	The valid values are:
	• *And.* Expresses a logical AND separator.
	• *Comma.* Expresses a logical OR separator.
	To specify that only one of the conditions needs to be true (similar to OR), separate the conditions with commas. Quantifiers can be added for all features except orientation, scan, and grid.
Quantity	Where allowed and desired, prefix a feature with one of the following values:
	• *min-.* Use this quantity to target a value that is greater than or equal to a value.
	• *max-.* Use this quantity to target a value that is smaller than or equal to a value.
Feature	From the drop-down list select the feature or features to include in the query. The valid values are:
	• *aspect ration.*
	• *color.*
	• *color-index.*
	• *device-aspect-ratio.*
	• *device-height.*
	• *device-width.*
	• *grid.*
	• *height.*
	• *monochrome.*
	• *orientation.*
	• *resolution.*
	• *scan.*
	• *width.*
Value	Enter a value for the selected feature.
Generate media query text	Click the button to generate a media query based on the information defined on the page.

The generated media query string appears under the button.

OK   Click the button to apply the generated media query to the template.

Using the Select Classes for Media Query Page

Use the Select Classes for Media Query page (IB_DOCCLASSEL_SEC) to select one or more containers, or container classes, to which to apply a media query.

The page displays all container classes defined for the template which have not yet been selected for the media query. To access the click the Create Media Classes link on the Template Designer – Media Queries page.

Image: Select Classes for Media Query page

This example illustrates the fields and controls on the Select Classes for Media Query page. You can find definitions for the fields and controls later on this page.

Select classes for media query

Classes	
Select	Container Class Name
☐	contentContainer
☐	footerContainer
☐	headerContainer
☐	mainContainer

Create Classes   Cancel

Select   Select the box to include the corresponding container class for the media query.

Container Class Name   Name of the container class.

Create Classes   After you select the container classes to include in the media query, click the button to create the classes.

Cancel   Click the button to exit the page without creating a media query classes.

Using the Media Query Class Page

To access the Media Query Class page (IB_DOCTPLDEFN_SEC) on the Template Designer – Media Queries page, in the Containers section click the View Summary link. The Containers section appears on the page only when you have defined containers classes for a media query.

Use the Media Query Class page to modify styles, selectors, selector properties, and container elements for a media query, without changing the base container.

Image: Media Query Class page

This example illustrates the fields and controls on the Media Query Class page. You can find definitions for the fields and controls later on this page.

The fields and controls on this page are the same as those that appear on the Template Designer – Containers page.

See Using the Containers Page for information about the fields that appear on this page.

Click the OK button to save changes to the page and return to the Template Designer – Media Queries page. Click the Cancel button to exit the page without saving any changes and return to the Template Designer – Media Queries page.

Building Media Query Strings

This topic describes the syntax to use when building media query strings for a Mobile Application Platform layout template. This topic also describes how to build media query strings for layout templates.

The PeopleSoft Mobile Application Platform supports the W3C recommendation for media queries.

Understanding Media Query String Syntax

The media query string syntax is:

```
@Media [Logic] Type [and ([Quantify]Feature: Value) ... [Conditional Separator]
([Quantify]Feature: Value)
```

An example of a media query string is:

```
@Media only handheld and (device-height:50px) and (max-device-width:100px)
```

Building Media Query Strings

This section describes using the Build Media Query Strings page to create a media query.

However, you can also manually code a media query by entering the string directly in the Media Query edit box in the Template Builder – Media Queries page.

To build a media query string:

1. Access the Build Media Query String page.

   a. Select PeopleTools >Mobile Application Platform >Template DesignerPeopleTools, Mobile Application Platform, Template Designer. The Template Designer page appears.

   b. Click the Media Queries tab. The Media Queries page appears.

   c. Click the Build Media Query link.

2. (Optional.) From the Logical Prefix drop-down list select a prefix.

3. From the Type drop-down list select the media type to which the query applies.

4. From the Conditional Separator drop-down list select a separator.

5. Add features to the query string. In the Feature List grid:

   a. From the Quantify drop-down list select the quantifier for the feature.

   b. From the Feature drop-down list select a feature.

   c. In the Value field enter a value.

6. To add more features, click the Add a Row button (+) and repeat Step 5.

7. Click the Generate Media Query Text button to preview the query string.

   The media string appears in the text box below the button. Change any parameters as needed to the string and click the Generate Media Query Text button again to view the changes.

8. Click the OK button.

9. The Media Queries page appears and the query string that you built appears in the Media Query edit box.

10. Click the Save button.

Click the Add a Row button (+) to add an additional media query to the template.

Click the Create Media Classes link to select one or more containers for the media query.

After you define a media query string, add the container or containers to which to apply the media query.

Related Links
Using the Build Media Query String Page
Adding Containers to Media Queries

Adding Containers to Media Queries

After you define a media query string, add the container or containers to which to apply the media query. After you add the containers you can modify them as necessary for the query.

To add containers to a media query:

1. Access the Select Classes for Media Query page.

    a. Select PeopleTools >Mobile Application Platform >Template Designer. The Template Designer page appears.

    b. Click the Media Queries tab. The Media Queries page appears.

    c. Click the Create Media Classes link.

        All the containers defined for the template and that are not yet assigned to the media query appear on the page.

2. Select the container(s) to add to the media query.

3. Click the Create Classes button.

4. Click the Save button.

The Media Queries page appears and the container(s) you selected appear in a Containers grid at the bottom of the page.

You can now modify the containers defined to a media query without impacting the base container definition.

Modifying Media Query Containers

Use the Media Query Class page to modify a container assigned to a media query. You can modify the container added to a media query without impacting the base container.

The controls and fields that appear on the page are the same as those that appear on the Template Designer – Containers page.

To access the Media Query Class page:

1. Select PeopleTools >Mobile Application Platform >Template Designer. The Template Designer page appears.

2. Click the Media Queries tab. The Media Queries page appears.

3. Click the Create Media Classes link.

Related Links
Using the Containers Page
Adding Selectors and Defining Selector Properties
Adding Container Elements

Viewing Media Query Container Metadata

The Container Summary page (IB_DOCTPLDEFN5) displays summary information about media query containers.

To access the Container Summary page:

1. Select PeopleTools >Mobile Application Platform >Template Designer. The Template Designer page appears.

2. Click the Media Queries tab. The Media Queries page appears.

3. In the Containers grid click the View Container Summary link.

Related Links
Using the Container Summary Page

Chapter 10

Documenting and Viewing Layout Template Structure

Understanding Documenting and Viewing Layout Template Structure

Use the Template Designer - Viewer page (PeopleTools, Mobile Application Platform, Template Designer and click the Viewer tab) to document the structure of a template and document any other information that could help a developer understand the template and determine whether to use it in a layout.

You can use HTML or plain text to document the structure of a template. If you use HTML to document a template you can also define HTML bind values to insert into your code. Note that no validation occurs on the page when you use HTML to document a layout template.

The information you document on the Viewer page is also accessible in read-only format on the Template page (PeopleTools, Mobile Application Platform, Template Designer and click the View Template Layout link).

Using the Viewer Page

To access the Viewer page (IB_DOCTPLDEFN4), select PeopleTools, Mobile Application Platform, Template Designer and click the Viewer tab.

Use the Viewer page to document the structure of a template. Developers can then use the information to evaluate using the template in their mobile applications. You can document the structure of the template in HTML or plain text. Note that no HTML validation occurs on this page.

The information you enter on this page appears in read-only mode on the Templates page (PeopleTools, Mobile Application Platform, Template Designer and click the View Template Layout link). If you describe the template on the Viewer page using HTML, it will render the HTML styles on the Template page.

Image: Template Designer – Viewer page

This example illustrates the fields and controls on the Template Designer – Viewer page. You can find definitions for the fields and controls later on this page.
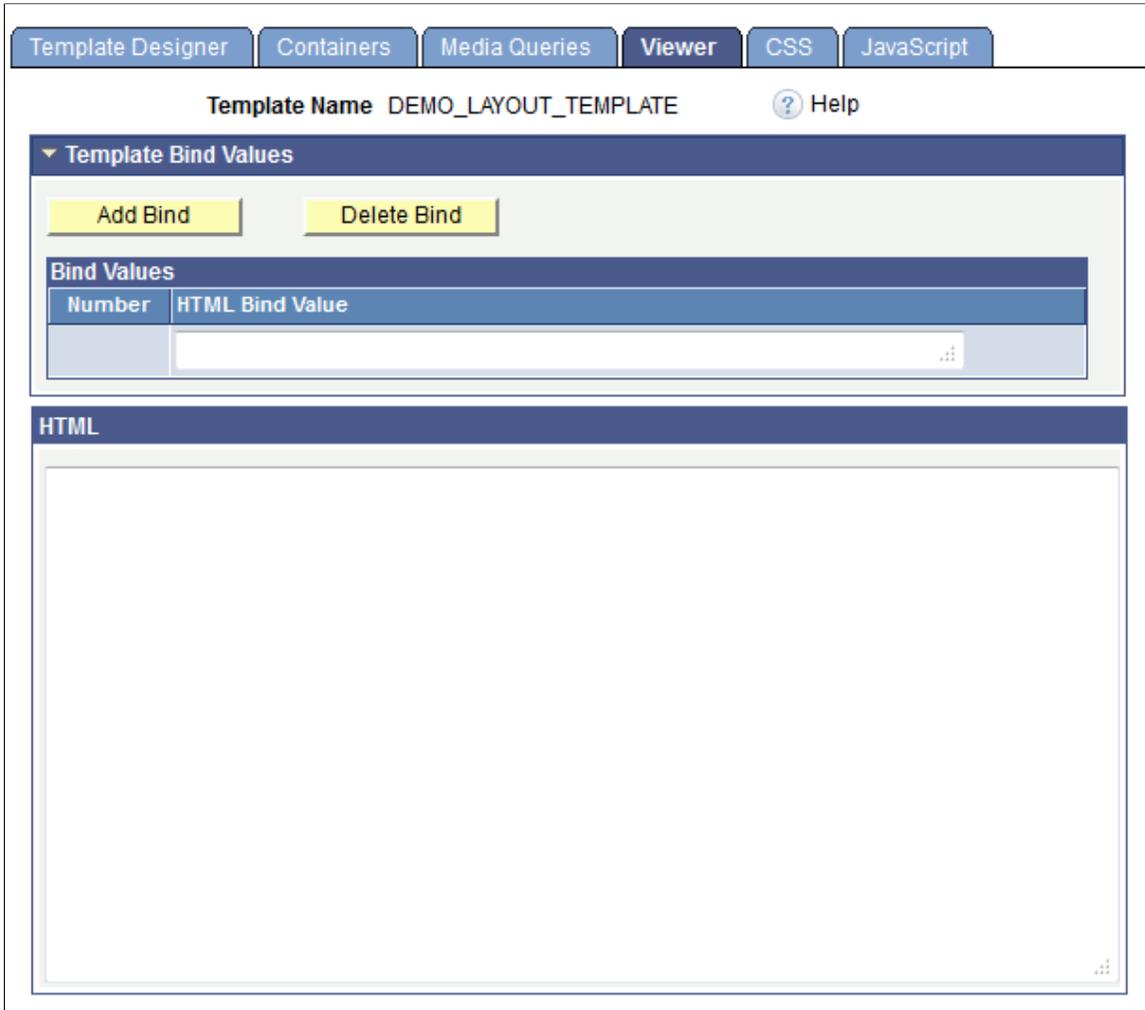

Add Bind	Click the button to add a row to the Bind Values grid to enter a bind variable.
Delete Bind	Insert the cursor in a row in the Bind Values grid and click the button to delete the entry.
Number	Number of the bind variable entry. The system assigns this value when you save the page.
HTML Bind Value	Enter a description for the bind.
HTML	Enter descriptive information about the template.
	The information you enter does not have to be HTML. Note, however, if you do enter HTML no validation occurs.

The following example shows the page with data defined.

Image: Template Designer – Viewer page

This example shows data defined in the Template Designer – Viewer page, including bind variables and HTML.

The data in the previous example renders as follows in the Template page for read-only viewing.

Image: Template page

This example shows the rendering of the data in the Template Designer – Viewer page shown in the previous example as it appears in the Template page.
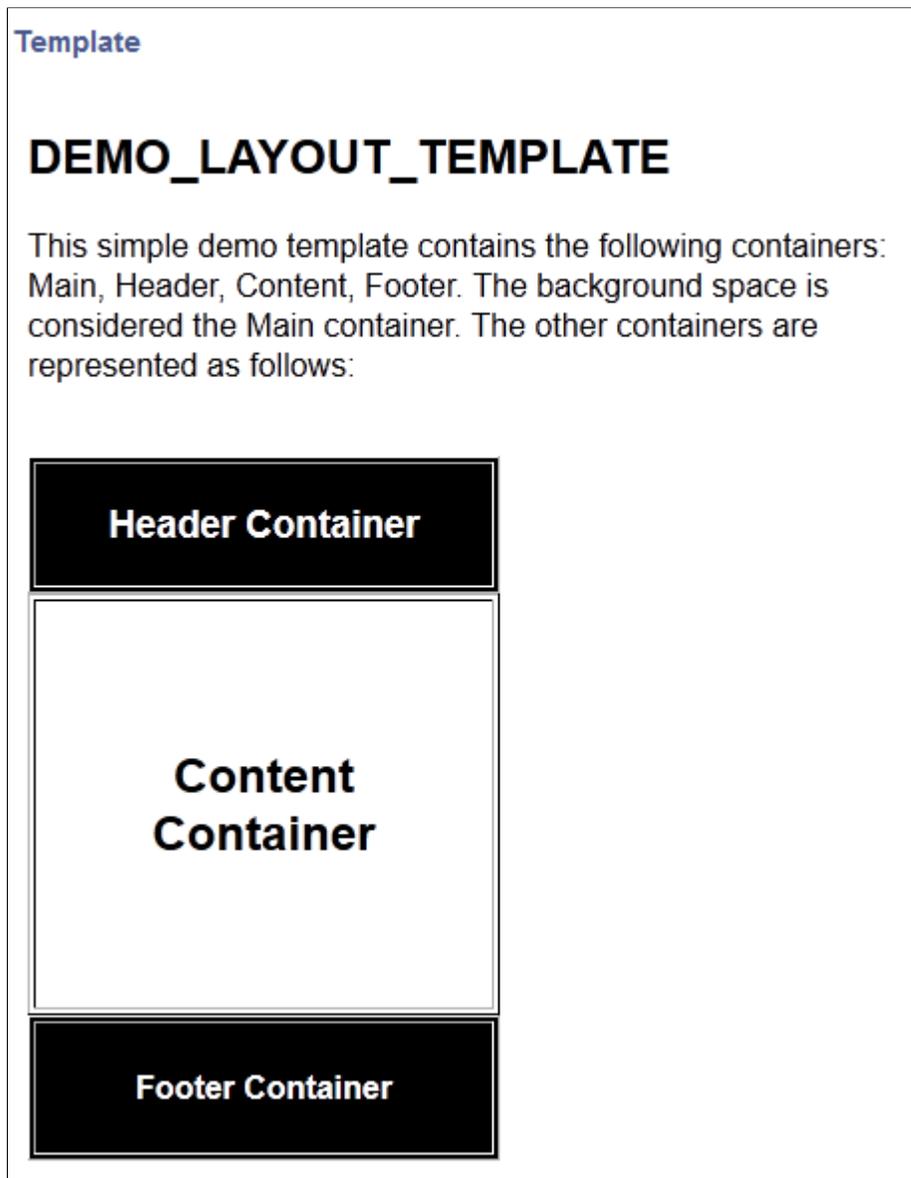
Documenting Layout Template Structure

To document the structure of a layout template, use the Template Designer – Viewer page. Use HTML or plain text to describe and document the structure of a template. You can also define and use HTML bind variables in HTML code on the page.

To document the structure of a layout template:

1. Access the Template Designer – Viewer page. (Select PeopleTools >Mobile Application Platform >Template Designer and click the Viewer tab.)

2. In the HTML edit box enter HTML or plain text to describe the template.

Note: No validation occurs on any HTML you enter on this page.

---

3. Click the Save button.

Using HTML Bind Values to Document Layout Template Structure

This topic discusses using HTML bind values to document layout template structure and describes how to:

• Add HTML bind values.

• Delete HTML bind values.

• Insert bind values into HTML.

Note that using HTML bind values is optional.

Adding HTML Bind Values

To add HTML bind values:

1. Access the Template Designer – Viewer page. (Select PeopleTools >Mobile Application Platform >Template Designer and click the Viewer tab.)

2. Locate and expand the Template Bind Values section.

3. In the HTML Bind Values field enter a bind value.

4. To add additional bind values:

   a. Click the Add Bind button to add another row to the grid.

   b. In the HTML Bind Values field enter a bind value.

5. Save the page.

When you save the page the system assigns a number to each bind value defined.

Deleting HTML Bind Values

To delete a bind value, insert the cursor into the HTML Bind Value field that you want to delete and click the Delete Bind button.

Inserting Bind Values into HTML

To insert bind values into HTML on the Template Designer – Viewer page, use the following format:

```
%bind(:<bind number>)
```

The <bind number> is the system-generated bind number assigned to a bind value on the Template Designer – Viewer page.

Insert the bind values between tags in your HTML code, as shown in this example:

```
<html>
<head>
<title> %bind(:1) </title>
</head>
<body>
<h2> %bind(:2) </h2>
```

Viewing Layout Template Structure

Use the Template page to view the structure of a layout template.

The information available and how the information appears are determined by the developer who documents the structure of the template.

To access the Template page select PeopleTools >Mobile Application Platform >Template Designer and click the View Template Layout link.

Related Links
Using the Template Page

Chapter 11

Managing Layout Template Cascading Style Sheets

Understanding Managing Layout Template Cascading Style Sheets

When you save a layout template definition in the Mobile Application Platform framework, the system generates a cascading style sheet from the definition metadata. You can also add additional CSS code and plug-ins to a template.

The framework provides a PIA-based viewer and editor for managing CSS code.

The system-generated CSS files and any CSS code you create in the Template Designer are saved to the following location:

```
<PIA_HOME>\webserv\<DOMAIN_NAME>\applications\peoplesoft\PSIGW.war\map\<NODE_NAME>\⇒
templates\<LAYOUT_TEMPLATE_NAME>\css
```

Note that depending on where you configured the integration gateway, the base folder could be a different location than the current PIA_HOME.

The system adheres to an order of precedence when applying CSS. See Cascading Style Sheets (CSS) for more information.

Using the Template Designer – CSS Page

When you save a template in the Template Designer, the system generates a Cascading Style Sheet (CSS) document based on the container and media query data you've defined in the template. You can view the generated CSS on the Template Designer – CSS page (IB_DOCTPLDEFN2). You can also add additional CSS code and plug-ins to a template using this page.

To access the page select PeopleTools >Mobile Application Platform >Template Designer and click the CSS tab.

Image: Template Designer – CSS page

This example illustrates the fields and controls on the Template Designer – CSS page. You can find definitions for the fields and controls later on this page.
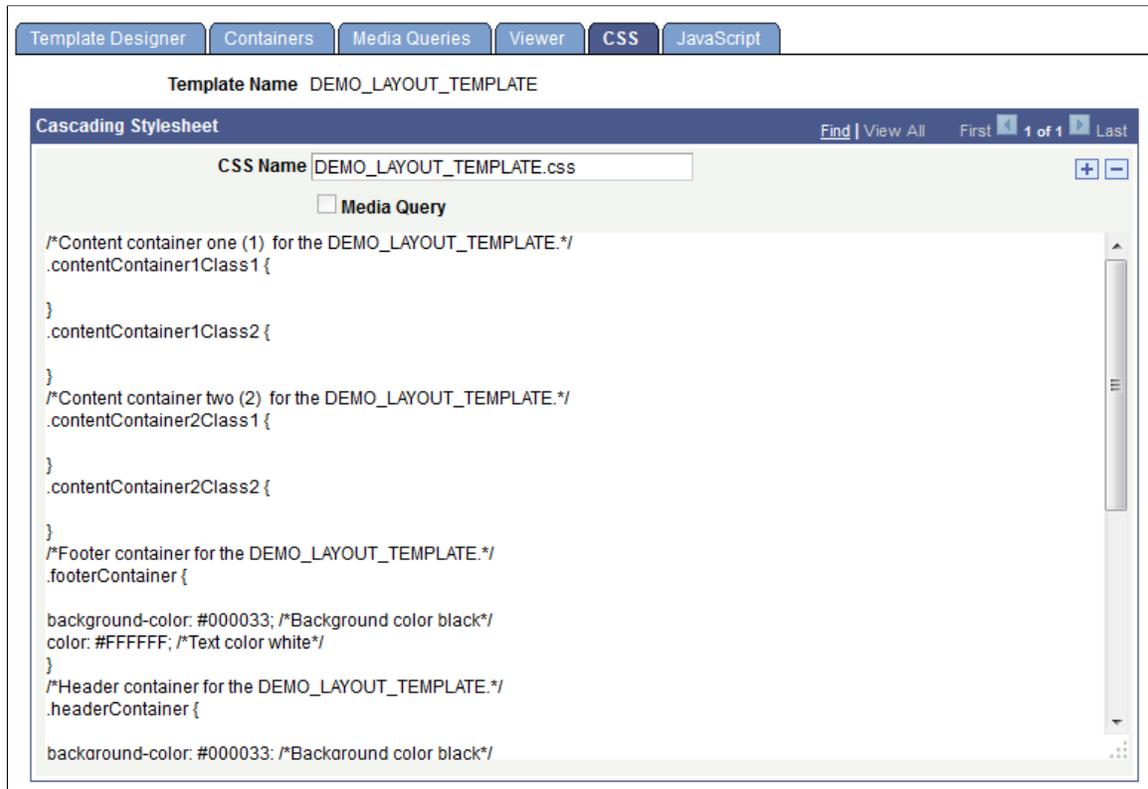
Template Name	This field displays the name of the template.
CSS Name	This field displays the name of the CSS file.
	The CSS file generated by the system takes the template name, but you can rename it.
	If you are entering additional CSS documents or plug-ins, enter a unique name.
Media Query	This field is read-only when you are viewing the system-generated CSS file.
	Check the box to enter a media query (in CSS3 format), instead of using the Media Queries page to build one.
	When you check the box the Add/View Media Query link appears. Click the link to access the Template Media Query page and to code a media query.
Add/View Media Query	Click the link to access the Template Media Query page to code a media query. This link appears only when you are view/editing a custom CSS file and you have checked the Media Query box.
Cascading Stylesheet	Enter the content of your CSS or plug-in in the long edit box.

Using the Template Media Query Page

Use the Template Media Query page (IB_DOCTPLMEDIA_SEC) to code media queries. This page provides an alternative to defining media queries using the Media Queries page. You can directly code media queries in CSS3 format on the page.

To access the page select PeopleTools >Mobile Application Platform >Template Designer and click the CSS tab. Check the Media Query box and then click the Add/View Media Query link.

Image: Template Media Query page

This example illustrates the Template Media Query page.

Use the long edit box to code your query. When using the Template Media Query page to code media queries, use the following CSS3 attribute:

```
link media=
```

An example of a query is:

```
<link media="handheld, only screen and (max-width: 480px), only screen and (max-dev⇒
ice-width: 480px)" rel="stylesheet" href="[CSS_FILE_NAME]">
```

Where CSS_FILE_NAME is the name of the CSS file name that appears in the CSS Name field. The framework will generate the proper URL reference to that CSS file. Click the OK button when you are done coding your query to return to the Template Designer – CSS page.

Viewing System-Generated CSS Code for Layout Templates

Use the Template Designer – CSS page (PeopleTools >Mobile Application Platform >Template Designer and click the CSS tab) to view system-generated CSS code for a layout template. The page provides a viewer to view the generated CSS file. The system uses the template name for the generated CSS file name. You can change the name as appropriate by entering a new name in the CSS Name field and saving the page.

The CSS code includes the following metadata:

- General metadata defined on the Template Designer page.

- Container, selector, and element metadata defined on the Template Designer - Containers page.

- Media query metadata defined on the Template Designer - Media Queries page and sub-pages.

Adding CSS Code to Layout Templates

To add CSS to a layout template:

1. Access the Template Designer – CSS page. (PeopleTools >Mobile Application Platform >Template Designer and click the CSS tab.)

2. Click the Add a Row button (+).

3. In the CSS Name field enter a name for the CSS file.

4. In the long edit box, enter the CSS code following the CSS3 specification.

5. Click the Save button.

Note: The system does not perform any CSS validation.

Adding Images to Layout Template CSS

PeopleSoft provides a built-in function for adding images to a layout template or a layout.

Saving Changes to Layout Template CSS

Each time click the Save button in the Template Designer the systems generates a CSS file based on the metadata defined in the template. The save process also attempt to update the web server with all CSS file(s).

If the integration gateway is down or if the process of updating the file is unsuccessful, the template metadata will still be saved and the appropriate error message appears. You can subsequently update the CSS changes on the web server in one of three ways:

- Re-save the template.

- In the Administration folder of the MAP Work Center, select the Layout File Synchronization link and perform a manual synchronization.

- If AutoSync is enabled on the Layout File Synchronization page then the update occurs on the next synchronization.

Related Links

Applying and Synchronizing Templates in Layouts

Chapter 12

Managing Layout Template JavaScript

Understanding Layout Template JavaScript

Any JavaScript you add to a template is stored in the following location:

```
<PIA_HOME>\webserv\<DOMAIN_NAME>\applications\peoplesoft\PSIGW.war\map\<NODE_NAME>\⇒
templates\<LAYOUT_TEMPLATE_NAME>\js
```

Note that depending on where you configured the integration gateway, the base folder could be a different location than the current PIA_HOME.

Using the Template Designer - JavaScript Page

Use the JavaScript page (IB_DOCTPLDEFN6) to stage JavaScript libraries and plug-ins for use in the layout.

To access the page select PeopleTools >Mobile Application Platform >Template Designer and click the JavaScript tab.

Image: Template Designer – JavaScript page

This example illustrates the fields and controls on the Template Designer – JavaScript page. You can find definitions for the fields and controls later on this page.
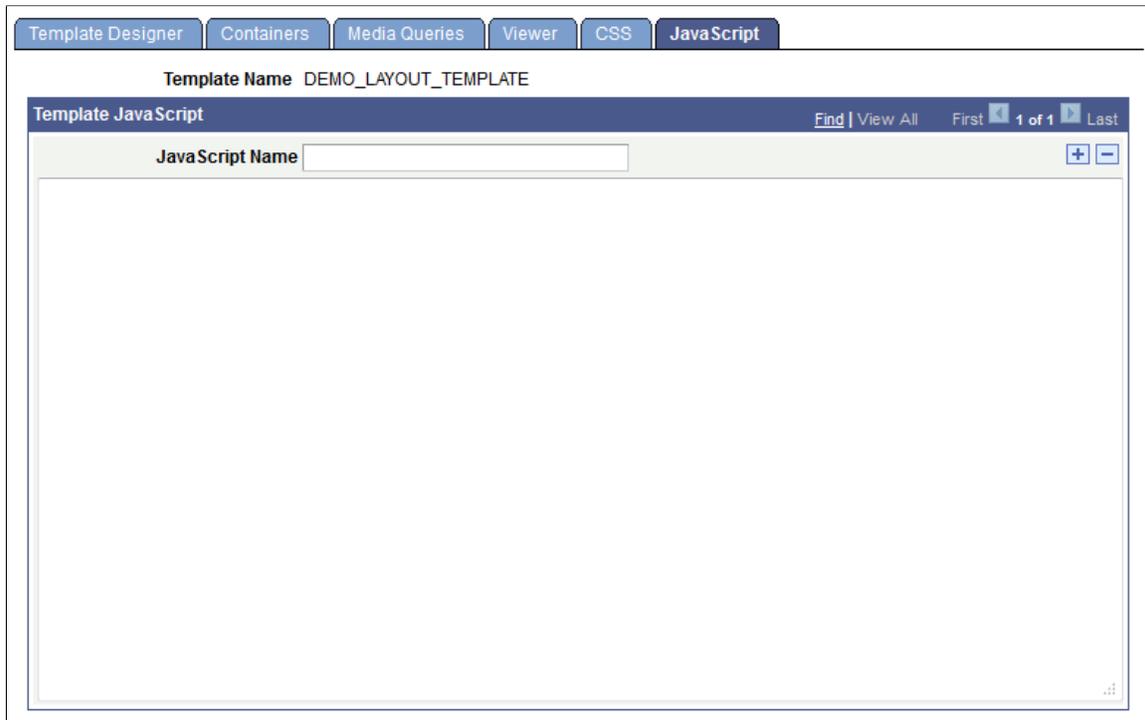
Template Name	Displays the name of the layout template.
JavaScript Name	Enter the name of the JavaScript, JavaScript library, or JavaScript plug-in.

Enter the contents of the script, library or plug-in in the long edit box.

Adding JavaScript to Layout Templates

To add JavaScript to a layout template:

1. Access the JavaScript page. (Select PeopleTools >Mobile Application Platform >Template Designer and click the JavaScript tab.

2. In the JavaScript Name field enter a name for the JavaScript name.

3. In the Template JavaScript long edit box, enter the JavaScript code.

4. Click the Save button.

Note that the system does not perform any JavaScript validation.

Saving Changes to Layout Template JavaScript

If the integration gateway is down or if the process of updating the file is unsuccessful, the template metadata will still be saved and the appropriate error message appears.

You can subsequently update the JavaScript changes on the web server in one of three ways:

- Re-save the template.

- Under the Administration folder of the MAP Work Center, select the Layout File Synchronization link and perform a manual synchronization.

- If AutoSync is enabled on the Layout File Synchronization page then the update occurs on the next synchronization.

Related Links
Applying and Synchronizing Templates in Layouts

Chapter 13

Managing Layout Sub-Templates

Adding Layout Sub-Templates

This topic describes how to add layout sub-templates to the database.

Understanding Adding Layout Sub-Templates

The minimum requirements for adding a sub-template to the database closely mirror those for adding a regular template, although there are some slight, but important, differences. To add a sub-template to the database you must:

- Name the layout sub-template.

- Designate the template a sub-template.

- Add a container to the sub-template.

- Add HTML to the sub-template.

You cannot save the template until you have provided the minimum information.

Unlike a regular layout template, you must specifically designate the definition a sub-template type. After you initially save a regular or sub-template definition you cannot change the template type. That is, you cannot change a regular template to a sub-template, or vice versa.

As with a regular layout template, you must create a container for a sub-template before you can save the definition. However, you cannot create a main container for a sub-template. The regular template to which you attach a sub-template already has a main container defined, and there can be only one main container in a template.

Note: This topic describes the minimum steps and information to enter to add a layout sub-template to the PeopleSoft database.

Related Links
Layout Templates and Sub-Templates

Naming and Designating Layout Sub-Templates

To name and designate a layout sub-template:

1. Select PeopleTools >Mobile Application Platform >Template Designer.

2. Click the Add Template tab.

3. In the Template Name field enter a name for the sub-template.

4. Click the Add button.

    The Template Designer page appears.

5. Select the Sub Template box.

6. In the Description field, enter a short description for the sub-template.

7. In the Long Description field, enter a more lengthy description.

    Proceed to the next section and add a container to the sub-template.

Related Links
Using the Template Designer Page

Adding Containers to Layout Sub-Templates

To add a container to a sub-template:

1. Navigate to the Template Designer – Containers page.

    Select PeopleTools >Mobile Application Platform >Template Designer and click the Containers tab.

2. In the Container Class Name field, enter a class name for the container.

    Note that the Main Container control is disabled. You cannot create a main container for a sub-template.

3. In the Comments box enter any comments about the container.

Proceed to the next section and add HTML to the sub-template.

Related Links
Using the Containers Page

Adding HTML to Layout Sub-Templates

To add a sub-template to the database you must enter some HTML in the HTML long edit box in the Template Designer – Viewer page.

This page is where you document information and details about the sub-template, so that other developers may understand the template contents and evaluate using it for their applications.

When you initially add a sub-template to the database you can add as little as opening and closing HTML tags in the HTML box. For example:

```
 <html>
</html>
```

To add HTML for a layout sub-template:

1. Navigate to the Template Designer – Viewer page.

    Select PeopleTools >Mobile Application Platform >Template Designer and click the Viewer tab.

2. In the HTML long edit box, enter some HTML related to your sub-template.

3. Click the Save button.

Related Links
Using the Viewer Page

Attaching Layout Sub-Templates

Use the Add Sub Template page to attach one or more sub-templates to a layout template.

The order in which you add layout sub-templates to a layout is important. The definitions on the last sub-template added to the database take precedence over other sub-templates should duplicate class names be used.

To attach a layout sub-template to a template:

1. Access the Add Sub Template page.

   Select PeopleTools >Mobile Application Platform >Template Designer and click Add Sub Templates link.

2. In the Sub Templates grid click the Lookup button to search for a sub-template to add.

3. Click the Add a Row button (+) and repeat Step 1 to add additional sub-templates.

4. Click the Save button.

5. Click the Go to Sub Template link to access the definition for a sub-template.

To view the definition for a sub-template, after search for a sub-template click the Go to Sub Template link in the grid.

Related Links
Using the Add Sub Templates Page

Chapter 14

Developing PeopleSoft Documents for Layouts

Understanding PeopleSoft Documents in the Mobile Application Platform

PeopleSoft Documents provide the data structure for mobile applications you build with the Mobile Application Platform framework.

PeopleSoft Pure Internet Architecture applications pass the complete business object to the client where the data must be serialized. However, since the mobile applications you build using the Mobile Application Platform use PeopleSoft documents for the business object definition, only the necessary data is passed to and serialized on the client.

The PeopleSoft Documents framework is documented in the Documents Technology.

Comparing PeopleSoft Document and PeopleSoft Rowset Structures

If you are familiar with developing applications for the PeopleSoft Pure Internet Architecture (PIA), the following comparisons may be helpful.

- In the PeopleSoft Pure Internet Architecture a rowset provides the data structure for a PeopleSoft Pure Internet Architecture page. In the Mobile Application Platform, a PeopleSoft document provides the data structure for your mobile application pages contained in a layout.

- A parent PeopleSoft document roughly equates to a Level 0 record in the PeopleSoft Pure Internet Architecture framework. And document compound and collection elements roughly equate to Level 1 records in the PeopleSoft Pure Internet Architecture framework.

Documents allow multi-level objects to be flattened down to the equivalent of Level 1 records because there is no key dependency needed between the levels. The actual keys for a row of data are part of the application URL created. Any processing of the data is then performed on the web server.

The following examples show a three-level typical rowset keyed structure, flattened to an equivalent PeopleSoft document structure.

Image: Three-level rowset keyed structure

This example illustrates a typical three-level rowset keyed structure:

In the example the Level 0 record is *QE_FLIGHTDATA,* the Level 1 record is *QE_ARMAMENT,* the Level 2 record is *QE_WPN_PRESETS,* and the Level 3 record is *QE_STAT_CONFIG.*

Image: PeopleSoft document equivalent of the three-level rowset structure

This example shows the same structure defined in a PeopleSoft document:
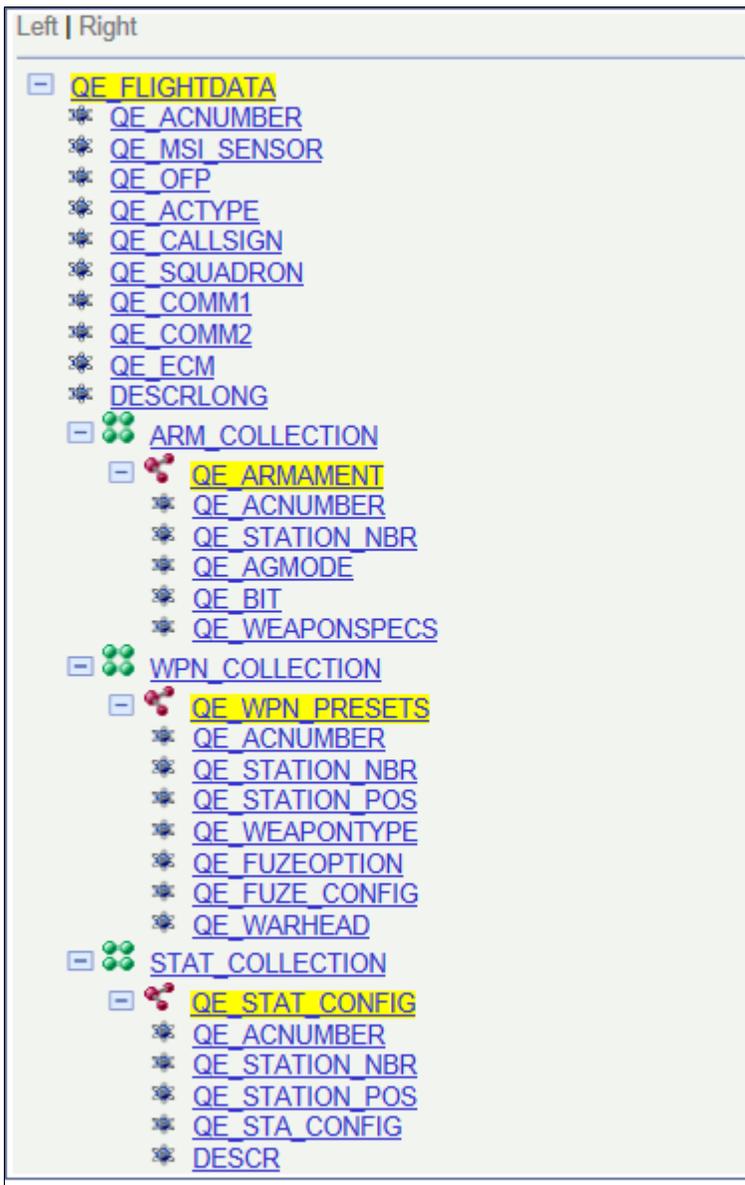
In this example *QE_FLIGHTDATA* is equal to a Level 0/root record. *QE_ARMAMENT, QE_WPN_PRESETS,* and *QE_STAT_CONFIG* are equivalent to Level 1 records. The following icons appear in the document structure:

Icon	Description
	Primitive document element.
	Collection document element.
	Compound document element.

In the PeopleSoft documents terminology, the previous example shows *QE_FLIGHTDATA* is the root element, that has ten (10) primitive elements defined (*QE_ACNUMBER, QE_MSI_SENSOR, QE_OFP,* and so on.)

ARM_COLLECTION is a compound child element of the *QE_FLIGHTDATA* element. Within that the compound element, a collection child element called *QE_ARMAMENT* is defined. And QE_ACNUMBER, QE_STATIONNBR, QE_AGMODE, and so on, are primitive child elements of the *QE_ARMAMENT* compound element.

Similarly, *WPN_COLLECTION* and *STAT_COLLECTION* are collection child elements of *QE_FLIGHTDATA*. Each of those collection elements has a child compound element defined, *QE_WPN_PRESETS* and *QE_STAT_CONFIG* respectively. And each of those compound elements has several child primitives defined.

Developing PeopleSoft Documents for Layouts

This topic discusses how to:

- Map layout page elements to PeopleSoft document elements.

- Develop document for layouts.

This topic also discusses development considerations for building PeopleSoft documents for layout.

Mapping Layout Page Elements to PeopleSoft Document Elements

When creating documents for Mobile Application Platform applications you should be aware of the layout page elements that you can map to elements in the PeopleSoft document linked to the layout.

The following table lists the types of layout page elements that you can map to PeopleSoft document elements:

Layout Page Element	PeopleSoft Document Element
• Static Text. • Image Static Text. • URL. • Flip Toggle. • Push Button. • Input. • Radio Button. • Check Box. • Slider. • HTML Area.	Primitive, compound (not wrapped in a collection)

Layout Page Element	*PeopleSoft Document Element*
• Table. • (Dynamic) List View	Collection
• Select Box. • Menu.	Primitive wrapped in a Collection

Consider the following additional points relative to mapping layout page elements to PeopleSoft document elements:

- Compound elements and collection elements must contain only primitive elements. The exception is for the List View page element, where you can define collection elements that contain compound elements.

- The Select Box and Menu page elements can consist of primitives or primitives wrapped in a collection. Define primitives wrapped in a collection if the user is to choose multiple values/options in the select box or menu.

- Primitive elements to map to Radio Button page elements and Check Box page elements must be defined as Boolean data types in PeopleSoft documents.

Developing Documents for Layouts

This section discusses developing documents for layouts.

Developing New Documents

Use the Document Builder page (IB_LOGICALSCHEMA) in the Documents component (IB_LOGICALSCHEMA) to create PeopleSoft documents for layouts. To access the page select PeopleTools >Documents >Document Builder.

Developing Documents from PeopleSoft Records

In addition to developing PeopleSoft documents from the ground up, you can also create documents from existing PeopleSoft records. You can use a document created from a record by itself as a document to link to a layout. You can also define the record-turned-document as a compound child element in a document.

Use the Create Document from Record page (IB_DOCUMENT_CRREC) in the Documents Utilities component (IB_DOCUMENT_CRREC) to create a PeopleSoft document from a PeopleSoft record. To access the page select PeopleTools >Documents >Document Utilities >Create Document from Record.

Related Links
"Adding Document Definitions" (PeopleTools 8.57: Documents Technology)
"Creating Documents from Records" (PeopleTools 8.57: Documents Technology)

Development Considerations for Building PeopleSoft Documents for Layouts

Consider the following points when developing PeopleSoft documents for Mobile Application Platform applications:

• Valid documents for layouts can only be one-level deep.

• Use collections elements to use multiple rows of data.

• Documents typically contain child documents. However, since the documents you link to a layout can be only one level deep, child documents cannot contain compound elements or collection elements.

Modifying PeopleSoft Documents Linked to Layouts

After linking a PeopleSoft document to a layout, you can modify it in the Document Builder, as long as the document isn't restricted or in use in the runtime tables.

To access the Document Builder select PeopleTools >Documents >Document Builder.

To ensure that you are able to modify a document, write-restrict the document as described later in this topic.

Using Existing PeopleSoft Documents in Layouts

You can link an existing PeopleSoft document to a layout.

However, if a document is used by other metadata in the system, there is a chance that you will not be able to modify the document. For example, if a document is write-restricted or in used by a message in the runtime tables, you cannot modify the document.

If you think you will need to modify a document the best course of action is to make a copy of the document and then write-restrict the document in the Mobile Application Platform component or in the Documents component.

Restricting PeopleSoft Documents Used in Layouts

Restricting PeopleSoft Documents Used in Layouts You can write-restrict a PeopleSoft document in the Mobile Application Platform component or in the Document component.

To restrict a PeopleSoft document in the Mobile Application Platform component use the Restricted Layout page (IB_DOCLORES). To access the page select PeopleTools >Mobile Application Platform >MAP Utilities >Layout Status and click the Restricted Layouts tab.

To restrict a PeopleSoft document in the Documents component use the Restricted Documents page (IB_DOCUMENT_RSTRT). Select PeopleTools >Documents >Documentation Administration >Restricted Documents.

Chapter 15

Using the Layout Designer

Prerequisites for Using the Layout Designer

Before you can use the Layout Designer you must:

1. Configure the integration gateway.

   The framework uses the integration gateway to communicate between mobile applications and the database.

   This includes setting the gateway URL, loading delivered target connectors, setting Oracle Jolt connection properties, and setting the Java keystore password.

   See "Administering Integration Gateways" (PeopleTools 8.57: Integration Broker Administration)"Setting Oracle Jolt Connection Properties" (PeopleTools 8.57: Integration Broker Administration)"Configuring Security and General Properties" (PeopleTools 8.57: Integration Broker Administration)

2. Set the web service target location *and* REST service target location.

   See "Setting Target Locations for Services" (PeopleTools 8.57: Integration Broker Administration)"Setting Target Locations for REST Services" (PeopleTools 8.57: Integration Broker Administration)

Navigating the Layout Designer

The Mobile Application Platform features a Layout Designer for building mobile applications.

Using the Layout Designer you:

- Add mobile application pages and set page properties, such as styles, themes, transitions to other application pages, caching parameters, and so on.

- Add elements to mobile application pages, such as text fields, buttons, action sheets, tables, and so on.

- Define properties for page elements, including display options, input properties, image properties, and more.

- Add cascading style sheets.

- Define media queries.

- Add JavaScript and Ajax events.

- Set security parameters.

The Layout Designer is located in the IB_DOCTPLDEFN component in the PeopleSoft Pure Internet Architecture. To access the Layout Designer select PeopleTools >Mobile Application Platform >Layout Designer.

Image: Layout Designer – Layout page

This example illustrates the default view of the Mobile Application Platform Layout Designer:
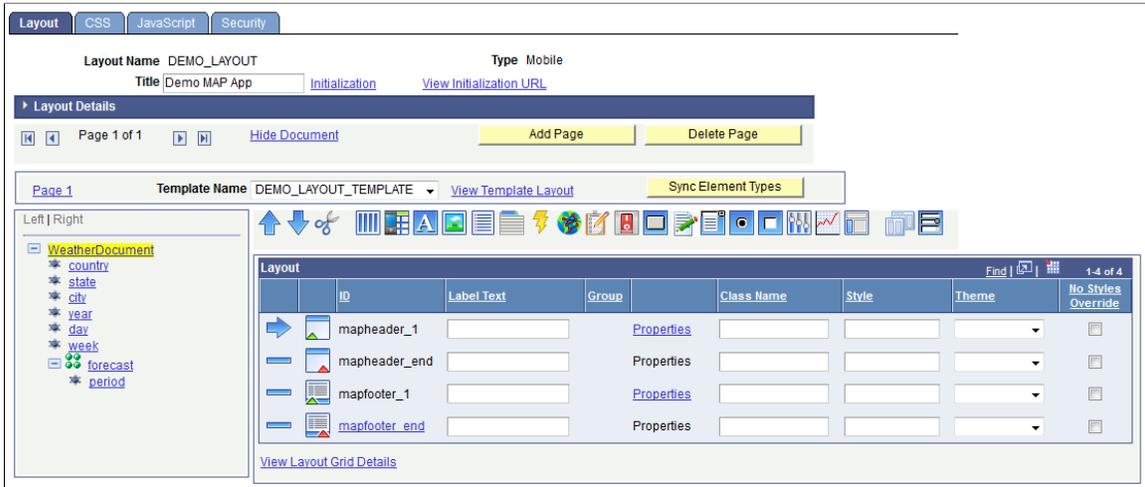
The Layout Designer features four pages. The following table lists and briefly describes the pages how to access them:

Page	Object ID	Description	Navigation
Layout	IB_DOCLODEFN	Use the page to: • Add pages to layouts. • Add elements to pages. • Define element properties. • Attach MAP PeopleCode to your application. • View the URL to access the application.	Select PeopleTools >Mobile Application Platform >Layout Designer.
CSS	IB_DOCLOCSS	Use the page to: • Add cascading styles sheets to a layout. • Add custom themes. • Define media queries.	Select PeopleTools >Mobile Application Platform >Layout Designer and click the CSS tab.

Page	Object ID	Description	Navigation
JavaScript	IB_DOCLOJS	Use the page to: • Add JavaScript to a layout. • Add jQuery Mobile events to a layout.	Select PeopleTools >Mobile Application Platform >Layout Designer and click the JavaScript tab.
Security	IB_DOCLOSECURE	Use the page to: • Define authentication for a layout. • Set system status for a layout. • Grant permission list access to a layout.	Select PeopleTools >Mobile Application Platform >Layout Designer and click the Security tab.

93

Chapter 16

Adding Layouts

Understanding Adding Layouts

This topic provides information about adding layout definitions to the system.

Layout Data Structure and PeopleSoft Documents

The data structure for a mobile application that you build in the Mobile Application Platform must be defined in a PeopleSoft document. When you add a layout to the system, you must specify the PeopleSoft document to link to the layout. Therefore you must create the PeopleSoft document that you want to link to a layout before you add the layout to the system.

If attempt to link an existing PeopleSoft document to a layout and it is used by other metadata anywhere in the database, the system displays a message indicating so. If you link an existing PeopleSoft document to a layout and the document is used by other metadata, there is a chance that the document cannot be modified, depending on its state. If you believe that modifications will be made to the document, it's better to make a copy of the document thereby making the layout the only owner and consider restricting write-access to the document. You can restrict write access to the document in the document definition or in the layout definition.

Layout Metadata Created

The Mobile Application Platform uses REST services as the communication layer between applications built using the Mobile Application Platform and the database.

You do not need any knowledge of REST services and you do not need to develop or administer REST services to use the framework. However, the framework does provide the capability to call other REST services for those with knowledge of REST services or business requirements for using REST services in their applications.

The information in this section is provided for those who are familiar with REST services or who have an interest in understanding the metadata the system uses.

When you add a layout to the database the system creates the following metadata:

Metadata	Name	Description
REST service	`<LAYOUT_NAME>1_MAP`	System –generated read-only REST service that the application uses to communicate with the database. The name of the REST service created for a layout appears in the following locations: • Layout Details section of the Layout Designer – Layout page. • Layout Designer – Security page. You can also view the service in the Service page by selecting PeopleTools >Integration Broker >Integration Setup >Services.
REST service operations	• `<LAYOUT_NAME>1_MAP_GET` • `<LAYOUT_NAME>1_MAP_POST`	System-generated read-only REST service operations that the application uses to communicate with the database. These service operations belong to the <LAYOUT_NAME>1_MAP REST service. You can view the service operations in the Service page, by selecting PeopleTools >Integration Broker >Integration Setup >Services and selecting the service to which the service operations belong.
Document template	`LAYOUT_NAME>1_MAP.map`	The system generates a read-only document template, which is a Document type message based on the PeopleSoft document that you link to the layout. The document message has primitive elements defined with names used for value replacement within the URI template. This document can only contain primitive elements. The document cannot contain any imports (compounds). The advantage of using documents is that they cannot be inadvertently changed, as is not the case with defined record and/or fields.

Metadata	*Name*	*Description*
Messages	• IB_GENERIC_REST.V1 • IB_REST_STUB.IB_REST_STUB	PeopleTools delivers a restricted read-only message for each of the REST service operations. The GET service operation uses the PeopleTools delivered and restricted message IB_GENERIC_REST.V1 message. The POST service operation uses the PeopleTools –delivered and restricted message IB_REST_STUB.IB_REST.STUB. You view these messages from the service or service operation definition. See details for *REST services* and *REST service operations* in this table for information on accessing these definitions.
URI document	NA	Use this document to build URIs/map to specific resources in a document. You can access and view this document using the URI Document link in the Layout Initialization section of the Initialization page (PeopleTools >Mobile Application Platform >Layout Designer and click the Initialization link)

Prerequisites for Adding Layouts

Before you can add a layout to the system you must create the data structure for the layout in a PeopleSoft document.

See Developing PeopleSoft Documents for Layouts

Adding Layout Definitions

To add a new MAP Layout:

1. Navigate to PeopleTools >Mobile Application Platform >Layout Designer. Click the Add a New Value tab.

2. Enter the Layout Name and click the Add button.

   Note: Leave the Custom Layout checkbox cleared.

3. Use the Add Layout page (IB_DOCLODEFNSRCH) to add a new layout definition to the system.

Image: Add Layout page

This example illustrates the fields and controls on the Add Layout page. You can find definitions for the fields and controls later on this page.

Layout Name	Layout Name
Package	Enter or select the name of the document package that contains the PeopleSoft document to link to the layout.
Document	Enter or select the name of the PeopleSoft document to link to the layout.
Version	Enter or select the version of the PeopleSoft document to link to the layout.
Template Name	Enter or select the name of the layout template to apply to the layout.
Selected Document	Click the link to view the PeopleSoft document you specified in the Package, Document, and Version fields in the Document Builder.
Include PeopleSoft Header	Check the box to include the header from the PeopleSoft fluid interface as the header for the application.

When the PeopleSoft Header is selected, it will automatically include the PT_FLUID_TEMPLATE in your MAP application. The PT_FLUID_TEMPLATE is used to get FLUID styles for MAP applications. |

| Link Document to Layout | Click the button to add the layout to the database and generate the layout metadata. |

To add a layout definition to the system:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

   The Layout Designer search page appears.

2. Click the Add a New Value tab.

3. In the *Layout Name* field, enter a name for the layout.

4. Click the *Add* button.

   The Layout Designer page appears.

5. In the *Package* field, enter the name of the package that contains the PeopleSoft document to link to the layout.

6. In the Document field, enter the name of the PeopleSoft document to link to the layout.

7. In the Version field, enter the version of the PeopleSoft document to link to the layout.

8. (Optional.) In the Template Name field, enter the name of the layout template to apply to the layout.

9. (Optional.) Select the Include the PeopleSoft Header to include that header in the layout.

10. Click the Link Document to Layout link.

    The Layout Designer page appears.

11. Click the OK button to create the layout and generate the layout metadata.

The Layout Designer – Layout page appears.

Viewing and Defining Metadata and Details

View and define layout metadata and details in the top portion of the Layout Designer – Layout page.

Image: Layout Designer – Layout page

This example illustrates the fields and controls on the Layout Designer – Layout page. You can find definitions for the fields and controls later on this page.

The Layout Details section is collapsed on the page by default. Click the arrow button to expand the section.

The table lists the metadata fields and controls that appear on the Layout Designer – Layout page:

Layout Name   Displays the name of the layout.

Type   Displays the type of layout. Currently, Mobile is the only
                                        supported layout type.

Title   Enter the title of the application as you wish it to appear in the
                                        browser title bar. This value defaults to the layout name.

Initialization   Click the link to access the Initialization page to set the
                                        following:

                                        •   Base event method.

                                        •   Server cache options.

                                        •   Initialization parameters.

View Initialization URL   Click the link to display the URL for the mobile application that
                                        you are developing.

                                        Copy the URL and paste it in the browser URL field to view and
                                        test the application.

This table lists the fields and controls that appear in the Layout Details section of the Layout Designer –
Layout page:

Package   Displays the package name that contains the PeopleSoft
                                        document linked to the layout.

Document   Displays the name of the PeopleSoft document linked to the
                                        layout.

Version   Displays the version of the PeopleSoft document linked to the
                                        layout.

Description   Enter a description of the layout.

Details   Enter a more lengthy description of the layout.

Service   Displays the name of the system-generated REST service for the
                                        layout.

Related Actions   Click the link to access the Related Actions page to override the
                                        default Class, Style, Popup Class and Image to use for defined
                                        Related Actions on the MAP application.

Group   Enter a group name to group layouts together. Layout search
                                        pages can search by group name to show a logical grouping of
                                        layouts.

Owner ID   From the drop-down list, select an owner for the layout.

Validation   From the drop-down list select the type of data input validation
                                        for the layout. The options are:

- *Custom Validation*

  The system performs no input validation. A JavaScript validation file is provided for implementing custom validation.

- *No Validation*

  The system performs no input validation and a JavaScript validation file is not provided.

- *Standard Validation* (Default.)

  The framework performs standard data input validation.

  Access the JavaScript validation file in the application HTML source:

  ```
  <script type="text/javascript" src="/PSIGW/map/defaults⇒
  /js/jquery.validate.js">
  </script>
  ```

  Or navigate to the following path to access the validation file:

  ```
  <PIA_HOME>\webserv\<DOMAIN_NAME>\applications\peoplesof⇒
  t\PSIGW.war\map\defaults\js\
  jquery.validate.js
  ```

  Note that depending on where you configured the integration gateway, the base folder could be a different location than the current PIA_HOME.

Raphael	Select the box to add links to the application HTML source for Raphael JavaScript libraries for charting.
Document Dump	Select the box to generate in JSON the PeopleSoft document linked to the layout. The system adds the JSON to the header in the application source HTML as a variable.
Bookmark	Select the box to add Google JavaScript links to the application HTML source to create bookmarks.
Retain State	Select the box to save the state of the PeopleSoft document data on the initialization event.
Touch Image	If you are implementing an application for the Apple framework, select an image for the tile.
Exception Layout Name	Enter the Exception Layout Name. An exception layout is simply another layout. If an exception occurs the MAP framework will invoke the defined Exception Layout. The developer can get access to the exception information thrown via People Code.

Exception Layout

An exception layout is created the same as any other MAP layout. When an exception occurs in the MAP application, the OnInit event for the exception layout is invoked The methods/ properties available are the same for the standard Exception Object.

This is an example of code for the OnInit event for the exception layout.

```
&Exception = &Map.MAPException;

To get the Layout name that actually threw the exception:
&Map.ExceptionLayoutName;

To retrieve the Document information at the time of the thrown exception:
&ExceptionDocument = &Map.GetExceptionDocument();

To retrieve the URI Document information at the time of the thrown exception:
&ExceptionURIDocument = &Map.GetExceptionURIDocument();
```

See "Understanding Exception Class" (PeopleTools 8.57: PeopleCode API Reference)

This table summaries how the exception is handled based on the event.

Event	*How exception is handled*
Base Event: (OnInit) and Update Event(s)	Will always invoke the Exception Layout
Save Event	Will not invoke the Exception Layout
AJAX Event	By default will not invoke the Exception Layout. However, the developer can override this by setting the *Use Exception Layout* property See Defining Ajax Event Page Element Properties

Note: It will be up to the developer to properly handle the HTML that will now be returned to the client as a result of the Exception Layout being thrown.

Chapter 17

Defining Base Event Methods for Layouts

Understanding Base Event Methods for Layouts

You can specify a base event method to run every time the application URL is invoked.

The PeopleCode method to use for layout initialization is the Initialization event, OnInitEvent. Use the event to populate data or perform any other processing logic.

Related Links
Understanding PeopleCode for the Mobile Application Platform

Defining Base Event Methods

Use the Base Event Method section on the Initialization page (IB_DOCBASEVNT_SEC) to define the base event method for a layout. To access the page select PeopleTools >Mobile Application Platform >Layout Designer and click the Initialization link next to the *Title* field at the top of the page.

Image: Base Event Method section of the Initialization page

This example illustrates the fields and controls on the Base Event Method section of the Initialization page. You can find definitions for the fields and controls later on this page.

Root Package ID	Enter the name of the application package that contains the method to invoke.
Path	Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.
Class ID	Enter the name of the application class that contains the method to invoke.

Method	From the drop-down list select the method to invoke.
Pending Approval	This field is currently not used and is reserved for future use.

Chapter 18

Defining Layout Initialization Parameters

Understanding Layout Initialization

This section discusses default layout initialization in the Mobile Application Platform, as well as defining additional initialization parameters and values.

Default Layout Initialization

The Initialization page (PeopleTools >Mobile Application Platform >Layout Designer and click the Initialization link) features a Layout Initialization section where you define initialization parameters.

Image: Layout Initialization section of the Initialization page

This example illustrates the fields and controls on the Layout Initialization section of the Initialization page.

By default, there is always one row which defines no parameters, as shown in the previous example. You'll note the number *1* located to the right of the Description field. This is an index number, called a *URI index,* which the system assigns to each initialization parameter defined in the system.

By default the system passes the page ID and layout ID and you can view these values on the URI Document page (IB_DOCDISPLAY_SEC) by clicking the URI Document link.

Image: URI Document page

This example shows the default view of the URI Document page.
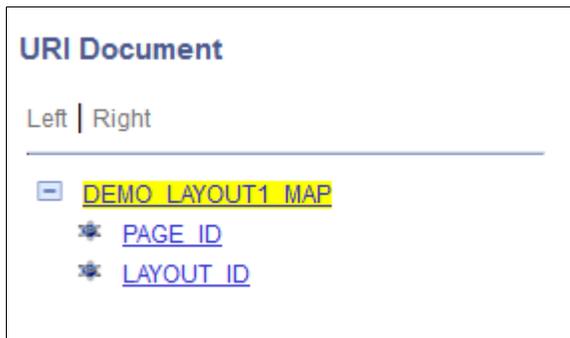
The system builds URLs from the initialization parameters by appending the initialization index to the base URL. The base URL is defined by the REST target location that you set as a prerequisite for using the Layout Designer. For example:

```
http://<machinename>:<port>//PSIGW/RESTListeningConnector/<NODE_NAME>/
<LAYOUT_NAME.VERSION>/INIT/URI_index
```

When you click the View Initialization URL, the system generates a URL based on the default initialization parameter shown in the first example:

```
http://<machinename>:<port>//PSIGW/RESTListeningConnector/<NODE-NAME>/
DEMO_LAYOUT1_MAP.v1/INIT/-1
```

Initialization Parameters and Values

You can define additional parameters to pass in one or more URIs. The system displays the URIs that you define when you are working with page elements that allow URLs to be defined as properties, including URL, push button, and list view page elements.

By default no arguments are passed and you can examine the user ID to get context for the request to populate the Initialization event. However, adding additional parameters to the URL allows the Initialization event to use these to pass back data with respect to the parameters.

For example, if an additional parameter of *Purchase_Order* is passed, the Initialization event can read the purchase order and use that to populate the return document with additional information. The passing of additional parameters allows the layout to generate different context information.

If you are familiar with REST services, defining initialization parameters is similar to building URI templates.

When you define an initialization parameter you specify variables/primitive elements from the PeopleSoft document linked to the layout as parameter values. The parameter values are used to construct URLs at runtime.

When you build initialization parameters the system assigns a sequential integer index value to each parameter. You can use these URI indexes in PeopleCode to call and manipulate content.

Using the Initialization Page for Defining Layout Initialization Parameters and Values

This topic describes using the Initialization page and sub-pages to define layout initialization parameters and values.

Using the Layout Initialization Section of the Initialization Page

Use the fields and controls in the Layout Initialization section on the Initialization page to define layout initialization parameters. To access the Initialization page, select PeopleTools >Mobile Application Platform >Layout Designer and click the Initialization link near the Title field at the top of the page.

Image: Layout Initialization section on the Initialization page

This example illustrates the fields and controls in the Layout Initialization section on the Initialization page. You can find definitions for the fields and controls later on this page.

Package	Displays the package to which the PeopleSoft document linked to the layout belongs.
Document	Displays the name of the PeopleSoft document that is linked to the layout.
Version	Displays the version of the PeopleSoft document that is linked to the layout.
URI Document	Click the link to access the URI Document page.
Description	Enter a description for the initialization parameter.
Add a new row	Click the Add a new row button (+) to add a new parameter.
Delete row	Click the Delete row button (-) to delete a parameter.

Add Parameter	Click the link to add a row in the Parameter List section.
Delete Parameter	Click the link to delete a selected row in the Parameter List section.
Select Box	Select the box to flag a parameter value for deleting.
Name	Enter a name for the parameter value/argument.
Select Primitive	Click the link to access the Select a Primitive page to select and define a primitive as a parameter value.

Using the URI Document Page

Use the URI Document page (IB_DOCDISPLAY_SEC) to view the structure a URI for an initialization parameter. To access the page, click the URI Document link in the Layout Initialization section of the Initialization page.

Image: URI Document page

This example illustration the URI Document page.
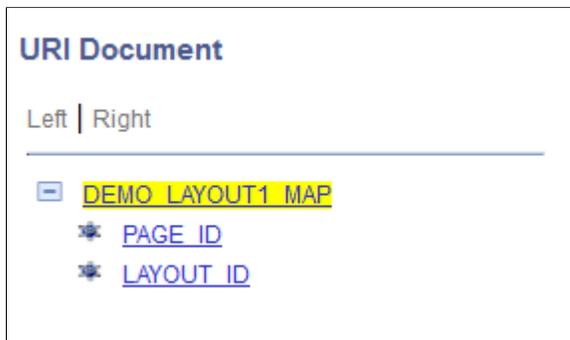
Click a primitive element in the document tree to view its data type. Click the Return button to return to the Initialization page.

Using the Select a Primitive Page

Use the Select a Primitive page (IB_DOCLOPRMSEL_SEC) to select a primitive as a parameter value when you are defining initialization parameters.

To access the page, click the Select Primitive link in the Layout Initialization section of the Initialization page.

Image: Select a Primitive page

This example illustrates the Select a Primitive page.

On the document tree, select a primitive to add an initialization parameter. The system adds the parameter and the initialization page appears.

Defining Layout Initialization Parameters and Values

This section discusses how to define layout initialization parameters and values.

Adding Initialization Parameters

You must add a row to begin defining initialization parameters. After you define the parameter, add arguments by clicking the Add Parameter link.

Image: Layout Initialization section of the Initialization page

The following example illustrates the Layout Initialization section of the Initialization page after a new row is added and the Add Parameter link has been clicked

In the Description field add information that describes the parameters to pass and the data returned as a result. This is important because the information will assist other developers determine whether to call the layout from their layouts.

Adding Initialization Parameter Values

The parameter values/arguments you specify are primitive elements defined in the PeopleSoft document linked to the layout, or constants that you define.

Important! The order in which you add parameter values is important, as that is the order in which the system constructs the URL.

If the parameter to pass is a primitive click the Select Primitive link to access the Select a Primitive page and click the appropriate element.

Image: Select a Primitive Page

Select a primitive in the document tree to pass as a parameter.
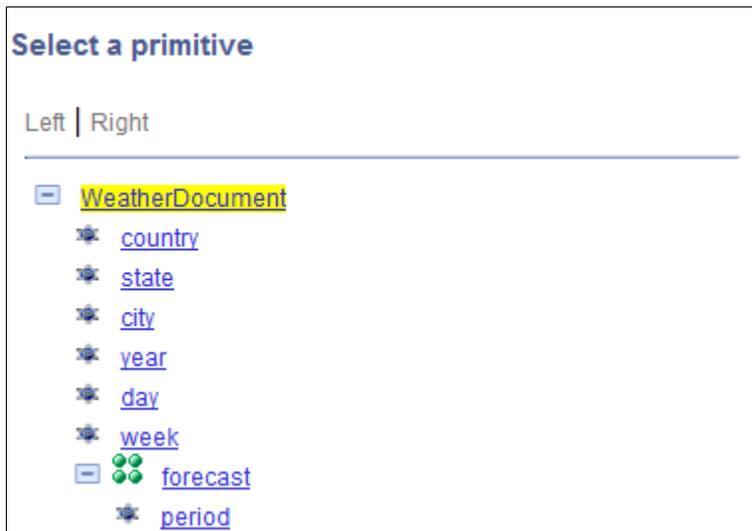
After you select a primitive from the document tree, the Layout Initialization page appears as shown in the following example:

Image: Layout Initialization section of the Initialization page

The primitive selected in the Select a Primitive page appears in the Parameter List section.

You'll note that the system assigns an index of 2 to the parameter and the generated URL for this initialization parameter and value is:

```
http://<machinename>:<port>//PSIGW/RESTListeningConnector/<NODE-NAME>/
DEMO_LAYOUT1_MAP.v1/INIT/-2
```

To add additional parameter values click the Add Parameter link. Another row appears in the Parameter List section.

After you've completed defining all values for the parameter, click the OK button. The Layout Designer – Layout page appears. Then, click the Save button to save the changes to the layout.

Using Constants for Initialization Parameter Values

This section discusses using constants for initialization parameter values.

Understanding Using Constants for Initialization Parameter Values

For the Mobile Application Platform framework to determine which URI is called, all URIs must be unique.

The following example shows two URIs that are not unique:

```
/<PAGE_ID>/<LAYOUT_ID/<primitive state>
/<PAGE_ID>/<LAYOUT_ID/<primitive city>
```

If the following values were passed in, the framework would be unable to determine to which URI these values belong:

```
/2/12/Kansas City
```

A simple solution for this problem is to use a constant for one of the URIs as shown below:

```
/<PAGE_ID/<LAYOUT_ID/State/<primitive city>
```

When the URI *2/12/State/Kansas City* is passed in, the framework will be able to determine the proper URI and properly populate the URI document within the MAP object. You can now properly get this initialization/URI index from the MAP Object.

Defining Constants for Initialization Parameter Values

If a parameter value is a constant, enter the constant name in the Name field. This example shows adding a list of parameters, including a constant:

Image: Layout Initialization section of the Initialization page

This example shows defining a constant as a initialization parameter value. In the example the value, *state,* is a constant. The value, *city,* is a primitive.
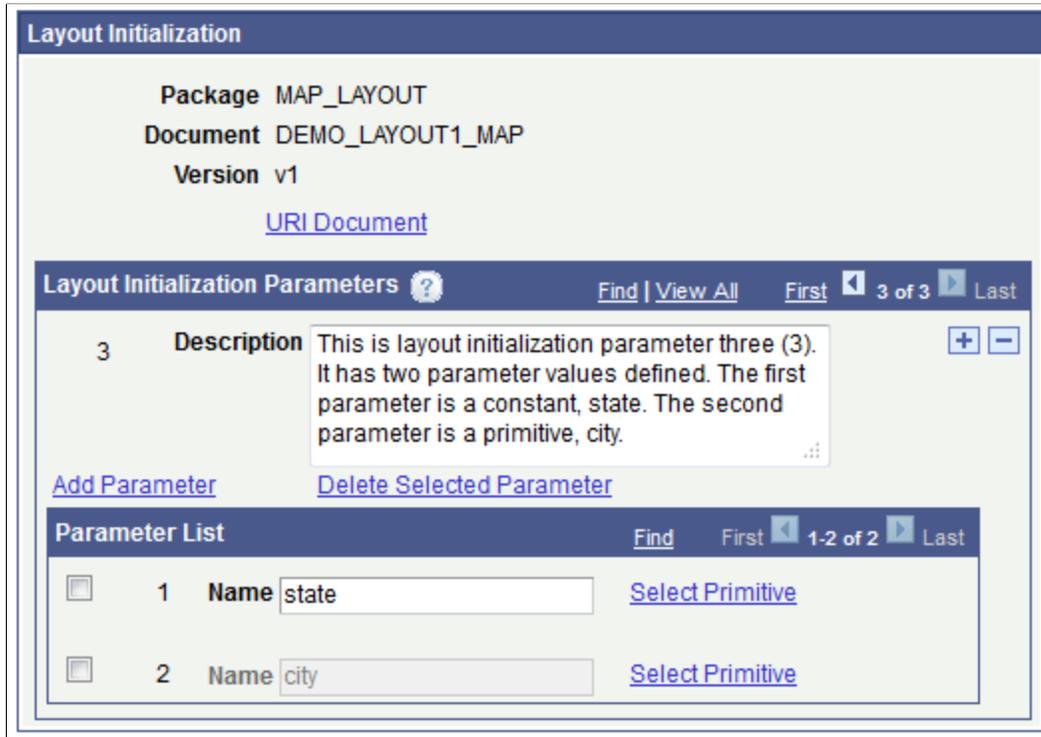
In this example, the first parameter value, *state,* is a constant and the second value, *city,* is a primitive selected from the Select a Primitive page. The URL the system generates from the parameter list values is:

```
http://<machinename>:<port>//PSIGW/RESTListeningConnector/<NODE-NAME>/
DEMO_LAYOUT1_MAP.v1/INIT/-3
```

Defining Initialization Parameters and Values for Layouts

To add a layout initialization parameter:

1. Access the Layout Initialization section of the Initialization page. (Select PeopleTools >Mobile Application Platform >Layout Designerand click the Initialization link.)

2. In the Layout Initialization Parameters section, click the Add a Row button (+).

3. In the Description field enter a description of the parameter.

4. Click the Add Parameter link.

5. In the Name field enter a parameter value.

   • To define a constant, enter it in the field.

- To define a primitive, click the Select Primitive field to access the Select a Primitive page and select the primitive to add.

6. To add additional parameter values, repeat Steps 4 -5.

7. Click the OK button. The Layout Designer – Layout page appears.

8. Click the Save button.

Example: Defining Initialization Parameters and Values

To better illustrate defining initialization parameters, consider an example where the following PeopleSoft document is defined for a layout:

Image: PeopleSoft Document

This example illustrates a sample PeopleSoft document defined for a layout.
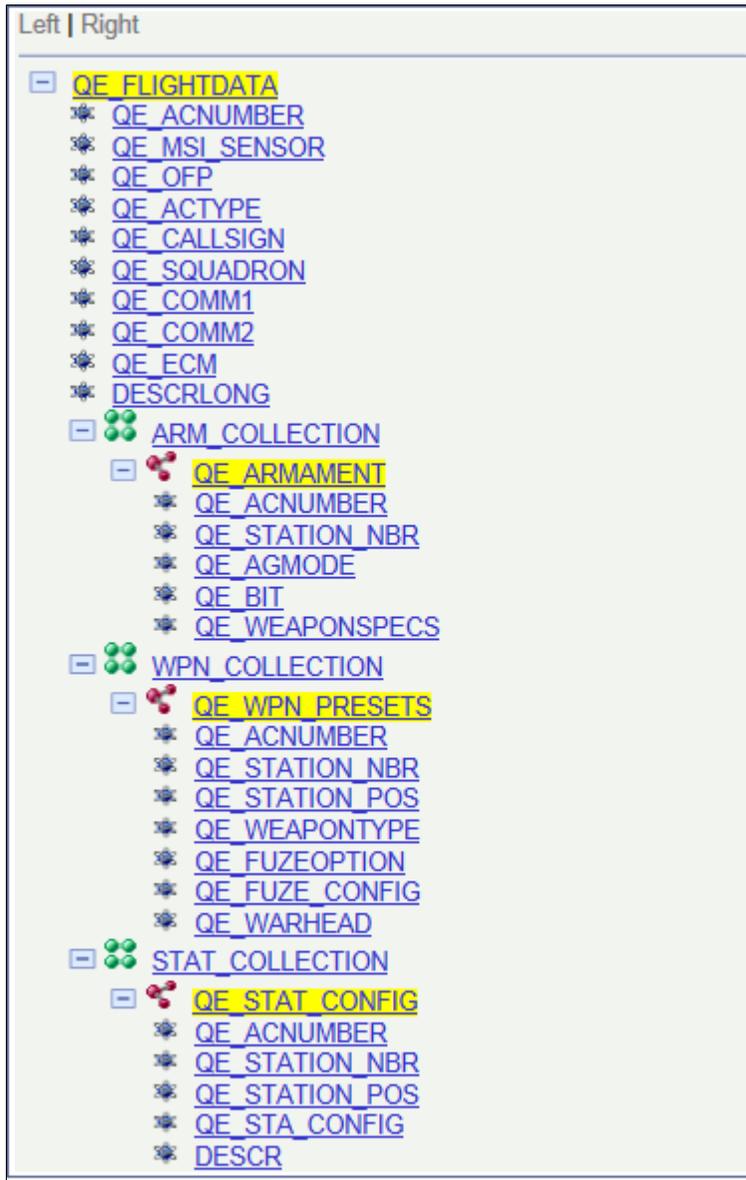
The following parameters are defined on the Layout Initialization Parameters page:

Image: Layout Initialization section of the Initialization page

This example illustrates layout initialization parameters and values defined.

In this case, four URI templates are created for the layout to use when mapping the record data on a list view or table.

Image: URI Document page

The following example illustrates the URI document created based on these definitions and that you access using the URI Document link at the top of the Layout Initialization section.
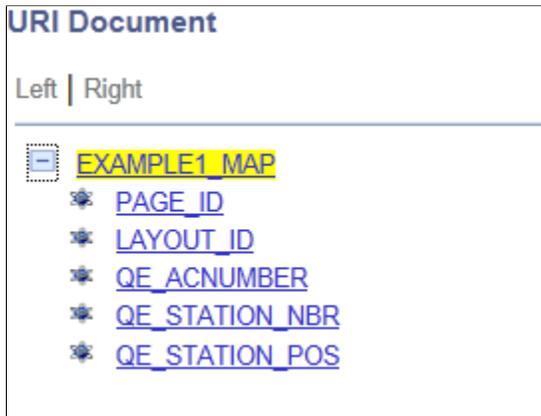
To see rows of data in a list view of *QE_ARMAMENT* record data, select the compound element *QE_ARMAMENT* to map to the list view and the URL selected becomes an external page where the URI template selected is *Layout Initialization 2*. The key data define in the primitive element *QE_ACNUMBER* is passed as part of the URL. The resulting external page has a list view defined with the compound *QE_WPN_PRESETS* to map to the list view, and the URL selected is an external page where the URI Template selected is *Layout Initialization 3*. The key data defined in primitive *QE_ACNUMBER* and *QE_STATION_NBR* are passed as part of the URL. This sort of pattern can be used to show the data for compound *QE_STAT_CONFIG*.

Deleting Layout Initialization Parameter Values

To delete a parameter list value:

1. In the parameter list of the Layout Initialization section select the value to delete.

2. Click the Delete Selected Parameter link.

3. Click the OK button. The Layout Designer – Layout page appears.

4. Click the Save button.

Deleting Layout Initialization Parameters

To delete an initialization parameter:

1. In the Layout Initialization Parameters grid of the Layout Initialization section of the Initialization page, click the Delete Row button (-).

2. Click the OK button at the message prompt.

3. Click the OK button at the button of the Initialization page.

    The Layout Designer – Layout page appears.

4. Click the Save button.

Chapter 19

Applying and Synchronizing Layout Templates

Understanding Applying Layout Templates

You can apply a layout template to one or more pages in a layout. If a template was applied to a layout when the layout was added to the system, any template you add to an individual page takes precedence.

Understanding Synchronizing Layout Templates

When you add a new template to a layout page or after you modify an existing template that's already assigned to layout page, you must synchronize the template with the layout so that the template gets applied to the layout page.

If you've manually defined styles in the Layout grid for an element and do not want the template to override them, check the No Styles Override box. When you check this box the synchronization process will not override any of the manually-defined display options during the synchronization process.

Applying and Synchronizing Templates in Layouts

This topic describes how to apply and synchronize templates to layouts and pages.

Applying Templates to Layouts and Pages

To apply a layout template to a layout or page:

1. Access the Layout Designer – Layout page. (Select PeopleTools >Mobile Application Platform >Layout Designer.

2. Navigate to the page in the layout where you want to apply a template.

3. From the Template Name drop-down list, select a template.

4. (Optional.) In the Layout Grid, select the No Styles Override box for any styles defined that you do not want overridden during the synchronization process.

5. Click the Sync Element Types button.

6. Click the Save button.

7. Exit the layout definition and access it again.

You must exit the layout definition and access it again for any changes and additions to CSS and JavaScript to appear on the Layout Designer – CSS page and Layout Designer – JavaScript page, respectively.

Synchronizing Templates with Layouts and Pages

If a template is applied to a layout and you make changes to the template, you must synchronize the template changes with any layouts or individual pages to which the template is applied.

To synchronize a layout template with a layout or page:

1. Access the Layout Designer – Layout page. (PeopleTools >Mobile Application Platform >Layout Designer.

2. Navigate to the page that contains the template to synchronize.

3. (Optional.) In the Layout Grid, select the No Styles Override box for any styles defined that you do not want overridden during the synchronization process.

4. Click the Sync Element Types button.

5. Click the Save button.

6. Exit the layout definition and access it again.

You must exit the layout definition and access it again for any changes and additions to CSS and JavaScript to appear on the Layout Designer – CSS page and Layout Designer – JavaScript page, respectively.

Chapter 20

Adding and Deleting Layout Pages

Adding Pages to Layouts

When you add a page to a layout, the system adds it as the last page in the Layout Designer.

To add a page to a layout:

1. Access the Layout Designer – Layout page. (Select PeopleTools >Mobile Application Platform >Layout Designer.

2. Under the Layout Details section, click the Add Page button.

3. Click the Save button.

Deleting Pages from Layouts

All layouts must contain at least one page, therefore you cannot delete a page if it is the only page in a layout.

After you delete one or more pages from a layout, you must save the changes, exit the Layout Designer, and access the component again, so the system can renumber the remaining pages in the layout.

To delete a page from a layout:

1. Access the Layout Designer – Layout page. (PeopleTools >Mobile Application Platform >Layout Designer.

2. Navigate to the page to delete.

3. Under the Layout Details section, click the Delete Page button.

    The system displays a message to confirm the delete.

4. Click the OK button in the message box.

5. Click the Save button.

6. Exit the Layout Designer.

7. Access the Layout Designer to continue working on the layout.

Chapter 21

Setting Layout Page Properties

Understanding Page ID, Page Loading, and Page Caching Properties

This topic describes the following layout page properties:

- Page ID.

- Page Loading.

- Page Caching.

Page ID

A page ID uniquely defines a page in a layout and is included in the URL for the page.

In addition, depending on your requirements you may need to reference the page ID as you develop PeopleCode, JavaScript, jQuery events and Ajax events for your application pages.

The system sequentially assigns a default page ID to all pages, assigning the default page ID value of the first page in a layout *page1,* the second page in the layout being *page2,* and so on.

You can change the default page ID value, however you are responsible for ensuring that page IDs remain unique throughout a layout, as the system does not perform page ID validation.

If you change the default page ID when a page is defined as an internal page, a warning message appears to alert you that changing the ID may invalidate URLs/links to the page from other pages in the layout. If you have created links to a page and later change the page ID, navigate to the page or pages that have button elements that call the URL, access the button properties and rebuild the links using the new page ID.

Page Loading and Caching

The Mobile Application Platform framework provides several page properties to assist with memory management.

Page Pre-Load   The framework provides a cache pre-load option that allows you to specify that any external pages referenced on a page to be cached even before they a users links to the pages and before they are displayed in the browser. Each pre-loaded page creates additional HTTP requests, and there for uses more bandwidth. Therefore, it is important to use the pre-load option only when you think visitors are likely to view the page.

Page Caching	The framework provides an option to cache pages in the Document Object Model (DOM) using the jQuery framework, instead of browser caching.

Using the Page Details Page

Use the Page Details page (IB_DOCPAGEDTLS_SEC) to set properties for a layout page.

To access the select PeopleTools >Mobile Application Platform >Layout Designer. Navigate to the page for which to set properties and click the page number above the document tree.

Image: Page Details page

This example illustrates the fields and controls on the Page Details page. You can find definitions for the fields and controls later on this page.

Page Index	Displays the page number.
Template Name	If applied, displays the name of the template applied to the page.
Page Name	If using tabbed pages, enter the name to display on a tabbed page.
Page ID	Enter the ID to assign to the page. The default value is the page number in the format *page1*.
Main Class	(Optional.) Select a main class to apply to the page.
	If a layout template is applied to the page, this field displays the main class defined on the Main container in the template if one is defined. To override the value, click the Lookup button and select another Main class from any of the other container classes defined in the template.
Main Style	(Optional.) Select a main style to apply to the page.
	If a layout template is applied to the page, this field displays the main style defined on the Main container in the template if one is defined. To override the value, delete the contents of the field and enter styles in CSS3 format.
Main Theme	(Optional.) Select a theme to apply to the page.
	If a layout template is applied to the page, this field displays the main theme defined on the Main container in the template if one is defined. To override the value, select another theme from the drop-down list.
	See the "Themes" section in the <u>Understanding Layout Template Container Terms and Concepts</u> topic for more information about the values that display in the drop-down list.
Content Class	(Optional.) Enter a class name to apply to the content of the page.
	If a layout template is applied to the page, this field displays the class defined for the content container if one is defined. To override the value, click the Lookup button and select another class from any of the other container classes defined in the template.
Content Theme	(Optional.) Select a theme to apply to the content on the page.
	If a layout template is applied to the page, this field displays the theme defined on the content container in the template if one is defined. To override the value, select another theme from the drop-down list.

See the "Themes" section in the <u>Understanding Layout Template Container Terms and Concepts</u> topic for more information about the values that display in the drop-down list.

(Page) **Transition**

(Optional.) Define the transition from this page to other pages in the layout. The options are:

• *Fade.* Fade the page or dialog in over the previous content.

• *Flip.* An animated page flip, rotating the current view out with the other view on the reverse side.

• *Flow.* Throws the current page away and comes in with the next page.

• *Pop.* The page springs into view from the center of the screen.

• *Slide.* Slide in from the left or right, pushing the previous content out of the way.

• *Slidedown.* Slide down from the top, over the top of the current container.

• *Slidefade.* Slide in from the left or right then fade the page.

• *Slideup.* Slide up to the top, revealing the next content below.

• *Turn.* The page turns as in turning a page in a book.

Form Transition

(Optional.) If the page has a Form control as part of its page (that is, a Save button is defined) the default transition can be set.

Select the transition from the drop-down list.

The same values for page transitions described in the previous row are available for you to choose for form transitions.

Client Preload/ Cache

(Optional.) Use this section of the page to set preload and cache settings. The valid values are.

• *Preload.* Select the box to cache external page(s) referenced on the page, even before they are linked to or displayed on the browser.

• *Cache Page.* Select the box to cache the page(s) in the Document Object Model (DOM) via the jQuery framework versus the browser.

Sidebar Responsive

(Optional.) If a sidebar is part of the page, select the Sidebar Responsive box to set a breakpoint in the widget at 55em (880px).

This breakpoint is not applied by default to make it easier for you to write custom breakpoints that work best for your content and design.

Page Type   Select the page type.(Optional.) Select the box to define the page as an external page. You can define any page in a layout an external page, other than the first page of a layout. This option is not enabled on the first page of a layout.

- Internal Page

  The first page in a layout is always internal.

- External page

  You can define any page in a layout an external page, other than the first page of layout. This option is not enabled on the first page of a layout.

- Sub Page

  You can define any page in a layout an sub page, other than the first page of a layout. This option is not enabled on the first page of a layout.

This setting impacts page and layout navigation. See <u>Defining Page and Layout Navigation</u> for more information about the impact of this setting.

External Page as a Sub Page

You can use a Sub Page as a re-direction when it comes to an External Page. For example say Page 3 is defined as an External Page. When the External Page is invoked, if the External Page is defined with an Optional Sub Page Number then that page will be rendered instead of the External Page

Image: External Page Type Options

This example illustrates External Page Type selected.
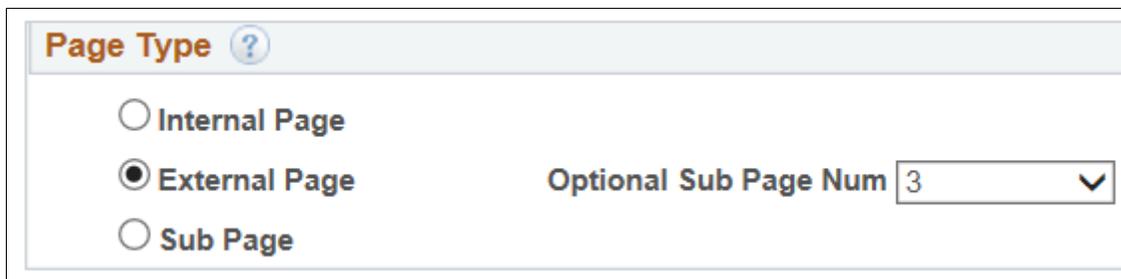
Chapter 22

Inserting Page Elements into Pages

Using the Layout Toolbar and Layout Grid

The Layout Designer – Layout page features a toolbar and grid that you use to:

- Insert page elements into a layout page.

- Access page element properties.

- View summary details about the page elements inserted in a page.

To access the Layout grid and toolbar, select PeopleTools >Mobile Application Platform >Layout Designer.

Image: Layout toolbar and Layout grid

This example illustrates the fields and controls on the Layout toolbar and Layout grid. You can find definitions for the fields and controls later on this page.

Using the Layout Toolbar

Use the layout toolbar to insert, move and delete page elements in the layout grid.

Image: Layout toolbar

This example illustrates the fields and controls on the Layout toolbar. You can find definitions for the fields and controls later on this page.

The toolbar is divided into three sections. Use the first three buttons to move and delete page elements in the Layout grid. Use the middle buttons to add page elements to the layout grid. And use the last two

buttons to copy pages and containers into the layout grid, and to add references to other layouts in the grid.

Use the following buttons to move and delete page elements in the Layout grid:

Button	Label	Description
	Move Item Up	Click a row in the Layout grid and click the button to move the selected page element up in the grid.
	Move Item Down	Click a row in the Layout grid and click the button to move the selected page element down in the grid.
	Delete Current Row	Click a row in the grid and click the button to delete the selected page element from the page.

Use the following toolbar buttons to add elements to the Layout grid:

Button	Label
	Add Container
	Add Mobile Grid
	Add Static Text
	Add Image/Static Text
	Add Listview
	Add Select Menu
	Add URL
	Add HTML Area
	Add Action Sheet
	Add Flip Toggle
	Add Pushbutton
	Add Input
	Add Select Box

Button	Label
⊙	Add Radio Button
☐	Add Checkbox
⫼	Add Slider
📈	Add Table
▤	Add Sidebar

Use the following toolbar buttons to copy and reference containers in the Layout grid:

Button	Label
▥	Copy page/container
▤	Reference Layout Container
▦	Sub Container

Using the Layout Grid

Use the Layout grid on the Template Designer – Layout page to add elements to a mobile application page and define the properties for the elements.

Image: Layout grid

This example illustrates the fields and controls on the Layout grid. You can find definitions for the fields and controls later on this page.

		ID	Label Text	Group		Class Name	Style	Theme	No Styles Override
➡	▲	mapheader_1		Properties				▼	☐
—	▲	mapheader_end		Properties				▼	☐
—	▤	mapfooter_1		Properties				▼	☐
—	▤	mapfooter_end		Properties				▼	☐

View Layout Grid Details
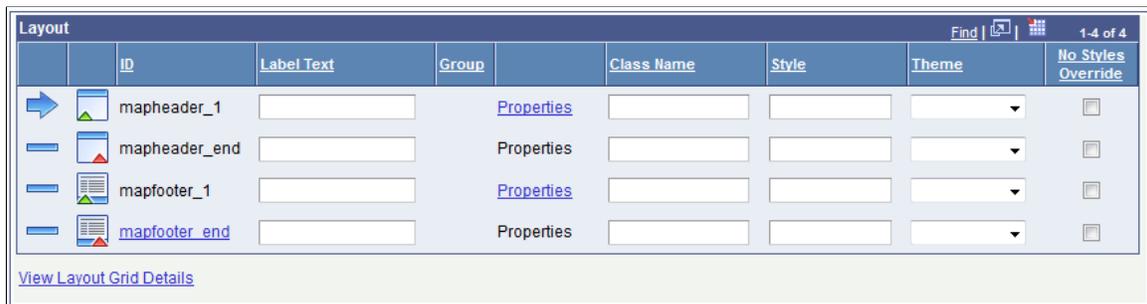
By default, header and footer elements are added to every page in a layout. These elements appear in the Layout grid by default with the default element IDs *mayheader_1/mapheader_end* and *mapfooter_1/ mapfooter_end,* respectively.

The following table describes the buttons and icons that appear in the Layout grid:

Button/Icon	Label	Description
	Current Row	Indicates the current row in the grid.
	Non-current Row	Indicates a non-current row.
	NA	Several page elements allow you to nest other page elements within them, including these element: • Header. • Container. • Mobile Grid. • Action Sheet. • Footer. A green triangle at the bottom right-corner of a page element icon indicates where you can begin to nest other elements, and functions similar to a visual start tag. To nest an element, click the icon in the grid under which to nest a page element, and then click a page element in the Layout toolbar to insert the new page element within the top element.
	NA	A red triangle at the bottom right-corner of a page element icon indicates the end of a top element for nesting purposes.

This table describes the fields and controls on the Layout grid:

ID   Indicates the unique system-generated ID for an element.

This ID can be used in JavaScript.

If an element ID appears in the grid as a link, you can assign a custom ID by clicking the link.

If you choose to customize element IDs, you are responsible for ensuring that you use unique IDs, as the system does not scan for duplicate IDs.

Label Text   Enter a label for the element as it is to appear on the user interface.

Group   You can group together certain element types to achieve specific behaviors.

Grouped elements are denoted by a system-generated group number and the group numbers appear in this field.

Properties	Click the link to define properties for the element.
Class Name	Enter a class name to assign to the element.
Style	Enter styles for the element.
Themes	Choose a jQuery theme for the element.
No Styles Override	Select the box so that styles you define for an element are not overridden by the application of a template, or when you use the Sync Element Types button to synchronize a layout with a template.
View Layout Grid Details	Click the link to view a list of elements and element properties defined in the Layout grid.

Inserting Page Elements into the Layout Grid

To insert a page element into the layout grid, select the row/icon directly above where you want to insert the page element. The Current Row button (the large blue arrow) in the first column of the grid indicates the current row.

Image: Layout grid

By default the first row in the grid is the current row, as indicated Current Row icon in the first column of the grid, positioned in front of the element with the ID *mapheader_1* shown in this example.

If you click an element on the toolbar, say for example the Add Pushbutton button, the system adds a button element to the grid directly under the current row.

In this example, the *mapheader_1* element, as shown in the following example:

Image: Layout grid

This example shows inserting a push button element into the Layout grid. After you insert an element into the grid, the row that contains the element you added becomes the current row.

In this case, the inserted button appears in the header of the page.

Except for headers and footers, you can insert more than one of any element type into the grid. To help identify multiple instances of element types, the system adds a sequential number every time you add one.

So for example, if you were to add another button to the grid in the previous example, it would appear as *mapbutton_2* in the grid.

Note for the majority of page elements, including the button element, after you insert an element into the Layout grid, the a definition page for the element appears and you must define the element. The exceptions are the page elements which allow nested elements, that is, elements where you can insert other elements, including the Container, Mobile Grid, and Action Sheet elements.

After you insert and define an element, you can click the Properties link in the Layout grid to define additional parameters for the element.

To insert a page element into the layout grid:

1. Access the Layout Designer – Layout page. (Select PeopleTools >Mobile Application Platform >Layout Designer.)

2. Navigate to the page where you want to insert a page element.

3. Click the row directly above the position where you want to insert the element. The Current Row icon appears in front of the element.

4. On the Layout grid toolbar, click the page element to insert.

5. Depending on the element you select, the properties page for the element may appear. If so, define the page properties and click the OK button.

6. The Layout Designer – Layout page appears and the page element appears in the Layout grid.

7. Click the Save button.

Setting Alternate Page Element IDs

When you add a page element to the Layout grid, the system assigns the element a unique element ID. This value appears in the ID column in the Layout grid.

As an example, the system assigns the element ID *mapbutton_1* to the first push button element you add to a layout page.

Depending on your business and development requirements, you may need to use the page element ID if you develop any JavaScript for your application.

You can override the system-generated page element ID for any element ID that appears as a hyperlink in the layout grid and set an alternate value.

Click the link to access the Add Alternate Label page (IB_DOCLOGUILBL_SEC) to define a unique value for the page element ID.

Image: Add Alternate Label page

This example illustrates the fields and controls on the Add Alternate Label page. You can find definitions for the fields and controls later on this page.
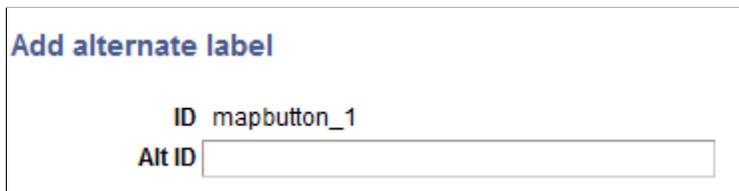
| **ID** | Displays the system-generated element ID. |
| **Alt ID** | Enter an alternate value for the system-generated element ID. |

Note: The system does not perform any validation on user-defined page element IDs. You are responsible for ensuring that any alternate page element IDs that you define are unique.

To set an alternate page element ID:

1. Access the Layout Designer – Layout page. (PeopleTools >Mobile Application Platform >Layout Designer.)

2. In the ID column in the Layout grid, click the hyperlinked element ID.

    The Add Alternate Label page appears.

3. In the Alt ID field, enter a unique ID for the page element.

4. Click the Save button.

    The Layout Designer – Layout page appears.

Defining Display Labels for Page Elements

The Layout grid on the Layout Designer – Layout page (PeopleTools >Mobile Application Platform >Layout Designer) features a Label Text field. Use the field to define the label that appears on a page element in the generated HTML.

For example, for a push button element you could enter a value of *Save* in the Label Text field, and in the rendered HTML the button would appear labeled *Save*.

Defining Styles and Themes for Page Element Labels

You can define display styles and themes for any page element you add to the Layout grid.

The Layout grid features the following fields for defining styling characteristics for page elements. In most cases the styling applies to the element label as it appears in the application at runtime. One exception is the HTML Area element. Styling you define in these fields for an HTML element is applied to the HTML area.

Note that the location in the HTML where an element is embedded with respect to other elements can impact if styles defined in these fields is applied at runtime.

Class Name	If you want to apply a class to the element label, enter the class name in the field.
Style	If you want to add additional styles to a page element label or override styles defined in a template, enter them in CSS3 format in the Style field. If you have not applied a layout template and simply want to apply styles to the element label, enter them in CSS3 format in the *Style* field.
Theme	If you want to override a theme defined for an element label, select a theme from the drop-down list. If you have not applied a layout template and simply want to apply a theme to the element label, select a theme from the drop-down list.

Other styling properties can be defined for an element using the properties page for the element.

To define and manage display styles for a page, use the Page Details page. See Using the Page Details Page for more information.

Preventing Display Style Overrides for Page Elements

You can prevent the system from overriding display styles you set for a page element in the Class Name, Styles and Theme fields in the Layout grid. Each row in the Layout grid features a No Styles Override box. Select the box to prevent display style overrides for an element anytime template synchronization occurs using the Synch Element Types button, such as when a new template is applied to a layout page or when a template has been modified and the changes are synchronized with a layout page.

In these cases, the displays styles you have defined for a page element remain in effect when the No Styles Override box is selected for a page element.

Moving Page Elements in the Layout Grid

You can move or reposition the order in which page elements are placed in the layout grid.

To move an element in the layout grid:

1. Select the element in the grid that you want to move and verify that the blue arrow icon in the first column of the grid appears in front of the element to move.

2. On the Layout grid toolbar use the following buttons:

 Click the button to move the selected page element up in the grid.

 Click the button to move the selected page element down in the grid.

3. Click the Save button.

Deleting Page Elements from the Layout Grid

Use the Delete button on the Layout grid toolbar to delete elements from the Layout grid.

If you delete Container, Mobile Grid or Action Sheet elements, note that the system deletes any elements that are nested within these.

To delete a page element from the Layout grid:

1. Select the element in the grid that you want to delete and verify that the Current Row icon in the first column of the grid appears in front of the element that you want to delete.

2. On the Layout grid toolbar click the Delete button.

3. Click the Save button.

Page Element Reference

The following table lists elements that you can nest or insert into other elements:

Button	This element . . .	. . . can be inserted into this/these element(s).	Comments
	Header	NA	NA

Button	This element . . .	. . . can be inserted into this/these element(s).	Comments
	Footer	NA	NA
	Container	• Header • Footer • Container • Mobile Grid	If you delete a Container element from a Header or Footer element, all elements defined within the Container element are also deleted. You cannot nest multiple Container elements into Header, Footer, or Mobile Grid elements.
	Mobile Grid	• Header • Footer • Container	You cannot insert a Mobile Grid element into a Container element located in a Header or Footer element.
	Static Text	• Header • Footer • Container • Mobile Grid	NA
	Image/Static Text	• Header • Footer • Container • Mobile Grid	NA
	Listview	• Header • Footer • Container • Action Sheet	NA
	Select Menu	• Container • Mobile Grid • Action Sheet	NA
	URL	• Header • Footer • Container • Action Sheet	NA

Button	This element . . .	. . . can be inserted into this/these element(s).	Comments
	HTML Area	• Header • Footer • Container • Mobile Grid	NA
	Action Sheet	• Header • Footer • Container • Mobile Grid	You cannot insert an Action Sheet element into a: • Container element that's nested in a Header or Footer element. • Mobile Grid element that's nested into a Header element or a Footer element. If you delete an Action Sheet element from a Header or Footer element, all elements defined within the Action Sheet element are also deleted.
	Flip Toggle	• Container • Mobile Grid • Action Sheet	NA
	Push Button	• Header • Footer • Container • Action Sheet	NA
	Input	• Container • Mobile Grid • Action Sheet	NA
	Select Box	• Container • Mobile Grid • Action Sheet	NA
	Radio Button	• Container • Mobile Grid	NA

Button	This element . . .	. . . can be inserted into this/these element(s).	Comments
	Check Box	• Container • Mobile Grid • Action Sheet	NA
	Slider	• Container • Mobile Grid	NA
	Table	Container	NA
	Sidebar	Footer	Footer End element

Chapter 23

Defining Layout Page Elements and Properties

Defining Layout Page Elements and Properties

The following table features links to topics that describe how to define page elements in the Mobile Application Platform.

Layout Page Element	*Defining the Page Element*	*Defining Page Element Properties*
Action Sheet	See Defining Action Sheet Page Elements	See Defining Action Sheet Page Element Properties
Input	See Defining Input Page Elements	See Defining Input Page Element Properties
Check Box	See Defining Check Box Page Elements	See Defining Check Box Page Element Properties
Container	See Defining Container Page Elements	See Defining Container Page Element Properties
Flip Toggle	See Defining Flip Toggle Page Elements	See Defining Flip Toggle Page Element Properties
Footer	See Understanding Header and Footer Page Elements	See Defining Header and Footer Page Element Properties
HTML	See Defining HTML Page Elements	See Defining HTML Page Element Properties
Header	See Understanding Header and Footer Page Elements	See Defining Header and Footer Page Element Properties
Image/Static Text	See Defining Static Text and Image Page Elements	See Defining Image Properties
List View	See Defining List View Page Elements	See Defining List View Page Element Properties
Mobile Grid	See Defining Mobile Grids Page Elements	See Defining Mobile Grid Page Element Properties
Push Button	See Defining Push Button Page Elements	See Defining Push Button Page Element Properties
Radio Button	See Defining Radio Button Page Elements	See Defining Radio Button Page Element Properties

Layout Page Element	Defining the Page Element	Defining Page Element Properties
Select Box	See Defining Select Box Page Elements	See Defining Select Box Page Element Properties
Select Menu	See Defining Select Menu Page Elements	See Defining Select Menu Page Element Properties
Sidebar	See Defining Sidebar Page Elements	See Defining Sidebar Page Element Properties
Slider	See Defining Slider Page Elements	See Defining Slider Page Element Properties
Table	See Defining Table Page Elements	See Defining Table Page Element Properties
Text	See Defining Static Text and Image Page Elements	See Defining Static Text Properties
URL	See Defining URL Page Elements	See Defining URL Properties

Chapter 24

Common Pages Used in Defining Layout Page Elements

Managing Element Groups

This topic discusses how to manage element groups in the mobile application platform.

Creating Element Groups

Use the "Do you want to create a group?" page (IB_DOCLOGRP_SEC) to create a group of two or more of the same page element.

This page appears when you define the following page elements:

- List view.

- Select menu.

- Push button.

- Radio button.

- Check box.

Image: "Do you want to create a group?" page

This example illustrates the fields and controls on the "Do you want to create a group?" page. You can find definitions for the fields and controls later on this page.

> **Do you want to create a group?**
>
> Number of Group Items [ ]

If you do not want to create a group of elements, click the No button. The definition page for the element appears and you can continue to define the page element.

If you want to create a group of elements, in the Number of Group Items field enter the number of elements to include in the group and click the Yes button. The definition page for the first element in the group appears. In addition, a Group field appears at the top of the definition page, indicating the number of elements in the group and the instance of the element that is currently displaying.

The following example shows the first definition page when you are defining a push button, the Select Button Type page. The Group field above the Button Type box indicates that a group was defined for the element. The Group field value is *Item 1 of 3,* indicating that there is a total of three buttons in the group and you are viewing the first button definition in the group.

Image: Select Button Type page

This example illustrates the Select Button Type page when a group has been defined. In this example, the Group field indicates that you are viewing the first of three buttons in the group.

After you define the first item in the group, the system automatically displays the second item in the group for you to define, and repeats the process until you have defined all items in the group.

Viewing Element Groups

The system assigns a sequential number to each set of grouped elements that you create and the group number appears in brackets ([ ]) in the Group column next to each element in the group.

Image: Layout grid

This example illustrates the Layout grid on the Layout Designer – Layouts page with two groups of button elements defined in the layout.

The previous example shows that there is a group of button elements defined in the header (*mapbutton_1, mapbutton_2, mapbutton_3*), as denoted by *[1]* in the Group column for each element. There is a second group of buttons defined in the footer(*mapbutton_4, mapbutton_5, mapbutton_6*) , as denoted by the *[2]* in the Group column for each element.

Deleting Grouped Elements

If you delete one element in the group, the system deletes all elements in the group. You cannot delete individual elements in a group.

Defining Static Text and Image Page Elements

This topic describes how to define text page elements and image page elements.

Understanding Defining Text Page Element and Image Page Elements

Static Text Page Elements

When you define a static text page element you define the source of the text for the element. After you define the element you can set properties for the element using the Static Text Properties page.

Image Page Elements

When you define an image page element you have the option to define a text label for the element, which like the static text page element requires that you select the text source. In addition, you define the image for the element. You can select a static image from the database or define a dynamic image by selecting a primitive element.

Text Sources

The available sources of text when you define a static text page element or you define a text label for an image element are:

- *Use Label Text.* Label text from the Label Text field in the Layout grid on the Layout Designer – Layout page.

- *Message Catalog.* Text from a message catalog entry.

- *Primitive.* Text from a label of a primitive element defined in the PeopleSoft document defined for the layout.

Image Sources

The sources of images for image page elements are the application database or a primitive element defined in the PeopleSoft document defined for the layout.

The Mobile Application Platform provides a Layout Image Upload page that you can use to add and replace images for use in your mobile application. You can also use PeopleSoft Application Designer to add images to the database.

Defining Text Page Elements

This topic describes how to:

- Access the Select Static Text Type Page.

- Define label text as the text type.

- Define message catalog text as the text type.

- Define a primitive label as the text type.

- Modify text types.

Understanding Defining Text Page Elements

After you define a static text page element, use the Static Text Properties page to define properties for the element. See Accessing Page Element Properties for information about how to access the properties page.

All of the properties that appear on the Static Text Properties page also appear on the properties pages for other page elements. These common properties are described in the topic Common Layout Page Element Properties.

Accessing the Select Static Text Type Page

To access the Select Static Text Type page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid toolbar click the Add Static Text button.

Image: Select Static Text Type page

This example illustrates the fields and controls on the default view of the Select Static Text Type page. You can find definitions for the fields and controls later on this page.

Defining Label Text as the Text Type

To define label text as the static text type, access the Select Static Text Type page and select the Use Label Text box.

Image: Select Static Text Type page

This example illustrates the fields and controls on the Select Static Text Type page when you select Use Label Text as the source. You can find definitions for the fields and controls later on this page.

Select Static Text Type

☑ Use Label Text
◉ Message Catalog
○ Primitive

When you check the Use Label Text box, the system uses the text in the Label Text field in the Layout grid on the Layout Designer page for the static text.

Defining Message Catalog Text as the Text Type

To define Message Catalog text as the static text type, access the Select Static Text Type page and select the Message Catalog radio button.

Image: Select Static Text Type page

This example illustrates the fields and controls on the default view of the Select Static Text Type page. You can find definitions for the fields and controls later on this page.

Select Static Text Type

☐ Use Label Text
◉ Message Catalog
○ Primitive

Message Set Number [   ] 🔍   **Message Catalog Type**
                                   ◉ Use Message Text
Message Number [   ] 🔍   ○ Use Message Description

Message Text
Description [   ]

Use the fields and controls in the dialog box to select a message set/number. The message text and description appear at the bottom of the page. In the Message Catalog Type section, you then choose to use the message text or message description as the static text for the element.

Use these fields and controls to define a Message Catalog entry as the text type:

Message Catalog	Click the radio button to select Message Catalog for the static text type.
Message Set Number	Enter a message set number or click the Lookup button to search for one.
Message Number	Enter a message number or click the Lookup button to search for one.

Message Catalog Type   Choose the part of the message catalog enter to use:

- Use Message Text. Select the radio button to use the text that appears in the Message Text field.

- Use Message Description. Select the radio button to use the text that appears in the Description box.

Defining a Primitive Label as the Text Type

To define a primitive label as the static text type, the primitive must be defined in PeopleSoft document defined for the layout.

To define a primitive as the static text type, access the Select Static Text Type page and select the Primitive radio button.

Image: Select Static Text Type page

This example illustrates the fields and controls on the Select Static Text Type page when you select Primitive as the static text type. You can find definitions for the fields and controls later on this page.

Select	Name	Type
☐	country	String (30)
☐	state	String (30)
☐	city	String (30)
☐	year	String (30)
☐	day	String (30)
☐	week	String (30)
☐	period	String (30)

Select a primitive from the list.

After you select a primitive from the list, the Layout Designer – Layout page appears and the Label Text field in the Layout grid is populated with the text.

Modifying Text Types

After you initially define the text type for a static text page element, you cannot modify it.

Delete the existing page element and define a new one.

Defining Image Page Elements

This topic describes how to:

- Access the Select Image/Static Text Page.

- Define text labels for image elements.

- Omit text labels for image elements.

- Define images for image elements.

Understanding Defining Image Page Elements

After you define an image page element, use the Image Properties page to define properties for the element. See Accessing Page Element Properties for information about how to access the properties page.

All of the properties that appear on the Image Properties page also appear on the properties pages for other page elements. These common properties are described in the topic Common Layout Page Element Properties.

Accessing the Select Image/Static Text Page

To access the Select Image/Static Text page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid toolbar click the Add Image/Static Text button.

Image: Select Image/Static Text page

This example illustrates the fields and controls on the Select Image/Static Text page.

Defining Text Labels for Image Elements

As described previously, when you define an image page element, along with selecting an image for the element, you have the option to define a label for the image.

The label options for an image page element are the same as those for defining a text page element. Fields and controls for defining image label text are the same as those for defining text page elements using the Select Static Text Type page. See the Defining Text Elements section for defining label text.

See the Selecting Images for Image Elements later in this topic for information on how to define images for image elements.

Omitting Text Labels for Image Elements

To create an image element without including an image label, on the Select Image/Static Text page, select the Use Label Text option and in the Layout grid leave the Label Text field blank for the element.

Defining Images for Image Elements

The following example shows the bottom portion of the Select Image/Static Text page used for defining an image for an image element.

Image: Select Image/Static Text page (Image selection fields)

This example illustrates the fields and controls on the Select Image/Static Text page used for selecting an image for the page element. You can find definitions for the fields and controls later on this page.

Add Image	Select the control to define a static image for the element.
	When you select the control the Image Select link appears on the page.
Image Selector	This link appears after you select the Add Image box.
	Click the link to access the Image Selector page to select an image from the database.
Image Name	After you select an image from the Image Select page the image name appears in the field.
Select Primitive	Click the link to access the Select Primitive page to select a dynamic image for the page element.
Selected Element	After you select a primitive from the Select Primitive page the element name appears in the field.

Chapter 25

Accessing Page Element Properties

Accessing Page Element Properties

To access the properties page for a Mobile Application Platform page element:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

   The Layout Designer — Layout page appears.

2. In the Layout grid locate the page element.

3. Click the Properties link.

Chapter 26

Working with Common Page Element Properties

Common Layout Page Element Properties

Many layout page elements have common properties and use the same pages and page elements to define properties. This section lists the common page elements and provides links to the associated topics.

Note: Not all properties, pages, and page elements listed in this topic are present for all page properties.

When defining properties for an element, use this page as a reference to documentation for the specific properties that are available to define for the page element.

Common Properties Used to Define Element Properties

This table lists common groups of properties that appear in individual sections on many layout property pages. Links are provided to the associated topic that provide more detail on how to define the properties. The property (sections) are listed in alphabetical order.

Property Section	Description	Topic
Accessibility section	Define accessibility properties.	Defining Accessibility Properties
Ajax Events section	Define Ajax events.	Defining Ajax Event Page Element Properties
Change Type	Change the primitive type.	Changing Element Type
Class Selection section	Apply template containers and select style classes,	Applying Template Containers and Selector Style Classes
Image Properties section	Define image properties.	Defining Image Properties
JavaScript Events section	Define JavaScript events as element properties.	Defining JavaScript Event Page Element Properties
Optional Plug-In section	Define JQuery Mobile plug-ins.	Defining jQuery Mobile Plug-In Properties
Static Text Properties section	Define static text properties.	Defining Static Text Properties
URL Properties section	Define URL properties.	Defining URL Properties

Common Page Elements Used to Define Element Properties

This table lists a common page element used to define page element properties and provides a link to the associated topic that describes how to use it:

Page Element	Description	Topic
Themes drop-down list	Define jQuery Mobile themes for filters, icons, buttons, and so on.	Defining jQuery Mobile Theme Properties

Common Pages Used to Define Element Properties

This table lists common pages used to define page element properties and provides links to the associated topics that describe how to use them. The pages are listed alphabetically.

Page	Description	Topic
Image Selector page	Select images.	Using the Image Selector Page
Select Primitive page	Select primitive elements.	Using the Select Primitive Page

Defining jQuery Mobile Theme Properties

When defining page element properties oftentimes you can define jQuery Mobile themes for filters, icons, buttons, and so on.

PeopleSoft delivers jQuery Mobile theme swatches *a* through *e*.

Letter	Description	Color
a	Highest level of visual priority.	Black.
b	Secondary level of visual priority.	Blue.
c	Baseline level.	Silver.
d	Alternate secondary level.	Gray.
e	Accent.	Yellow.

You can also create custom themes using jQuery ThemeRoller. To make those themes available in a layout, add the CSS generated by jQuery ThemeRoller to the Layout Designer – CSS page and add the appropriate theme letter. The theme becomes available to select from the Theme drop-down list on all page element property pages where you can define jQuery Mobile themes.

Applying Template Containers and Selector Style Classes

This topic describes how to apply template containers and selector style classes.

Understanding Applying Template Containers and Selector Style Classes to Page Elements

To apply styles defined in a template to a page element, you must specify the container. Or, you can apply multiple classes from different containers to a page element.

Note that for any page element you can apply only one container or one or more selector classes.

After you select the container or classes to apply to a page element, the container or class names appear on the Layout Designer – Layout page in the in the Class Name field in the Layout grid.

Using Class Selection Controls

Every page element properties page in the Mobile Application Platform Layout Designer features a Class Selection section that features controls for applying layout template containers and selector classes to page elements.

To access the properties page for a layout page element, select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid, click the Properties link next to a page element.

Image: Class Selection controls for page element properties

This example illustrates the fields and controls in the Class Selection section for page element properties. You can find definitions for the fields and controls later on this page.

If there are multiple containers and selector classes defined for a template, use the View All and/or the page forward and back buttons to view and select them.

Template Name	Displays the name of the template applied to the page.
Sub Template Classes	This link appears only when one or more sub-templates are defined in the template that is applied to the layout page. Click the link to access the Select Sub Template Class Information

page and to select classes from a sub-template to apply to the element.

The Select Sub Template Class Information page is described later in this topic.

Select	Select the box to select the container or class.
Class Name	Displays the container class name as defined on the layout template.
View Container Summary	Click the link to access the Container Summary page and to view details about the classes defined for a container.
Comments	Displays comments for containers or classes as defined in the layout template.
Class	Displays the selector class name defined in the selected container.
View Class Summary	Click the link to access the Class Summary page and to view details about the selector classed defined for the selected container.
Add Class	After you select a containers or a selector class to apply to the page, click the Add Class button.
Clear	Click the button to clear selected classes from the Multiple Class Workspace area.
Class Name	This field displays the name of the class selected.
Style	This field displays the style(s) defined in the class.

Using the Select Sub Template Class Information Page

Use the Select Sub Template Class Information page (IB_DOCCONTSEL_SEC) to select containers and styles in a sub-template defined in a container to apply to the page element.

To access the page, click the Sub Template Classes link on the Class Selection section of properties page for a page element.

Image: Select Sub Template Class Information page

This example illustrates the fields and controls on the Select Sub Template Class Information page.

When you select a sub-template from the Sub Template drop-down list, the containers and selector classes appear on the page.

The fields and controls on the page function the same as those in the Class Selection section of the properties page for a page element. See Using Class Selection Controls earlier in this topic for descriptions of the fields and controls on this page.

Applying Layout Template Container Classes to Page Elements

To apply a layout template container class to a page element:

1. Navigate to the properties page for the element.

   Select PeopleToolsMobile Application Platform >Layout Designer , and in the Layout grid click the Properties link.

   If applying container classes from a sub-template, go to Step 2, otherwise, go to Step 3.

2. Click the Sub Template Classes link. The Select Sub Template Class Information page appears.

3. In the Containers area of the Class Selection section, locate the container that contains the styles to apply to the page element.

   Use the View All or page forward and page back buttons to view all containers defined in a template or sub-template. You can also click the View Container Summary link to view details about the containers defined in a template or sub-template.

4. Check the Select box for the container styles to apply to the page element.

   If applying container classes from a sub-template, go to Step 5, otherwise, go to Step 6.

5. Click theOK button at the bottom of the page. The properties page for the element appears.

6. Click the OK button at the bottom of the page. The Layout Designer – Layout page appears. The container class appears in the Class Name field for the page element.

7. Click the Save button.

Applying Multiple Layout Template Selector Classes to Page Elements

This topic describes how to apply multiple layout template and sub-template selector classes to page elements.

To apply multiple layout template selector classes to a page element:

1. Navigate to the properties page for the element.

   Select PeopleToolsMobile Application Platform >Layout Designer , and in the Layout grid click the Properties link.

   If applying selector classes from a sub-template, go to Step 2, otherwise, go to Step 3.

2. Click the Sub Template Classes link. The Select Sub Template Class Information page appears.

3. In the Containers area of the Class Selection section, locate the container that contains the selector classes to apply to the page element.

   • In the Containers section, use the View All or page forward and page back buttons to view all containers defined in the template or sub-template. You can also click the View Container Summary link to view details about the containers defined in a template sub-template.

   • In the Classes section, use the View All or page forward and page buttons to view all the selector classes defined for a container. You can also click the View Class Summary button to view details about selector classes defined in a container.

4. Check the Select box a selector class to apply to the page element.

5. In the Multiple Class Workspace section, click the Add Class button. The class name and styles defined appear in the section.

6. Repeat Steps 4-5 for each selector class to apply to the element.

   When finished, go to Step 7 if applying selector classes from a sub-template, otherwise, go to Step 8.

7. Click the OK button at the bottom of the page. The properties page for the element appears.

8. Click the OK button at the bottom of the page. The Layout Designer – Layout page appears. The selector classes appear in the Class Name field in the Layout grid for the page element.

9. Click the Save button.

Defining JavaScript Event Page Element Properties

This section describes how to define JavaScript events as page element properties in the Mobile Application Platform.

Understanding Defining JavaScript Event Page Element Properties

This section provides an overview of defining JavaScript events in the Mobile Application Platform.

Page Elements for JavaScript Events

You can add JavaScript events to enhance your mobile application for many page elements, including these:

*   Check box.

*   Container.

*   Flip toggle.

*   HTML area.

*   Image/Static text.

*   Input.

*   List view.

*   Mobile grid.

*   Push button.

*   Radio button.

*   Select box.

*   Slider.

*   Static text.

*   URL.

Default JavaScript Events

These JavaScript events are provided by default:

*   onblur

*   onchange

*   onclick

*   onbclick

*   onfocus

- onkeydown

- onkeypress

- onkeyup

- onmousedown

- onmousemove

- monouseout

- onmouseover

- onmouseup

- onselect

Defining JavaScript Events for Page Element Properties

To define a JavaScript event as a page element property, code the JavaScript on the Layout Designer – JavaScript page and then define the JavaScript event and function name on the properties page for the element.

As an example, you could define the following JavaScript on the Layout Designer – JavaScript page.

Image: Layout Designer – JavaScript page

This example illustrates sample JavaScript code in the Layout Designer – JavaScript page.
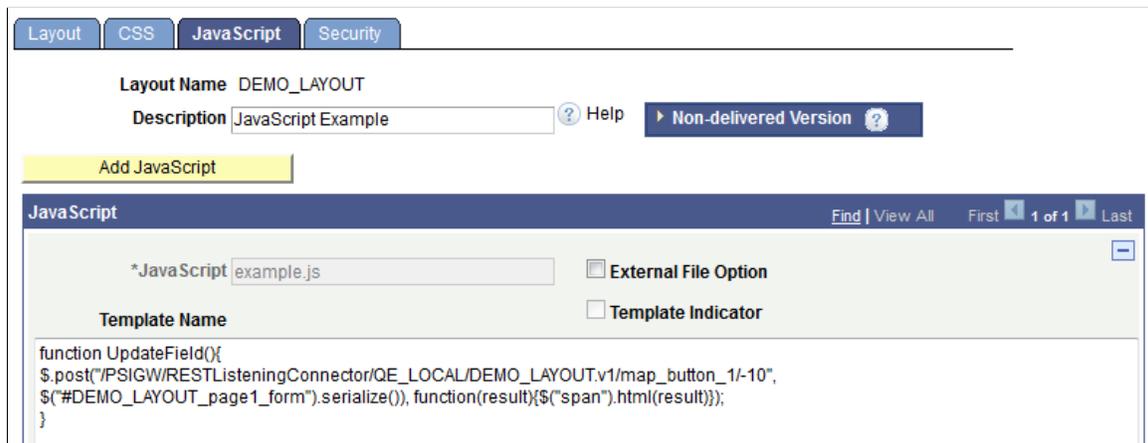
Then on the properties page for the element, you define the JavaScript event and JavaScript value in the JavaScript Event section shown in this example.

Image: JavaScript Events section of the page element properties page

This example illustrates the JavaScript Events section of a page element properties page, with a JavaScript event and value defined.

In this example, the JavaScript event is *onclick* and the value is the function name, *UpdateField(),* that you coded on the Layout Designer – JavaScript page. In the example shown, no arguments are passed. Be sure to add the argument list to a function, as necessary.

To define a JavaScript event for a page element property:

1. Write the JavaScript function on the Layout Designer – JavaScript page (PeopleTools >Mobile Application Platform >Layout Designer and click the JavaScript tab) or ensure that it is defined on a template that is assigned to the layout or page.

2. Access the page properties for the page element for which you want to define the JavaScript.

    a. On the Layout Designer, click the Layout tab.

    b. In the Layout grid, click the Properties link for the page element.

3. In the JavaScript Events section, select an event from the JavaScript Event drop-down list

4. In the JavaScript Value field, enter the name of the JavaScript function.

5. Click the Add a Row button (+) to add another row to define additional JavaScript.

6. Click the OK button.

    The Layout Designer - Layout page appears.

Defining Ajax Event Page Element Properties

This topic provides an overview of defining Ajax events as layout page element properties and describes how to:

•   Define Ajax events for page element properties.

•   Pass parameters in Ajax events.

•   Perform partial-page refreshes and updates using Ajax events.

This topic also provides an example of defining an Ajax event to update a field on a page.

Understanding Defining Ajax Events

The Ajax event is a general purpose event which allows you to pass any arguments to the server and return any data in a format of HTML, JSON, or XML.

An Ajax event can be invoked using a POST or GET and can be tied to any page element where you can define JavaScript events.

The Understanding Defining JavaScript Event Page Element Properties topic lists the page elements where you can define JavaScript events.

Ajax Events and Partial Page Refreshes and Updates

Using Ajax events allows you to perform partial page refreshes and updates very easily. For example the following available Mobile Application Platform APIs can be used:

- GenLayoutHTML - enables you to get a snippet of generated HTML for the same layout or any other layout to pass back to the client.

  You can also send back a snippet of HTML defined within a container on a page.

- GenLayoutDoc - enables you to call a layout that returns the Document object that was used to invoked the MAP application. This allows you to run MAP application business logic and return simply the data generated instead of the actual HTML. This method can be used to call an AJAX event directly by populating the PAGE_ID and LAYOUT_ID with the correct parameters.

  Note: If the Content Type is HTML then the data will not properly parse when loading the Document as the Content Data must be either XML or JSON.

You can also send back a snippet of HTML defined within a container on a page.

Ajax Events and Serialization

When defining Ajax events you have the option to perform data serialization if data you are passing is not PeopleSoft document data, or have the framework perform the serialization into a Document within the PeopleCode event.

Input Data – Serialization	You can use the unique system-generated value in the Form ID field in your JavaScript to serialize all the inputs on the page.
Input Data – No Serialization	If you select no serialization for data input, the framework does not attempt to serialize the name/ value pairs from the JavaScript into the defined PeopleSoft document for the layout.
	To get the contents of the data passed, use the GetAJAXNonSerializedData method of the Map object. You can also get parsed input data by using the following methods of the Map object: GetNumberofAJAXParms, GetAJAXName, GetAJAXValue, GetAJAXValuebyName; and the AJAXUrl method of the MapElement object.
Output Data – No Serialization	If you select no serialization for output data the contents of what is passed back is returned as-is to the client. If you choose no serialization, then the contents passed back are dependent on the

content type you select in the Content Type field when defining the Ajax event in the properties page for the element.

Defining Ajax Events as Page Element Properties

To define an Ajax event as a page element property you must at a minimum perform the tasks shown in the following list. There is no particular order for completing them.

- Access the JavaScript Events section of the properties page for the element. Several fields appear in the Ajax Event section on the element properties page that are important when creating the JavaScript event:

  - Ajax URL – You must include the system-generated value that appears in this field in your JavaScript so it can call to invoke the Ajax event.

  - Ajax URL – You must include the system-generated value that appears in this field in your JavaScript so it can call to invoke the Ajax event.

- Create a JavaScript function using the Layout Designer – JavaScript page, or ensure that the function is defined in the template assigned to the layout or page via the Template Designer – JavaScript page.

- Use PeopleSoft Application Designer to create an application class to run when the Ajax event is called.

- Use the JavaScript Events section on the properties page for the element o define the event.

To access the JavaScript Events section on the properties page for a page element, select PeopleTools >Mobile Application Platform >Layout Designer, and in the Layout grid click the Properties link for the page element to which to add the Ajax event.

The Ajax Event section is collapsed by default. Click the arrow next to the box label to expand the section.

Image: Ajax Event section of the page element properties page

This example illustrates the fields and controls for defining Ajax events. You can find definitions for the fields and controls later on this page.

JavaScript Event	Select a JavaScript event from the drop-down list.
JavaScript Value	Enter the JavaScript function name. Remember to add the argument list to a function, as necessary.
Package, Path, Class ID, Method	Enter the details of the application class if you choose to have the system perform serialization of data or to perform partial page updates/refreshes.

•   Package. Enter the name of the application package that contains the method to invoke.

•   Path. Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.

•   Class ID. Enter the name of the application class that contains the method to invoke.

•   Method. From the drop-down list select the method to invoke.

Pending Approval	This field is currently not used and is reserved for future use.
Use Exception Layout	If you check this box and an Exception Layout is defined for the Layout, then if an exception is thrown the MAP framework will invoke the defined Exception Layout.
Content Type	From the drop-down list, select the format of any returned data. The options are:

•   *HTML*. (Default.)

- *JSON.*

- *XML.*

Input No Serialization	If you check this box the system does not perform serialization on input data.
Output No Serialization	If you check this box the system does not perform serialization on output data.
Container Label ID	Displays the system-generated container ID of the container you select using the Select Container link. Use the ID in your JavaScript code to send back HTML data contained in containers on a page when an Ajax event is invoked.
Select Container	Click the link to access the Select a Container Label page to choose the container from which to get data.
Ajax URL	Displays the system-generated URL that the JavaScript must call to invoke the Ajax event.
Form ID	Displays the system-generated ID to include in JavaScript to serialize Ajax event input data.

Passing Parameters in Ajax Events

To pass specific parameters in Ajax events use name value pairs with the name the ID of the element when coding the function on the Layout Designer – JavaScript page.

The following JavaScript code shows an example of passing parameters in an Ajax event:

```
$.post("/PSIGW/RESTListeningConnector/QE_LOCAL/DEMO_LAYOUT.v1/1/4", {
prim_string__1 : "John", prim_string__2 : "bill"}).done(function(data)
{  alert("Data Loaded: " + data);});
```

In the example *prim_string__1* and *prim_string__2* are the element IDs. This code would pop-up an alert with the returned data.

Performing Partial-Page Refreshes and Updates Using Ajax Events

To perform page refreshes and updates using Ajax events, create a JavaScript event calling the unique system-generated URL that appears in the Ajax URL field in the Ajax Event section in the properties page for the element. This will properly call the application class assigned to the event.

To send back a snippet of HTML that is defined within a defined container on the page, click the Select Container link in the Ajax Event section in the properties page for the element to select the container. The Select a container label page appears.

Image: Select a container label page

The Select a container label page displays all containers defined on the mobile application page.
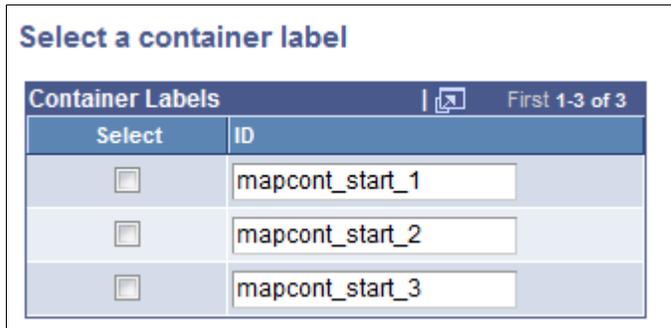
All of the containers defined on the page appear in the list.

To perform partial page refreshes and updates using Ajax events, following the steps outlined in the Defining Ajax Events as Page Element Properties section in this topic, with the following additional steps:

1. Select the container that contains the data to refresh/update.

   a. In the Ajax Events section on the page element properties page, click the Select Container link. The Select a container label page appears.

   b. Select the container that contains the data for partial page refresh/update or the HTML snippet to send back and click the OK button.

      The container ID appears in the Container Label ID field.

2. Use the Container Label ID value in your JavaScript code to send back HTML data contained in the container when the Ajax event is invoked.

3. When you define the Ajax event in the page element properties page, from the Content Type drop-down list, select *HTML*.

Example: Defining an Ajax Event to Update a Field on a Page

In the Ajax Events section of the properties page for a push button the following information is defined:

Image: Example: Ajax Events section

The Ajax Event section of the page illustrates sample data for defining an Ajax even to update a field on a page.

On the Layout Designer – JavaScript page, enter the following JavaScript code:

```
function UpdateField(){
$.post("/PSIGW/RESTListeningConnector/QE_LOCAL/DEMO_LAYOUT.v1/
map_button_1/-10", $("#DEMO_LAYOUT_page1_form").serialize()),
function(result){$("span").html(result)});
}
```

Defining jQuery Mobile Plug-In Properties

This topic discusses how to set and use jQuery mobile plug-ins.

Understanding jQuery Mobile Plug-In Properties

You can define jQuery mobile plug-in properties for the following mobile application platform page elements:

- Action sheet.

- Input.

- Check box.

- Container.

- Flip toggle.

- Footer.

- HTML area.

- Header.

- List view.

- Mobile grid.

- Push button.

- Radio button.

- Select box.

- Select menu.

- Slider.

- Table.

- URL.

Setting jQuery Mobile Plug-In Properties

For the case where a JavaScript adds new behavior to current jQuery Mobile widgets and/or core functionality, you can add data-* attributes to the type field. For example:

```
data-xxx="abc"
```

For example:

```
data-rel="back"
```

Use the Optional Plug-In section on the properties page for an element to add the attributes.

Image: Optional Plug-In section

This example illustration the Optional Plug-In section that appears properties page for an element.
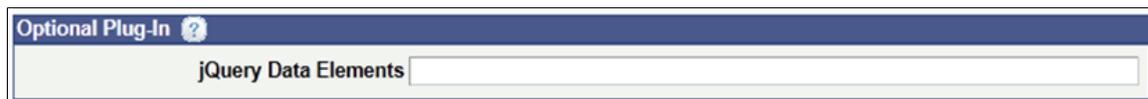
Using Third-Party jQuery Mobile Plug-Ins

To use third-party jQuery Mobile plug-ins in a Mobile Application Platform application, perform the following steps:

1. Add Java Script (if applicable) to the Layout Designer - JavaScript page.

   To access the page select PeopleTools >Mobile Application Platform >Layout Designer and click the JavaScript tab.

2. Add CSS (if applicable) to the Layout Designer - CSS page.

   To access the page select PeopleTools >Mobile Application Platform >Layout Designer and click the CSS tab.

3. Add the data attribute to the jQuery Data Elements field in the Optional Plug-In box on the properties page for the element, if applicable.

4. Use the element type *HTML Area* to add any specific HTML need to support the plug-in (if applicable).

Defining Static Text Properties

You can define static text properties when working with the following page elements:

•   Static Text.

•   Image/Static Text.

The properties page for static text and image/static text page elements features a Static Text Properties section, where you can define class, style, and heading properties to static text data.

Note: To apply properties to the *label* for static text, use the Class Name, Style, and Theme fields in the Layout grid on the Layout Designer page.

Image: Static Text Properties box

This example illustrates the fields and controls on the Static Text Properties box. You can find definitions for the fields and controls later on this page.

Class	(Optional.) Enter the class name to apply to the static text data for the page element.
Style	(Optional.) Enter the style name to apply to the static text data for the page element.
Heading Text	(Optional.) From the drop-down list, select the heading level for the static text data for the page element. The values are:

•   *h1.* Applies a Heading 1 style to the text data.

•   *h2.* Applies a Heading 2 style to the text data.

•   *h3.* Applies a Heading 3 style to the text data.

•   *h4.* Applies a Heading 4 style to the text data.

•   *h5.* Applies a Heading 5 style to the text data.

•   *h6.* Applies a Heading 6 style to the text data.

Defining Image Properties

This topic discusses how to:

- Define dynamic images for a page element.

- Define static images for a page element.

Understanding Defining Image Properties

You can define image properties when working with the following page elements:

- Image/static text.

- URL.

- Action sheet.

- Push button.

You can use define static or dynamic images.

To define a static image use the Image Selector page to select an image from the database. To specify a dynamic image you use the Select Primitive page to select a primitive element defined in PeopleSoft document defined for the layout.

Using the Image Properties Box

The Image Properties box appears on the properties page for those element for which you can define images.

Image: Image Properties (Dynamic image properties)

This example illustrates the fields and controls on the Image Properties box when defining dynamic image properties. You can find definitions for the fields and controls later on this page.

Image Properties		
☐ Add Image		Image Name
Dynamic Image Primitive Selection		Element Name
Type		
Primitive	Select Primitive	

The previous example shows the default view of the Image Properties box. This view features the fields and controls to define a dynamic image.

Click the Select Primitive link to access the Select Primitive page to select a primitive element for the image.

Image: Image Properties (Static image properties)

This example illustrates the fields and controls on the Image Properties box when defining static image properties. You can find definitions for the fields and controls later on this page.
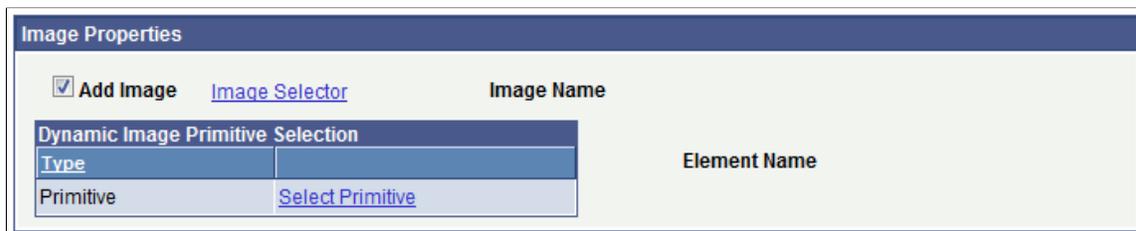
The previous example shows the fields and controls on the Image Properties box when defining static image properties. When you select the Add Image box, the Image Selector link appears. Click the link to access the Image Selector page to select an image defined in the database.

Add Image	Select the box to define a static image.
Image Selector	This link appears after you select the Add Image box.
	Click the link to access the Image Selector page to select an image from the database.
Image Name	Displays the name of the image selected using the Image Selector page.
	After you select an image using the Image Selector page, a thumbnail of the image appears to the right of the image name.
Select Primitive	Click the link to access the Select Primitive page to select a dynamic image for the page element.
	The primitives available to select are those defined in the PeopleSoft document defined for the layout.
Element Name	Displays the name of the primitive element selected using the Select Primitive page.

Defining Dynamic Image Properties

To define a dynamic image for a page element property:

1. Access the page element properties pageAccessing Page Element Properties

2. In the Image Properties box, click the Select Primitive link.

   The Select Primitive page appears.

3. Select the primitive to which the image you want to use is mapped.

4. Click the OK button.

The properties page for the page element appears and the name of the element appears in the Element Name field.

Defining Static Image Properties

To define a static image for a page element property:

1. Access the page element properties page<u>Accessing Page Element Properties</u>

2. In the Image Properties box, select the Add Image box.

    The Image Selector page appears.

3. Select an image.

4. Click the OK button.

The properties page for the page element appears. The name of the image appears in the Image Name field and a thumbnail of the image appears next to the name.

Defining Accessibility Properties

You can define alternate text for images to meet accessibility guidelines when working with the following page elements:

*   Image/Static text.

*   Action sheet.

An Accessibility section appears on the properties page for these elements where you can define alternate text for images.

Image: Accessibility properties

This example illustrates the fields and controls of the Accessibility properties.
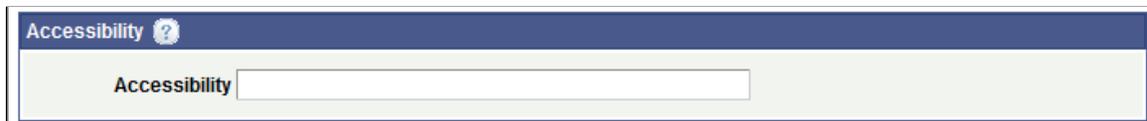
To define alternate text:

1. Access the page elements properties page. <u>Accessing Page Element Properties</u>for the page element.

2. In the Accessibility field enter alternate text.

3. Click the OK button.

Defining URL Properties

This topic provides an overview of viewing, defining, and using URL properties.

Understanding Defining URL Properties

You can define URL properties for the following page elements:

- URL.

- Push button.

- List view.

A URL Properties section appears on the properties page for these elements. The properties that you can define depend on the URL type defined for the page element, for example an external URL, a URL to a REST service, a URL to a layout, and so on, as well as the layout page type (internal and external).

Related Links

Accessing Page Element Properties

Viewing URL Properties for External URLs

Image: URL Properties (External URL)

This example illustrates the fields and controls of URL Properties when the page element is to an external URL. You can find definitions for the fields and controls later on this page.

Viewing URL Properties for URLs to External Layout Pages

Image: URL Properties (URL to External layout page)

This example illustrates the fields and controls of the URL Properties when the page element URL is to a layout page that is defined as an *External* page type. You can find definitions for the fields and controls later on this page.
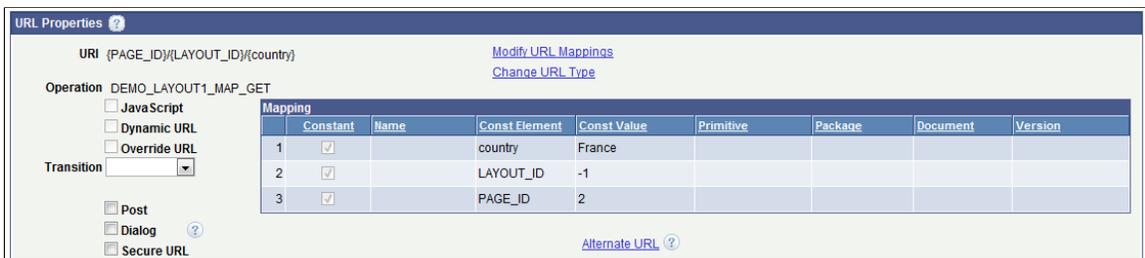
Viewing URL Properties for URLs to Internal Layout Pages

Image: URL Properties (URL to Internal layout page)

This example illustrates the fields and controls of the URL Properties when the page element URL is to a layout page that is defined as an *Internal* page type. You can find definitions for the fields and controls later on this page.

Viewing URL Properties for URLs to REST Services and to Other Layouts

Image: URL Properties (URL to REST service or to another layout)

This example illustrates the fields and controls of the URL Properties when the page element URL is to a REST service or to another layout. You can find definitions for the fields and controls later on this page.

Viewing URL Properties for URLs to PeopleCode Update Events

Image: URL Properties (URL to Execute Update Event)

This example illustrates the fields and controls of the URL Properties when the page element URL is to execute a PeopleCode Update event. You can find definitions for the fields and controls later on this page.

In addition to URL properties, you can also define properties related to the PeopleCode Update event using the Update Event section on the URL Properties page.

Image: Update Event properties

This example illustrates the fields and controls in the Update Event section of the URL Properties page when defining a URL to execute an Update event.. You can find definitions for the fields and controls later on this page.

The fields and controls in the Update Event section appear on the URL Properties page only when you are working with properties for an Update URL.

Defining URL Properties for Page Elements

The following table describes URL property fields and controls. As described previously, the fields and controls that appear when working with a URL page element depend on the URL type and the page type.

The properties are listed in alphabetical order:

Alternate URL	Click the link to access the MAP Alternate Primitive page to define a JavaScript function to call an alternative URL in the case of changing form factors for that specific element. See Using the MAP Alternate URL Primitive Page
Change URL Type	Click the link to access the Change URL Type page to change the URL type. You can change URL types for and to these page and URL types: • Internal page URLs. • External page URLs. • Update URLs. See Using the Change URL Type Page
Clear Mapped URI	Clear the currently selected URI. When you click the link the system clears the values that appear in the Mapping grid.

	Clearing this value is done in advance of selecting a different URI. Click the MAP to URI link to access the Select URI page to select a different URI.
Dialog	Select the box to show the target URL content in a modal page.
	The intent is for modal messages, lists or other information that does not have any hierarchal relation with the source page.
	Note the following points about modal pages:

- Modal pages feature a close action button (X) at the top left corner where the Back action appears on non-modal pages.

- A margin conveying to the user that the modal page is not a full-screen page, but rather a pop-up window.

- Modal pages do not appear on the navigation stack as new pages.

Dynamic URL	Select this option to determine the complete URL at user selection instead of at page initialization.
	Make sure to save the layout for any changes to layout initialization parameters in order to be able to view all available initialization parameters.
	The primitives available for selection (mapping) must be defined on the Layout Designer – Layout page as a Check Box, Radio, Input, Flip Toggle, Slider, or Select Box field type.

> **Note:** When calling a page using a dynamic URL the framework treats the link as an absolute external link, and therefore hides the automated Visual back button (if defined) and disables any links created via Layout page on the page being called.

JavaScript	Select this option to invoke a JavaScript function instead of the URL. The JavaScript function must be defined as a primitive element. You then select the primitive element using the MAP Alternate URL Primitive page.
	See Using the MAP Alternate URL Primitive Page
Map to an Internal Page	This link appears as a URL property when defining a URL on an Internal layout page type.
	Click the link to access the Map to Internal Page page to define a link to another Internal type layout page.
	See Using the MAP to Internal Page
Map to URI	Click the link to access the Select URI page to select a URI.

	See <u>Using the Select URI Page</u>
Mapping grid	Display information about the current URI mapped for the URL.
Modify URL Mappings	Click the link to access modify the value of a URL or URI. The page that appears depends on the type of URL/URI and the page type with which you are working as follows:
	• External URL — Modify URL page.
	• URL to a REST service — URI Selection page.
	• URL to another layout — Select URI page.
	• URL to a External type layout page — Select URI page.
	• URL to an Internal type layout page — Select URI page.
	• URL to a PeopleCode Update event — N/A.
	See <u>Using the Modify URL Page</u><u>Using the URI Selection Page</u><u>Using the Select URI Page</u>
Operation	When defining a URL to a REST service this link displays the name of the REST service.
	When defining a URL to a layout or layout page, this field displays the URI string selected when you defined the REST service operation.
Override URL	This control is read-only.
	This control is selected if the default URL has been overridden.
Pending Approval	This field is reserved for future use.
Post	When you select this property the request is made via an HTTP POST.
	Select this property if you are passing .sensitive arguments as part of the layout URI.
	Note: Dynamic URLs do not work if this property is selected.
Same Window	Select the box so that when the URL is click the content loads in the same window.
	If you enable this option, ensure that the loaded page has an appropriate back action so that the user can return to your application.
	If you leave this option clear, the system maintains the instance of the current page while opening the selected link in a different window.

Secure URL	Select this property for cases where the URL is constructed from a provider REST service or layout if applicable (secure REST target location defined).
Transition	From the drop-down list, select the transition to use to the layout page. The values are:

- *Fade.* Fade the page or dialog in over the previous content.

- *Flip.* An animated page flip, rotating the current view out with the other view on the reverse side.

- *Flow.* Throws the current page away and comes in with the next page.

- *Pop.* The page springs into view from the center of the screen.

- *Slide.* Slide in from the left or right, pushing the previous content out of the way.

- *Slidedown.* Slide down from the top, over the top of the current container.

- *Slidefade.* Slide in from the left or right then fade the page.

- *Slideup.* Slide up to the top, revealing the next content below.

- *Turn.* The page turns as in turning a page in a book.

URI	This field displays the URI defined for the page element.
	If in the URL properties you modify the URL mappings, define an alternate URL, override a URL, and so on, the changes you make are reflected in the URI that displays in the field.
URL	This field displays the current URL for the page element.
	If the URL is a static URL, the value appears in the standard URL form:

```
http://www.example.com
```

If the URL is a dynamic URL, the field displays the following:

```
Mapped to primitive <value>
```

If in the URL properties you modify the URL mappings, define an alternate URL, override a URL, and so on, the changes you make are reflected in the URL that displays in the field.

Using the MAP Alternate URL Primitive Page

Use the MAP Alternate URL Primitive page (IB_DOCALTURL_SEC) to define an alternate URL using a primitive element.

Image: Map Alternate URL Primitive page (Default)

This example illustrates the fields and controls on the default view of the Map Alternate URL Primitive page. You can find definitions for the fields and controls later on this page.

Click the Select Primitive link to access the Select Primitive page to select a new primitive element to which to map. After you select an element, the element name appears in the Mapped Element field.

The following example shows the name of the primitive element *Pilot* in the Mapped Element field, indicating that it is mapped as an alternate URL primitive element.

Image: Map Alternate URL Primitive (Alternate primitive value defined)

This example illustrates the fields and controls on the Map Alternate URL Primitive when an alternate primitive value is defined. You can find definitions for the fields and controls later on this page.

To delete an alternate URL primitive, click the Remove Mapping button.

Mapped Element	This field displays the name of the alternate URL primitive value.
Remove Mapping	Click the button to remove the alternate URL primitive.
Select Primitive	Click the link to access the Select Primitive page to select an alternate URL primitive.

Using the Change URL Type Page

Use the Change URL Type page (IB_DOCURLCHNGE_SEC) to change the URL type of the target URL.

To access the page, click the Change URL Type link on the URL properties page for a URL page element. The link appears in the URL properties when you are working with the following page and URL types:

- External page URLs.

- Internal page URLs.

- Update URLs.

In addition, you can change the URL to one of the preceding page/URL types.

For example, if a URL type was initially defined as *External,* you can change the type to *Internal* or *Update.*

The one exception is for the list view element where the only URL types can change between *External* and *Update.*

Note that the page type (*Internal, External*) is defined on the Page Properties page for the layout.

See Using the Page Details Page

Image: Change URL Type page

This example illustrates the fields and controls on the Change URL Type page. You can find definitions for the fields and controls later on this page.

| **Current URL Type** | This field displays the current URL type. |
| **Target URL Type** | Select the new URL type from the drop-down list. The list of values depends on the existing URL type, and may include: |

- External URL.

- Internal URL.

- Update URL.

Using the MAP to Internal Page

Use the MAP to Internal page (IB_DOCLOPAGSEL_SEC) when working with a URL on an Internal layout page type to map to another Internal type layout page in the same layout.

Image: Map to Internal Page page

This example illustrates the fields and controls on the Map to Internal Page page. You can find definitions for the fields and controls later on this page.

Using the Modify URL Page

Use the Modify URL page (IB_URLMOD_SEC) to modify the value of an external URL. You can modify a URL with a static or dynamic value.

Image: Modify URL page

This example illustrates the fields and controls on the default view of the Modify URL page. You can find definitions for the fields and controls later on this page.
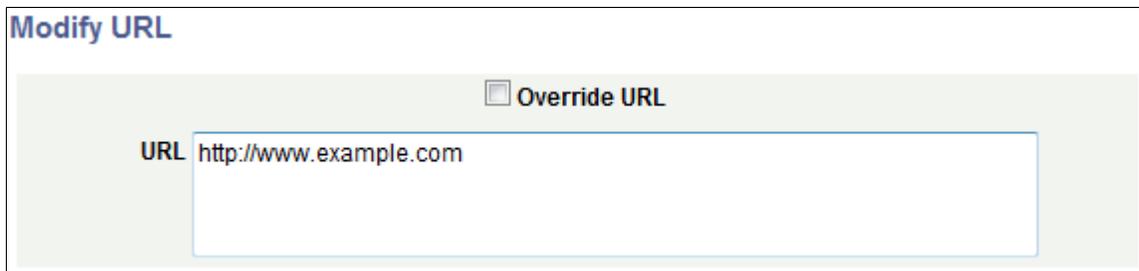
Use the default view of the Modify URL page, shown in the previous example, to modify an external URL with a static value.

To specify a dynamic URL, select the Override URL box . You then click the Select primitive for mapping link to access the Select Primitive page to select a primitive to which to map. After you select a primitive element and return to the Modify URL field, the name of the primitive element appears in the Element Name field and the URL field displays the message:

```
Mapped to primitive <name>
```

Image: Modify URL page (Dynamic URL override)

This example illustrates the fields and controls on the Modify URL page when using a dynamic URL override. You can find definitions for the fields and controls later on this page.
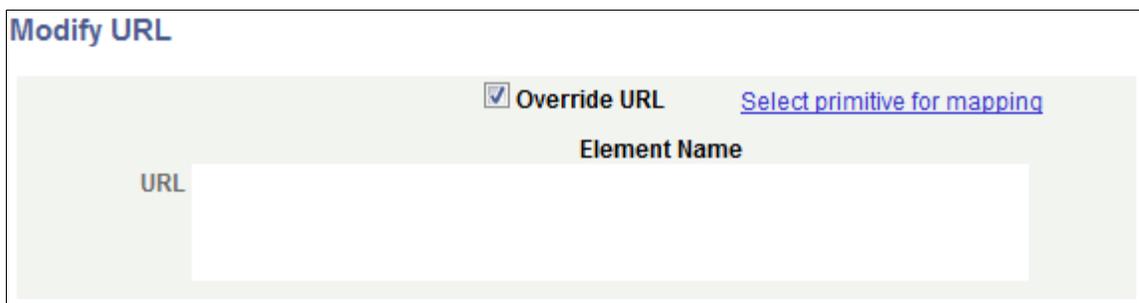
As with other dynamic page elements, the primitive values from which you can select are

The previous example shows

Using the Select URI Page

Use the Select URI page (IB_DOCLOLAYOT2_SEC) to modify URL mappings for URLs to other layouts and for URLs to other *Internal* and *External* layout page types.

To access the page, click the *Modify URL Mappings* link in the URL Properties section of the page element properties for the element.

Image: Select URI page (Default view)

This example illustrates the fields and controls on the default view of the Select URI page. You can find definitions for the fields and controls later on this page.

The previous example shows the default view of the Select URI page. After you enter a layout name the system populates the Layout URI List grid with the URIs defined as initialization parameters for the layout. You can then select a URI for the modified URL.

Image: Select URI page (Layout URI List grid populated)

This example illustrates the fields and controls on the Select URI page when the Layout URI List grid is populated. You can find definitions for the fields and controls later on this page.

If you select the default initialization URI for a layout, you need not set any additional parameters. However, if you select any other URI in the grid, a URI Parameters grid appears for you to define values for the parameters in the URI.

Image: Select URI page (Layout URI List grid and URI Parameters grid populated)

This example illustrates the fields and controls on the Select URI page when the Layout URI List grid and URI Parameters grid are populated. You can find definitions for the fields and controls later on this page.
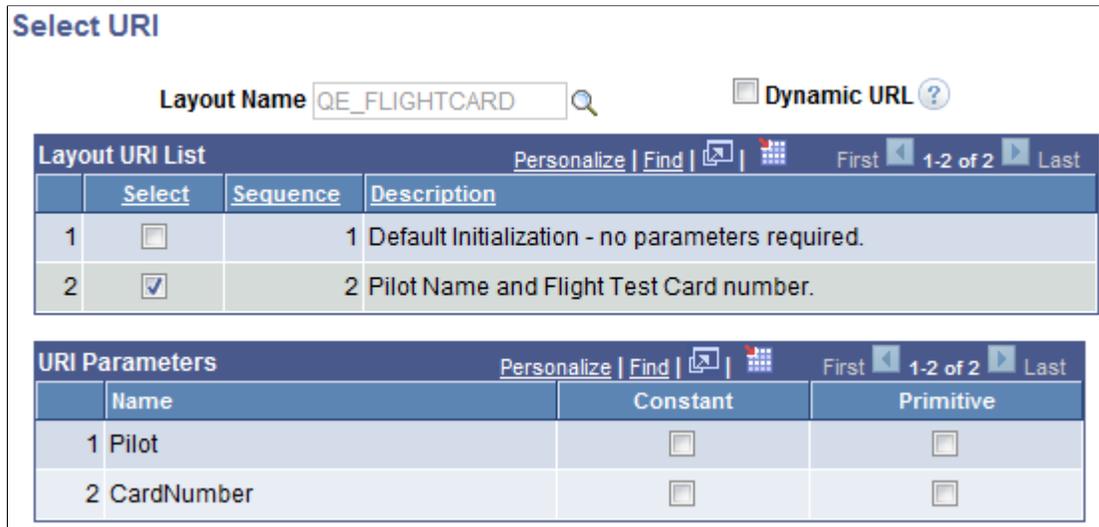
In the URI Parameters grid, select the parameters to include in the URI.

You can define constant or primitive values for URI parameter values. When you select the Constant control, a Const Value field appears where you enter the constant value. When you select the Primitive control, a Select Primitive link appears to access the Select Primitive page to define a primitive for the URI parameter.

Image: Select URI page (Defining URI Parameters)

This example illustrates the fields and controls that appear when defining URI parameters in the URI Parameters grid. You can find definitions for the fields and controls later on this page.
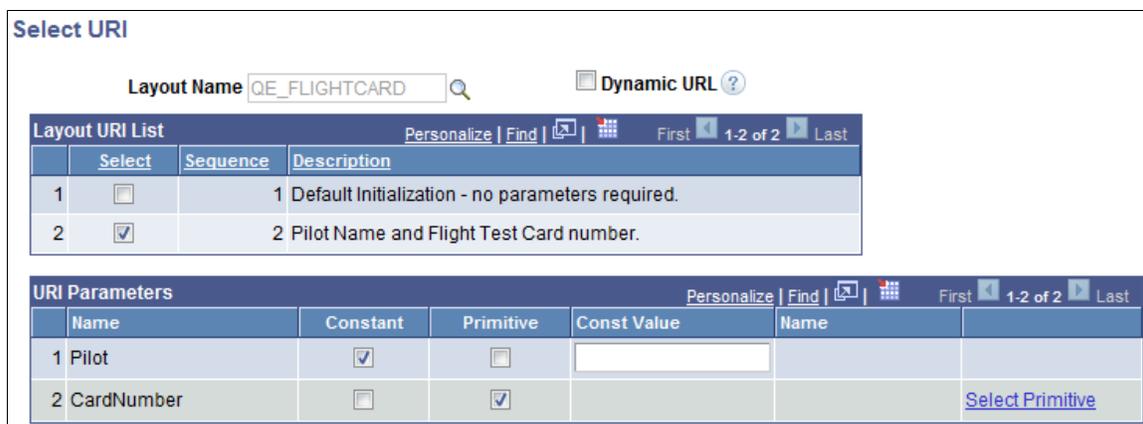
Layout Name	Enter of search for the layout that contains the URIs to which you want to map.
Dynamic URL	Select the box to execute the URL when the user selects it, instead of at page initialization.

The primitives available for selection (mapping) must be defined on the layout page as a check box, radio button, input, flip toggle, slider, or select box.

When calling a page using a dynamic URL, the framework treats the link as an absolute external link, and hides the Back button and disables any links created on the page being called.

Select	Select the box to select the corresponding URI in the Layout URI List grid.
Sequence	Indicates the sequence of the URI defined in the layout initialization parameters for the layout named in the Layout Name field.
Description	Displays the description of the URI as defined in the layout initialization parameters for the layout named in the Layout Name field.
(URI Parameter) **Name**	Displays the name of the URI parameter.
Constant	Select the control to define a constant value for the parameter. When you select the control, a Const Value field appears where you can enter the constant value.
Primitive	Select the control to define a primitive value for the parameter. When you select the control, a Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive value.
Const Value	This field appears when you select the Constant box. Enter a constant value for the URI parameter.
(Primitive) **Name**	If you define a primitive value for a parameter, the name of the primitive appears in the field.
Select Primitive	This link appears when you select the Primitive control in the URI Parameters grid. Click the link to access the Select Primitive page to define a primitive value for the parameter.

Using the URI Selection Page

Use the URI Selection page (IB_RESTSERV_SEC) to modify URL mappings for URLs to REST services.

To access the page, click the *Modify URL Mappings* link in the URL Properties section of the page element properties for the element.

Image: URI Selection page (Default view)

This example illustrates the fields and controls on the default view of the URI Selection page. You can find definitions for the fields and controls later on this page.

The previous example shows the default view of the URI Selection page. After you enter a REST service operation name the system populates the REST Operations grid with the URIs defined as initialization parameters defined for the layout. You can then select a URI for the modified URL.

Image: URI Selection page (REST Operations grid populated)

This example illustrates the fields and controls on the URI Selection page when the REST Operations grid is populated. You can find definitions for the fields and controls later on this page.
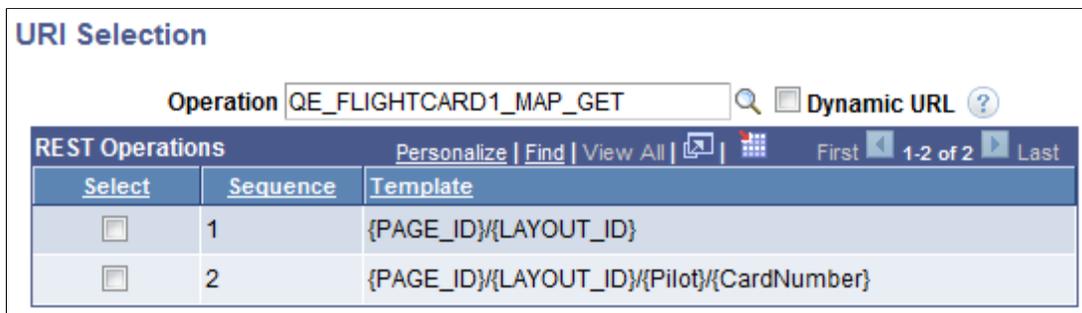
The REST operations grid displays the URIs defined as initialization parameters for the layout. Select the URI to which to link.

When you select a URI, the URI Parameters grid appears on the page, as shown in the following example.

Image: URI Selection page (REST Operations grid and URI Parameters grid populated)

This example illustrates the fields and controls on the URI Selection page when the REST Operations grid and the URI Parameters grid are populated. You can find definitions for the fields and controls later on this page.

In the URI Parameters grid, define the parameter values for the URI.

You can define constant or primitive values for the URI parameters. When you select the Constant control, a Const Value field appears where you enter the constant value. When you select the Primitive control, a Select Primitive link appears to access the Select Primitive page to define a primitive for the parameter.

Image: URI Selection page (Defining URI Parameters)

This example illustrates the fields and controls on the URI Selection page when defining URI parameters in the URI Parameters grid. You can find definitions for the fields and controls later on this page.
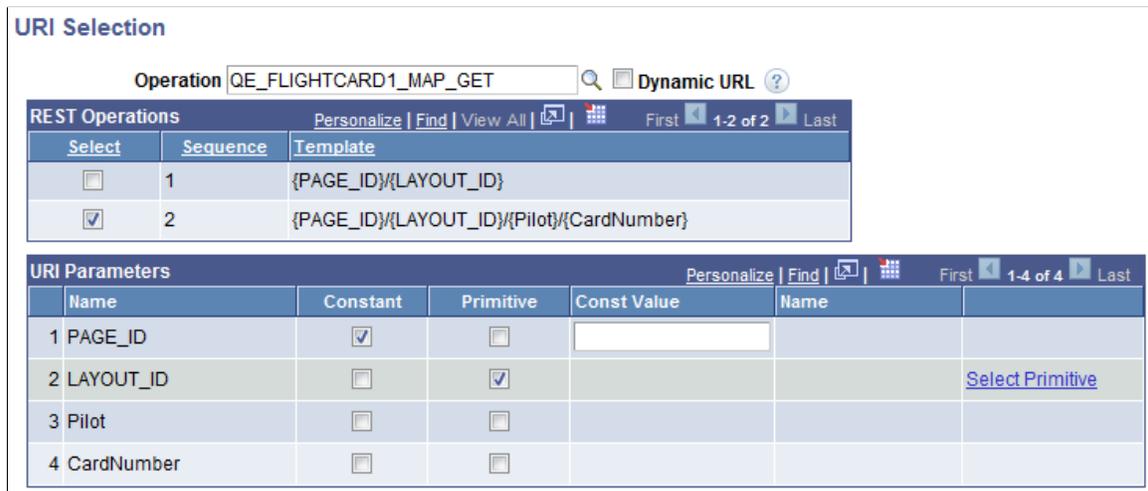
Operation Name	Enter of search for the MAP service operation that contains the URIs to which you want to map.

Dynamic URL	Select the box to execute the URL when the user selects it, instead of at page initialization.
	The primitives available for selection (mapping) must be defined on the layout page as a check box, radio button, input, flip toggle, slider, or select box.
	When calling a page using a dynamic URL, the framework treats the link as an absolute external link, and hides the Back button and disables any links created on the page being called.
Select	Select the box to select the corresponding URI in the REST Operations grid.
Sequence	Indicates the sequence for the selected URI, as defined in the in the service operation named in theOperation field.
Template	Displays a service operation template defined in the service operation named in the Operation field.
(URI Parameter) **Name**	Displays the name of the URI parameter.
Constant	Select the control to define a constant value for the parameter. When you select the control, a Const Value field appears where you can enter the constant value.
Primitive	Select the control to define a primitive value for the parameter. When you select the control, a Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive value.
Const Value	This field appears when you select the Constant box.
	Enter a constant value for the URI parameter.
(Primitive) **Name**	If you define a primitive value for a parameter, the name of the primitive appears in the field.
Select Primitive	This link appears when you select the Primitive control in the URI Parameters grid. Click the link to access the Select Primitive page to define a primitive value for the parameter.

Using the Select Primitive Page

Use the Select Primitive page (IB_DOCLOINPUT_SEC2) to select a primitive element for a page element property, such as an image or URL.

Image: Select Primitive page

This example illustrates the fields and controls on the Select Primitive page. You can find definitions for the fields and controls later on this page.

The primitives that appear on the page are those defined in the PeopleSoft document defined for the layout.

To select a primitive element:

1. Access the page element properties page. Accessing Page Element Properties for the page element.

2. Select the Select box next to an element name.

3. Click the OK button.

Using the Image Selector Page

Use the Image Selector page (IB_IMAGESELECT_SEC) to select an image for a page element property.

Image: Image Selector page

This example illustrates the fields and controls on the Image Selector page. You can find definitions for the fields and controls later on this page.

Images				Personalize | Find | View 100 | First 81-90 of 1724 Last
	Select	Content Name	Descr	Image Icon
81	☐	MAP_ACTIONS	MAP OBJECT	
82	☐	MAP_ACTIONS_1	MAP OBJECT	
83	☐	MAP_ALERT		
84	☐	MAP_APPROVED	MAP OBJECT	
85	☐	MAP_APPROVED_01	MAP OBJECT	
86	☐	MAP_BACKBUTTON	MAP OBJECT	
87	☐	MAP_BACKIMAGE	MAP OBJECT	
88	☐	MAP_BANNER_BG	MAP OBJECT	
89	☐	MAP_CHART	MAP OBJECT	
90	☐	MAP_DELGATE	MAP OBJECT	

The images that appear in the grid are those added to the database in PeopleSoft Application Designer.

To select an image:

1. Access the page element properties page Accessing Page Element Properties for the page element.

2. Select the Select box next to a content name/image name.

3. Click the OK button.

Chapter 27

Managing Header and Footer Page Elements

Managing Header and Footer Page Elements

This topic discusses how to manage header and footer page elements in the Layout grid.

Understanding Header and Footer Page Elements

By default header and footer page elements are defined in the Layout grid on the Layout Designer – Layouts page for every page in a layout. As a result, there is no need to manually add them to a layout page.

You can insert the following page elements into header page elements and footer page elements:

- Action sheet.

- Container.

- HTML area.

- Image/static text.

- List view.

- Mobile grid.

- Push button.

- Sidebar. (Footer elements only.)

- Static text.

- URL.

Viewing Defined Header and Footer Page Elements in the Layout Grid

To view header and footer page elements, access the Layout grid on the Layout Designer – Layouts page. To access the page select PeopleTools >Mobile Application Platform >Layout Designer.

The header is designated by the *mapheader_n* and *mapheader_end* element IDs. The footer is designated by the *mapfooter_n* and the *mapfooter_end* element IDs.

Image: Layout grid (Default view)

This example illustrates the fields and controls on the default view of the Layout grid. By default, header and footer elements are inserted into the Layout grid on any page you create in the Layout Designer.

Any text you enter into the Label Text field for the *mapheader_n* or *mapfooter_n* elements appears in the application at runtime.

Omitting Headers and Footers in Application Pages

To omit a header in an application page:

* Do not insert any text in the Label Text field next to the *mapheader_n* element ID.

* Do not insert any page elements between the *mapheader_1* and *mapheader_end* element IDs.

Similarly, to omit a footer in an application page:

* Do not insert any text in the Label Text field next to the *mapfooter_1* element ID.

* Do not insert any page elements between the *mapfooter_1* and *mapfooter_end* element IDs.

Defining Header and Footer Page Element Properties

This section discusses how to define header and footer page element properties.

Understanding Defining Header and Footer Page Element Properties

Many properties that you can define for header and footer page elements are also available to define for other page elements. These common properties are described in the topic Common Layout Page Element Properties

This topic discusses the properties that are unique to header and footer page elements only.

Defining Header and Footer Page Element Properties

Use the Header Properties page to define header page element properties and the Footer Properties page to define footer page element properties.

To access these pages:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

   The Layout Designer appears.

2. To access the Header Properties page or Footer Properties page:

Header Properties page	In the Layout grid click the Properties link next to the *mapheader* element ID.
Footer Properties page	In the Layout grid click the Properties link next to the *mapfooter* element ID.

Image: Header Properties section

This example illustrates the fields and controls in the Header Properties section of the Header Properties page. You can find definitions for the fields and controls later on this page.

Image: Footer Properties section

This example illustrates the fields and controls on the Footer Properties section. You can find definitions for the fields and controls later on this page.

Fixed Display Format	If selected as a header property, the header remains at the top of the view port as the user scrolls.
	If selected a footer property, the footer remains at the bottom of the view port as the user scrolls.
Show Navigation Bar	If selected as a header property, a navigation bar appears in the header.
	If selected as a footer property, a navigation bar appears in the footer.
	When you select this property, the system enables the property on all pages of the layout and the navigation bar appears on all pages. You can clear this option on any pages that you don't want the bar to appear.
	To call other pages via a URL or button instead of using the header or footer navigation bar, when defining the URL or button use the Layout Page option when defining the link. Be sure to include a link back to the calling page. When omitting a navigation bar in a header, you can enable the Visual Back Button property on the Page Details page of a page to have a link back to calling page accordingly.
Full Screen	If selected as a header property, the header appears within the viewport and stays present as the user scrolls, regardless of where the user is in the content.

	If selected as a footer property, the footer appears within the viewport and stays present as the user scrolls, regardless of where the user is in the content.
Multi-Page Navigation Bar	For layouts with multiple *Internal* type pages defined, the pages are shown as buttons on the header toolbar when you select this property as a header property or in the footer toolbar when you select this property as a footer property.
	The system uses the value defined in thePage Name field on the Page Details page of each page for the navigation button labels.
Equal Button Size	If you enable the Multi-Page Navigation Bar property, this property equalizes the display size of the navigation buttons.

Related Links

Using the Page Details Page
Defining URLs to Pages Within Layouts

Chapter 28

Managing Container Page Elements

Understanding Managing Container Page Elements

Use the container element to create a content container around other element types.

You can insert the following page elements into a container element:

- Action sheet.

- Check box.

- Containers.

- Flip toggle.

- HTML area.

- Image/Static text.

- Input.

- List view.

- Mobile grid.

- Push button.

- Radio button.

- Select box.

- Select menu.

- Slider.

- Static text.

- Table.

- URL.

And, you can insert containers into the following element types:

- Container. (Containers can be nested.)

- Footer

- Header

- List view.

- Mobile grid.

- Select menu.

To define a container element you insert it into the layout using the Layout Grid on the Layout Designer – Layout page. After inserting a container you can specify container properties.

Related Links
Inserting Page Elements into the Layout Grid
Defining Container Page Elements
Defining Container Page Element Properties

Defining Container Page Elements

This topic describes how to:

- Add container page elements to pages.

- View defined container page elements in the Layout grid.

Understanding Defining Container Page Elements

To define a container element you use the Layout grid toolbar on the Layout grid on the Layout Designer – Layouts page to insert it into the layout page. At the time you define/add a container page element to your application, there are no other tasks to perform or parameters to specify. After you add the element to a layout page, use the Container Properties page to define styles and options for the page element.

Adding Container Page Elements to Pages

To add a container page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                              The Current Row button.

4. Click the Add Container button on the Layout grid toolbar.

                                                  The Add Container button.

After you click the button the element appears in the Layout grid.

Viewing Defined Container Page Elements in the Layout Grid

Container elements are indicated in the Layout grid by the *mapcont_start* and *mapcont_end* element IDs.

Image: Container element inserted into the Layout grid

This example illustrates a container element inserted into the Layout grid.

Within the start and end elements you insert additional page elements to build the container.

To define properties for a container element, click the Properties link to the right of the start ID.

Related Links

Inserting Page Elements into the Layout Grid
Understanding Managing Container Page Elements
Defining Container Page Element Properties

Defining Container Page Element Properties

This topic provides an overview of container page element properties.

Understanding Defining Container Page Element Properties

Many properties that you can define for container page elements are also available to define for other page elements. These common properties are described in the topic Common Layout Page Element Properties

This topic discusses using the properties that are unique to container page elements only.

Defining Collapsible Content

This topic describes how to:

• Define collapsible content areas.

• Define accordion content areas.

This section also provides an example of creating collapsible content areas and an example of creating accordion content areas.

Understanding Collapsible Content

Collapsible content can be hidden and revealed by tapping a title or button. When you create collapsible content, a header is created based on the value in the Label Text field for the *mapcont_start* field in the Layout grid on the Layout Designer – Layout page. The header acts as the control that users touch to expand and collapse the content area.

To create collapsible content areas and accordion content areas in the Mobile Application Platform, you create nest containers within a container.

The collapsible content area properties determine how content is initially displayed, collapsed, not collapsed, or as an accordion.

Defining Collapsible Content Areas

Use the Collapsible Content section on the Container Properties page to define collapsible content. To access the page select PeopleTools >Mobile Application Platform >Layout Designer.

Image: Collapsible Content section

This example illustrates the fields and controls in the Collapsible Content section of the Container Properties page. You can find definitions for the fields and controls later on this page.

Collapsible	Select the property to create collapsible content.
Initially Collapsed	This field is enabled by default when you select the Collapsible property and indicates that the content appears collapsed by default. Clear the field to display content areas expanded by default.
Collapse Set	Use this property to create an accordion. Select the property on the top-level element for the accordion. See the following section for more information about creating accordions.

Defining Accordions Content Areas

An accordion allows you to group components of collapsible content so that only one panel is visible at a time, and so that when you are viewing one panel (content area) the others are closed.

To define an accordion, in the Collapsible Content section in the Container Properties page, select the Collapse Set check box for the top-level container of the set that you want to appear collapsed by default. Note that the top-level container does not display in an accordion.

Example: Defining Collapsible Content

This section provides a simple example of defining a shell for collapsible content for a top-level container and five (5) nested level 2 containers.

This example shows the finished top-level container as it appears in generated HTML.

Image: Example: Generated HTML of this example showing collapsible content set to display initially collapsed

This example illustrates the generated HTML for the application and how the collapsible content initially appears in the application. The properties for the top-level container are set to display the container initially collapsed on the application page.
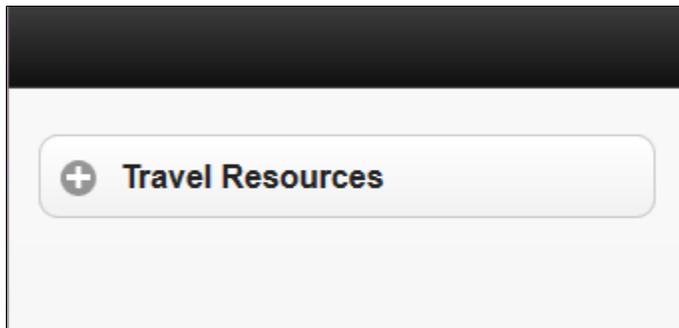
The following example shows the top-level container expanded to reveal additional containers/levels of information.

Image: Example: Generated HTML of collapsible content in this example with the top level expanded

This example shows generated HTML for the application when the user taps or selects the top-level container, revealing five (5) nested containers. As with the top-level container, in this example, the properties for the nested containers are set to display the content initially collapsed on the application page.
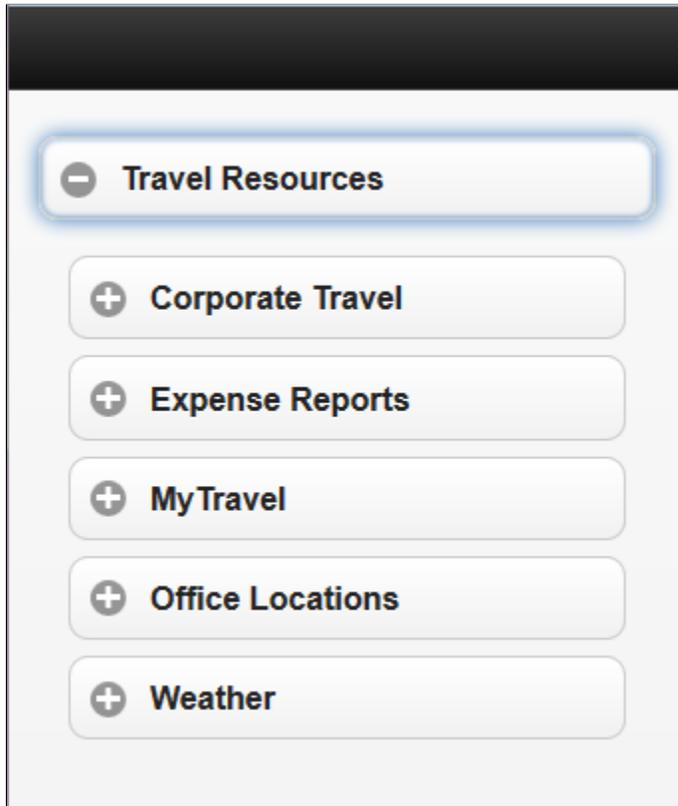
To build this example create a top-level container and then create five (5) containers inside the top-level container. The following example shows the Layout grid with the containers for the example created.

Image: Example: Containers created for the example

This example shows the container *Travel Resources* in the Layout grid. Defined within the *Travel Resources* container are five additional containers: *Corporate Travel, Expense Reports, MyTravel, Office Locations,* and *Weather.*

		ID	Label Text	Group		Class Name	Style	Theme	No Styles Override
		mapheader_1		Properties				▼	☐
		mapheader_end		Properties				▼	☐
➡		mapcont_start_1	Travel Resources	Properties				▼	☐
		mapcont_start_2	Corporate Travel	Properties				▼	☐
		mapcont_end_2		Properties				▼	☐
		mapcont_start_3	Expense Reports	Properties				▼	☐
		mapcont_end_3		Properties				▼	☐
		mapcont_start_4	MyTravel	Properties				▼	☐
		mapcont_end_4		Properties				▼	☐
		mapcont_start_5	Office Locations	Properties				▼	☐
		mapcont_end_5		Properties				▼	☐
		mapcont_start_6	Weather	Properties				▼	☐
		mapcont_end_6		Properties				▼	☐
		mapcont_end_1		Properties				▼	☐
		mapfooter_1		Properties				▼	☐
		mapfooter_end		Properties				▼	☐

Layout — Find | — 1-16 of 16

View Layout Grid Details

For each container define a value in the Label Text field that will become the label for the content header in the generated HTML. Notice that each container has two page element IDs, a *start* ID and an *end* ID. These IDs are similar to open and close tags in HTML and in practice you insert other page elements between the tags to build out the content for each container. So for example, in the *Corporate Travel* container you could insert an HTML Area page element and define HTML code, insert buttons that link to information on your company's web site concerning corporate travel, and much more. This example shows only how to create the shell containers for the collapsible content.

The Mobile Application Platform assigns all page elements system-generated IDs. You can use these IDs to reference the elements in any JavaScript you construct for your application. For any element ID that displays in hyperlink text, you can click the link to display the Alternate Label page and rename the ID. If you decide to rename any or all page element IDs it is your responsibility to ensure that they remain unique across your application.

The following example shows the element IDs renamed for this example. Adding alternate IDs is also a means for giving elements more descriptive and meaningful names, as well as making them easier to identify on the Layout grid.

Image: Alternate IDs created for the containers in the example

This example illustrates the page element IDs renamed for each of the container elements

Now that the page element IDs are renamed, it's easier to see how the *Travel Resources* container has five containers nested inside of it. The beginning of the *Travel Resources* container is indicated by the large blue arrow three rows from the top of the grid. The end of the *Travel Resources* container is three rows from the bottom of the grid. And within the *Travel_Resources_Container_Start* element ID and the *Travel_Resources_Container_End* element ID are the other containers that appear nested within the collapsible content.

The next step is to define properties for each container so that they will appear as collapsible content. But first, let's take a look at how the application renders given what has been developed so far.

(To preview a Mobile Application Platform application click the View Initialization URL link at the top of the Layout Designer — Layout page and copy the URL that displays into a browser.)

Image: Application preview with no collapsible content properties defined

The following example illustrates a preview of the application so far. Here you can see the top-level *Travel Resources* container, and under it are the five (5) nested containers. But there are no controls to expand or minimize content. After the container properties are set, the controls for expanding and minimizing the content containers will appear.
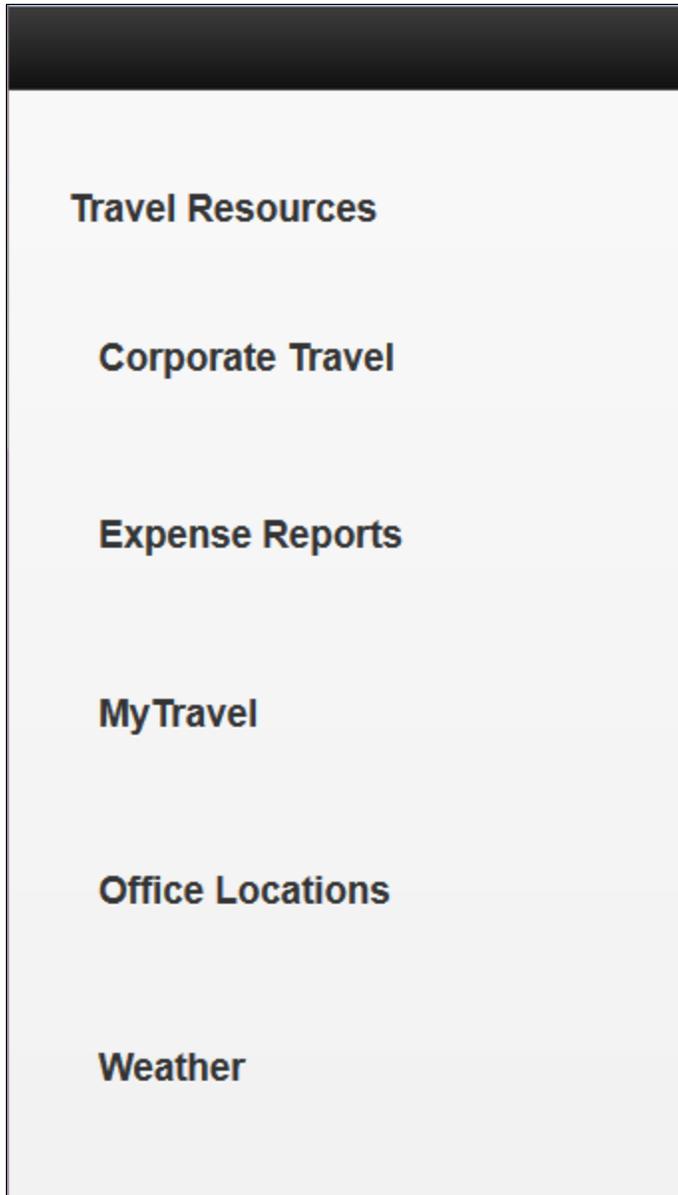
Note: The previous example originally opened full-screen in a browser window. In the image shown, the width has been manually adjusted/reduced to roughly simulate how the content might appear in a mobile device.

To set the properties for the containers in the application, in the Layout grid, click the Properties link next to a container name. The Container Properties page appears. The properties to set for this example are in the Collapsible Content section.

Image: Example: Collapsible content properties

This example illustrates the default view of the Collapsible Content section on the Container Properties page.

Collapsible Content ❓
☐ Collapsible
☐ Initially Collapsed
☐ Collapse Set

To enable collapsed content for a container, you must select the Collapsible control in the Collapsible Content section of the Container Properties page. When you select the option, by default the system also selects the Initially Collapsed option so that the content in the container appears collapsed by default.

If you try this example on your own, you'll notice that when you select the Collapsible box a message appears on the screen reminding you to make sure to enter a label in the Label Text field for the container on the Layout grid. As mentioned earlier, the system uses that label for the header label in the generated HTML, so it's important that you enter a value in the field.

Image: Example: Collapsible content properties set for the example

This example shows the collapsible content properties set for this example. The Collapsible control and the Initially Collapsed control are selected. Selecting the Collapsible property must be set to enable collapsible content to display. The Initially Collapsed property causes the content in the container to display collapsed by default.

Collapsible Content ❓
☑ Collapsible
☑ Initially Collapsed
☐ Collapse Set

These properties must be set as shown for *each* container in the example.

After the properties for each container are set and the application is saved, if you preview the application, the collapsed content should appear as shown in the examples at the beginning of this section.

Example: Defining Accordion Content

This section provides a simple example of defining a shell for an accordion. This example uses and builds off the containers used in the previous example.

The information in this section describes how to build the following accordion.

Image: Generated HTML of the accordion in this example

This example illustrates the generated HTML of the accordion content described in this example.
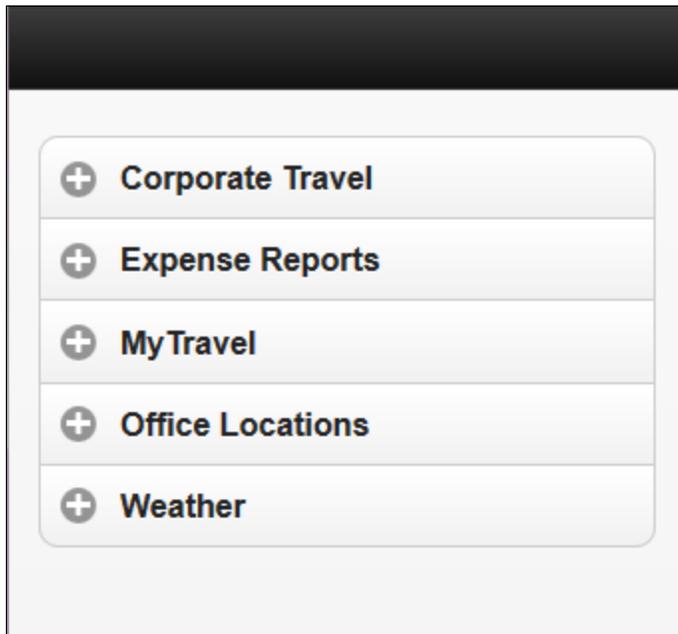
In this example, you use the same container structure as in the previous example for defining collapsible content. The only difference is in the properties you set for the containers.

Image: Example: Containers for the accordion example

This example illustrates the containers for the accordion collapsible content. These are the same containers and container structure used in the previous example.

To define accordion collapsible content, for the top-level container only, set the Collapse Set property in the Collapsible Content section on the Container Properties page.

Image: Example: Collapsible content properties set for the top-level accordion

This example illustrates the Collapse Set property selected in the Collapsible Content section on the Container Properties page. To define an accordion, set this property for the top-level container.

For all other containers defined for accordion collapsible content, at a minimum you must set the Collapsible property for each container. To display the content initially collapsed, as shown in the first example in this section, select the Initially Collapsed property for each container as well.

Image: Example: Collapsible content properties set for the second-level accordion containers

This example illustrates the collapsible content properties to set for each second-level container to define accordion collapsible content.

After the properties for each container are set and the application is saved, if you preview the application, the accordion collapsed content should appear as shown in the first example of this section.

Defining Pop-Up Content

When you define a container as a pop-up, the entire container acts as a pop-up with the content appearing on top of other content on the page. You typically invoke a pop-up container using a push button page element or a URL page element.

Use the Popup Options section on the Container Properties page to enable/define the container as a pop-up container, define where the pop-up appears on the page, whether to invoke the pop-up with a button or link, and so on.

To access the Container Properties page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid click the Properties link for the container you want to define as pop-up content.

Note: You cannot define a pop-up for a container that contains collapsible content.

Image: Popup Options section (Default view)

This example illustrates the fields and controls on the default view of the Popup Options section on the Container Properties page. You can find definitions for the fields and controls later on this page.
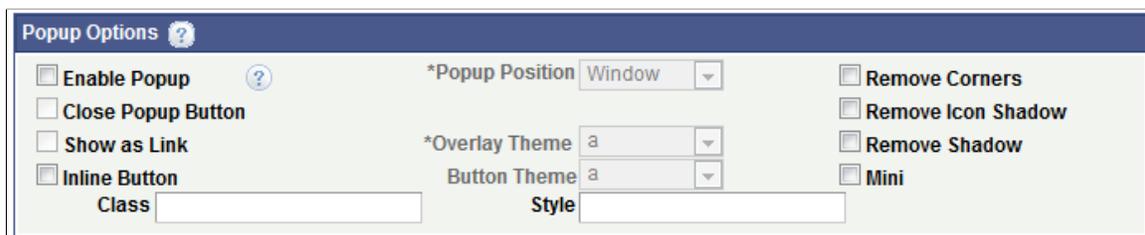
As you select properties in the Popup Options section of the Container Properties page, additional properties appear.

Image: Popup Options section (All properties)

This example illustrates the fields and controls on the Popup Options section. You can find definitions for the fields and controls later on this page.
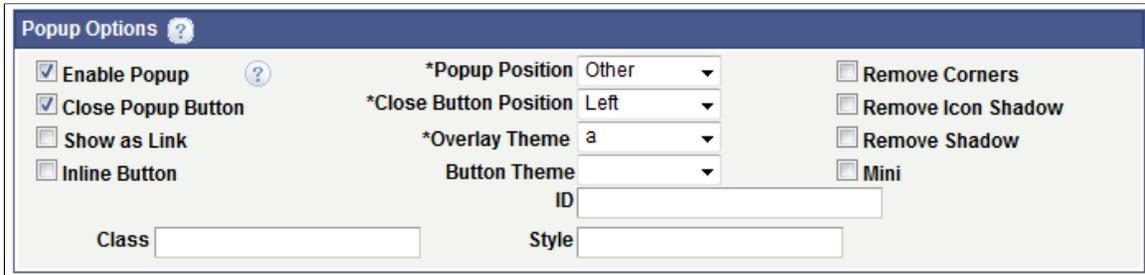
The following table describes the fields and controls that appear in the Popup Options section on the Container Properties page and are listed alphabetical order:

Button Theme	This drop-down list becomes enabled after you select the Enable Popup control.
	Choose a jQuery theme from the drop-down list to apply to the button.
	See Defining jQuery Mobile Theme Properties for more information about defining themes and the values that appear in the list.
Class	Enter the name of a custom CSS3 class to apply to the pop-up.
	The class must be defined in the Layout Designer – CSS page or in the layout template if a template is applied to the layout.
Close Button Position	This control appears after you select the Close Popup Button control.
	Select the position of the Close button. The values are:
	• *Left.*
	• *Right.*
Close Popup Button	Select the box to insert a close button for the pop-up content.
	After you select this control, the Close Button Position drop-down list appears for selecting the position of the close button in the pop-up.
Enable Popup	Select the control to enable a pop-up.
	After you select this control, the Popup Position, Overlay Theme, and Button Theme fields become enabled.
	In addition, the Image Properties section appears at the bottom of the Container Properties page for you to optionally define images for the pop-up button.

Inline Button	Select this box to use an inline button to invoke the pop-up content. Inline buttons do not use the whole width of the viewport.
Mini	Create a more compact version of the button element.
Overlay Theme	Choose a jQuery theme from the drop-down list to use as the background for the pop-up content. The theme covers the entire window behind the pop-up content.
	See Defining jQuery Mobile Theme Properties for more information about defining themes and the values that appear in the list.
Popup Position	By default, pop-up content opens centered vertically and horizontally over the page element/component/object selected (also known as the origin).
	The Mobile Application Platform framework also applies some basic collision detection rules to ensure that pop-up content appears on-screen, so the ultimate position may not always be centered over the origin.
	To change the default position of pop-up content, the options are:
	• *Window.* (Default.) Position centered vertically and horizontally over origin.
	• *Origin.* Position over the button or link used to invoke the pop-up content.
	• *Other.* Position over a defined page element. When you select this option, an ID field appears in the Popup Options section. Enter the element ID of the page element over which the pop-up content is to appear.
Remove Corners	By default buttons appear with rounded corners.
	Select this property to remove the rounded corners.
Remove Icon Shadow	By default icons appear with a shadow effect.
	Select this property to remove the shadow.
Remove Shadow	By default buttons and icons appear with a shadow effect. This option removes the shadow.
Show as Link	Select the option to invoke the pop-up with an image link. When you select this option the Image Properties section appears on the Container Properties page. You can define a static of dynamic link to invoke the pop-up container.
Style	Enter HTML5 styles for the pop-up container.

Use a semi-colon (;) to separate multiple style entries.

Example: Defining Pop-Up Containers

This example illustrates defining a pop-up container in the Mobile Application Platform framework.

The following example illustrates the button that will invoke the pop-up container.

Image: Example: Generated HTML of the button to invoke the contents of the container

This example illustrates the button control that invokes, or pops up, the contents of the container

While the example in this section illustrates using a push button page element to invoke the contents of the container, you can define a URL page element as a link or image to invoke the container contents.

In the example in this section, selecting the button will pop-up a container that contains some text and a group of buttons.

Image: Example: Generated HTML of pop-up container contents

This example illustrates the generated HTML of the pop-up container contents created in this example.
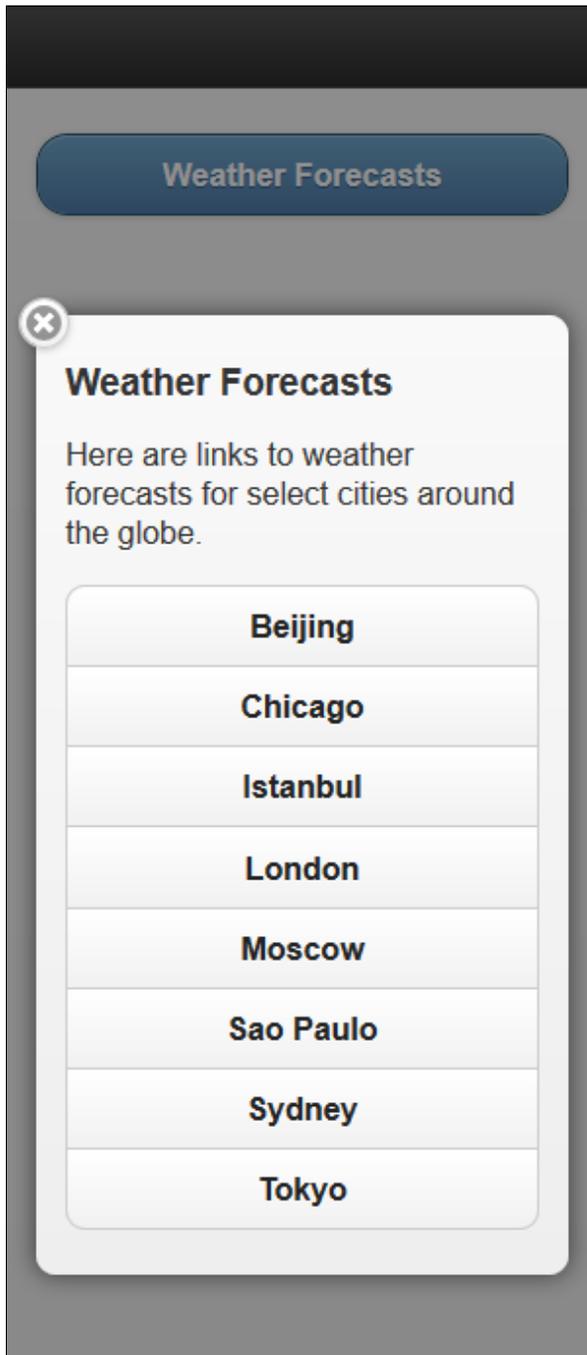
The previous example shows a title, some text, and eight (8) buttons. The title is from the container Label Text field as entered on the Layout grid on the Layout Designer – Layouts page. The display text was created by inserting an HTML Area page element into the container and entering the text in HTML tags. And the buttons were created by inserting a p button page element into the container and then creating a group of eight (8) buttons. After the page elements were defined, properties in the Popup Options section of the Container Properties page were set to enable the pop-up action and to define several display parameters.

Image: Example: Page elements added to the Layout grid for the example

This example illustrates the page elements added to the Layout grid for this example. The example shows a container inserted into the grid, and within the container an HTML Area element has been inserted, as well as a group of push button elements.

As noted in the example for creating collapsible content, some people may find that defining alternate IDs for page elements can help to more easily identify and manage page elements in the Layout grid. The following example shows the same layout, but with alternate IDs defined for the page element IDs.

Image: Example: Alternate IDs define for the page elements in the example

This example illustrates alternate page element IDs defined for the page elements in this example.

As a reminder, you can define an alternate page element ID for any page element that displays in the Layout grid with a hyperlinked ID value. To define the alternate ID, click the hyperlinked ID to access the Add Alternate Label page and define the alternate ID. Also remember that if you change system-defined page element IDs, you are responsible for ensuring that they remain unique within the application.

The steps to create the layout in the example are:

1. Insert a Container element into the Layout grid after the *mapheader_end* element (or end tag of the MAP header).

    In the Label Text field enter *Weather Forecasts.* This is the label that appears on the button that invokes the pop-up container contents, and that also appears at the top of the page when the container content opens.

2. Insert an HTML Area page element inside the *Weather Forecasts* container. After you insert the element, the HTML page appears where you enter the following HTML:

    ```
    <p>Here are links to weather forecasts for select cities around the globe.</p>⇒
    ```

    Note that after you define the HTML element you can easily and quickly modify the HTML by clicking the Properties link for the element in the Layout grid to access the HTML Area Properties page. The page features an HTML Area section where you can modify the HTML that displays on the page.

3. Define the button elements.

    a. Insert a push button element after the HTML Area element in the Layout grid.

When you insert the element, the "Do you want to create a group?" page appears as shown the following example:

Image: "Do you want to create a group?" page

The example shows the "Do you want to create a group?" page.
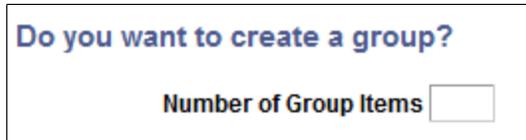
b. In the Number of Group Items field, enter *8* and click the Yes button.

The Select Button Type page appears for the first button.

Note the Group field at the above the Button Type section. It shows which button in the group with which you are working. For example, *Item 1 of 8*.

c. Click the URL radio button and click the *OK* button.

The Select Button URL page appears.

d. In the URL radio button.

A long edit box appears on the page.

e. In the URL field enter the URL to the target content. For this example you could enter a URL to a weather web site or any other site of your choosing.

The URL you enter must be fully-qualified and entered in the following format:

```
http://www.example.com
```

f. . Click the OK button.

The Select Button Type page appears and the Group field above the Button Type section displays the next item in the group. For example, *Item 2 of 8*.

g. Repeat *Step c* and *Step f* until all eight (8) items (URLs for the buttons) in the group are defined.

After you define the last item in the group and click the OK button.

The Layout Designer – Layout page appears.

4. Define the label to appear on each button at runtime.

Access the Layout grid and for each button enter the text you want to appear in the Label Text field next to the button element ID.

If you need to remind yourself of the URL you entered when you defined a button, click the Properties link in the Layout grid to access the Button Properties page. The URL Properties section on the page displays the URL. Notice how you can also click the Modify URL Mappings link to modify the URL if needed.

The next step is to define properties for the *Weather Forecasts* container so it appears as a pop-up container. But first, let's take a look at how the application renders given what has been developed so far.

(To preview a Mobile Application Platform application click the View Initialization URL link at the top of the Layout Designer — Layout page and copy the URL that displays into a browser.)

Image: Example: Application preview with no pop-up container properties defined

The following example illustrates a preview of the application so far. Here, the page title, *Weather Forecasts,* is from the Label Text field from the container page element ID on the Layout grid. The descriptive text is from the HTML Area element. And the buttons are from the push button element defined as a group of eight (8) items. But the content isn't appearing as an overlay. After the container properties are set, a button will display to invoke the content. And when the button is selected the content will pop up as an overlay.
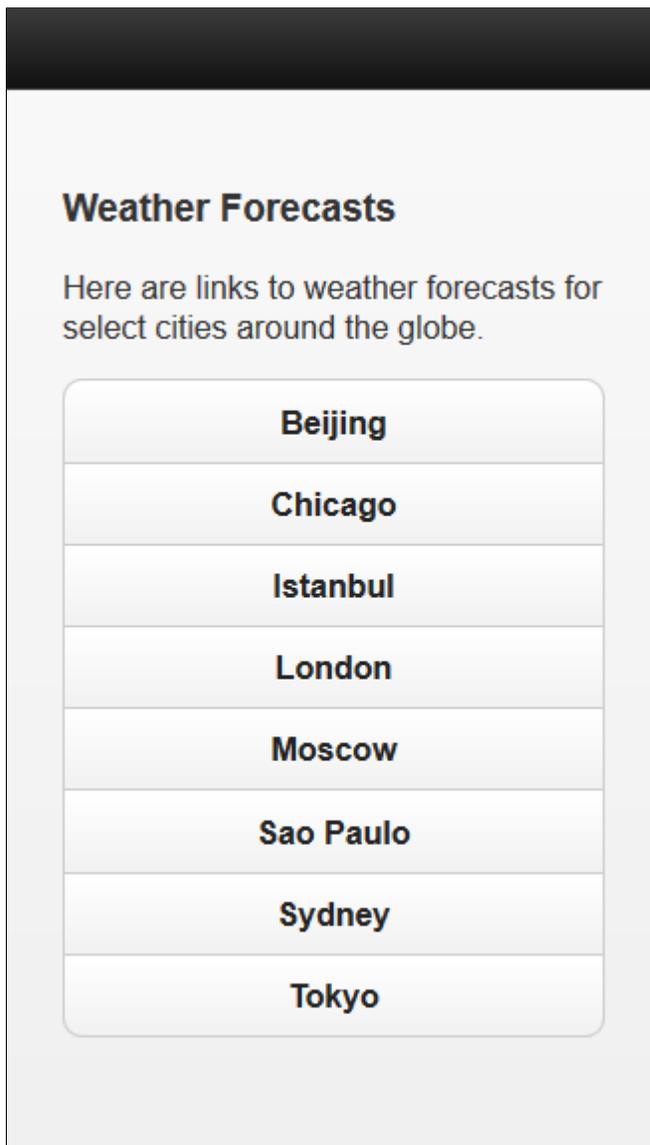
Note: The previous example originally opened full-screen in a browser window. In the example shown, the width was adjusted/reduced to roughly simulate how the content might appear in a mobile device.

To set the properties for the container, in the Layout grid, click the Properties link next the container's label text, *Weather Forecasts*. The Container Properties page appears. The properties to set for this example are in the Popup Options section.

Image: Example: Pop-up container properties set for the example

This example illustrations the properties in the Popup Options section of the Container Properties page set for the example. The Enable Popup control must be set for the content to appear as pop-up content.

After you set these properties for the container and save the layout, the content should appear as shown in the first two example in this section when you preview the application.

Hiding Container Labels in Generated HTML

By default any value you define in the Label Text field for a container in the Layout grid on the Layout Designer – Layouts page displays as a title for the container in the generated HTML.

The Container Properties page features a No Content Label control at the bottom of the page. Select this control to not display the label text defined on the Container. The label text can then be used simply as a reference during development to identify the of the container in the Layout grid.

Omitting the jQuery "Content" Data-Role in Container Division Elements

By default, containers that you create in the Mobile Application Platform contain the following element:

```
<div data-role="content">...</div>
```

The following attribute and value in the element are specific to jQuery Mobile:

```
data-role="content"
```

jQuery Mobile can insert styles and/or modify HTML for content contained within this element. This is most evident when working with nested containers.

You can omit the attribute and value from Mobile Application Platform containers by selecting the No Content property at the bottom of the Container Properties page for a container element.

If you select the No Content property the element becomes a basic HTML5 Division element, as shown in the following example:

```
<div>...</div>
```

Chapter 29

Managing Sub Container Page Elements

Defining Sub Container Page Elements

Sub Containers are defined Containers that are substituted in place of the defined element. Much like a sub record is defined with a record, sub containers allow for containers to be reused across the application or as a reference Container in another application. Moreover, the Sub Containers can be dynamically defined via People Code.

In order to add a Sub Container, a Sub Container must first be defined on a Sub Page.

The following edits pertain to sub containers:

- Similar to reference containers, sub containers can only be added to the appropriate area (header/footer/content section), based on where the sub container was created.

- If a sub page or container on the sub page is deleted then a warning popup message will be displayed if one of the page/container(s) is used on another page.

- The Sub Container can be defined more than once on a page, however they may have duplicate IDs, therefore a warning will be displayed.

- No Update or Save event(s) can be selected on Sub Page.

- Sub Page is not part of the URL Type *Layout Page* selection.

- A Sub Container cannot be defined on a Sub Page. You cannot nest Sub Containers.

Creating Sub Containers

To create sub containers:

1. Under the Layout Details section, click the Add Page button.

2. Click the page number above the document tree to access the page properties.

3. Select Sub Page radio button.

4. Click OK on the Warning message *changing Page Type may invalidate URL mappings*.

5. Click OK.

6. Insert containers with your desired elements on the page.

7. Save.

Inserting Sub Containers into the Layout

To insert sub containers into the layout:

1. Navigate to the page in the layout where you want to add the sub container.

2. Position the cursor where you want to add the sub container.

3. Click the Sub Container icon.

Sub Container icon

4. On the Sub Layout Container Selection page, select the page and layout container to use.

5. Save the MAP Layout.

Image: Sub Layout Container Selection page

This example illustrates the fields and controls on the Sub Layout Container Selection page. You can find definitions for the fields and controls later on this page.

Sub Page	Select the page from the drop down list. Only pages defined as sub pages are available.
Available Layout Containers	The sub page may contain multiple containers. Use the drop down list to select the container.

Dynamically Selecting Sub Container in PeopleCode

To dynamically select a different Sub Container in People Code simply use the *SubContainer* property on the MAP Element object and update with a different container ID which must (defined on same Sub Page.

This is an example of the code.

```
&Map.GetElement("mapsub_1").SubContainer = "mapcont_start_2";
```

Defining Sub Container Properties

To access the Sub Container Properties, click the Properties link for the Sub Container.

Image: Sub Container Properties page

This example illustrates the fields and controls on the Sub Container Properties page.
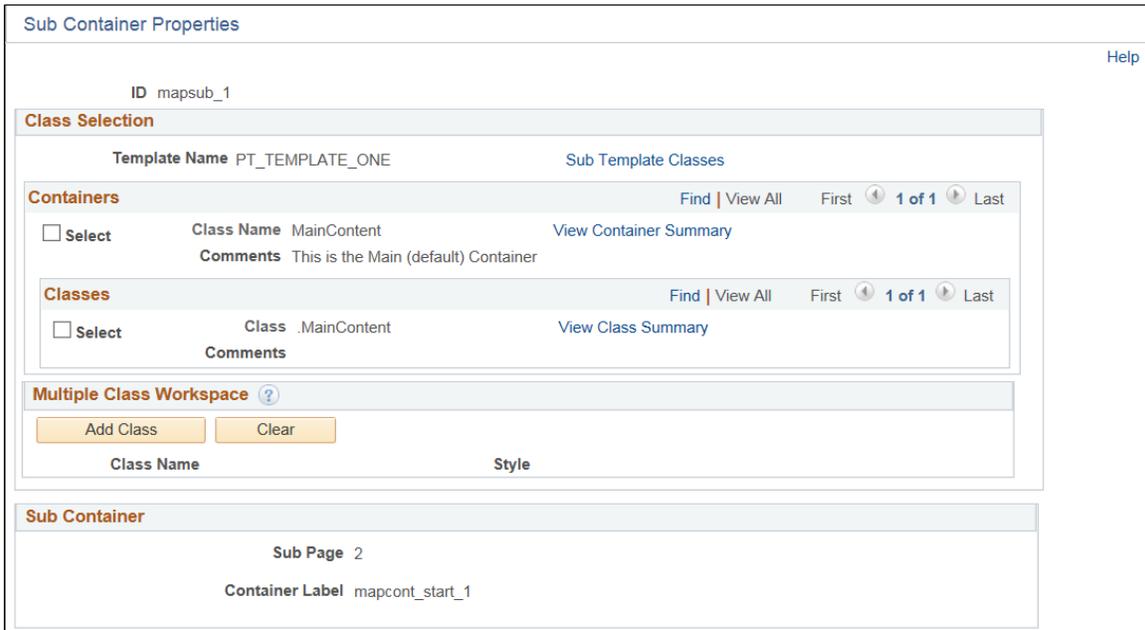
See <u>Applying Template Containers and Selector Style Classes</u> for information on the Class Selection properties.

Chapter 30

Managing Mobile Grid Page Elements

Managing Mobile Grid Page Elements

This topic describes how to manage mobile grid page elements.

Understanding Mobile Grid Page Elements

Mobile grids let you position field types next to one another in rows and columns.

Mobile grids are useful for smaller element types like buttons and icons. They are not recommended for large items or chunks of copy, since mobile device resolution is much lower than a desktop and usability is extremely diminished.

To define a mobile grid insert a mobile grid element into the Layout grid on the Layout grid on Layout Designer – Layout page. After you insert the element you can define a number of properties for it in the Mobile Grid Properties page, including the number of rows and columns in the grid.

You can insert the following Mobile Application Platform page elements into a mobile grid element:

• Action sheet.

• Button.

• Check box.

• Container.

• Flip toggle.

• HTML area.

• Image/static text.

• Input.

• Radio button.

• Select box.

• Slider.

• Static text.

• URL.

Note the following about using container elements in mobile grid elements:

- If you insert a container element inside a mobile grid element, you can use only the following elements in the container element:

    - Button.

    - Check box.

    - Flip toggle.

    - HTML area.

    - Image/static text.

    - Input.

    - Radio button.

    - Select box.

    - Select menu.

    - Slider.

    - Static text.

    - URL.

- You cannot nest container elements inside a mobile grid element.

Defining Mobile Grids Page Elements

To define a mobile grid element you use the Layout grid toolbar on the Layout grid on the Layout Designer – Layouts page to insert it into the layout page. At the time you define/add a mobile grid page element to your application, there are no other tasks to perform or parameters to specify. After you add the element to a layout page, use the Mobile Grid Properties page to define styles and options for the page element.

Adding Mobile Grid Page Elements to Pages

To add a mobile grid page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

   The Current Row button.

4. Click the Add Mobile Grid button on the Layout grid toolbar.

                                                  The Add Mobile Grid button.

Viewing Defined Mobile Grid Page Elements in the Layout Grid

Mobile grid elements are indicated in the Layout grid by the *mapmobl_start* and *mapmobl_end* element IDs.

Image: Mobile grid element inserted into the Layout grid

This example illustrates a mobile grid element inserted into the Layout grid.

The label text for sidebar elements does not appear on the user interface at runtime, however you may find it useful to enter values in these fields for reference during development.

To define properties for a mobile grid element, click the Properties link to the right of the start ID. Defining properties for mobile grid page elements is described in the next section.

Defining Mobile Grid Page Element Properties

This section describes how to define properties for mobile grid page elements.

Understanding Defining Mobile Grid Page Element Properties

Many properties that you can define for mobile grid page elements are also available to define for other page elements. These common properties are described in the topic Common Layout Page Element Properties

This topic discusses using the properties that are unique to mobile grid page elements only.

Defining Properties for Mobile Grid Page Elements

Use the Mobile Grid Properties section of the Mobile Grid Properties page to define properties for this element.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid click the Properties link next to the *mapmobile_start* element ID.

Image: Mobile Grid Properties section

This example illustrates the fields and controls in the Mobile Grid Properties section of the Mobile Grid Properties page. You can find definitions for the fields and controls later on this page.
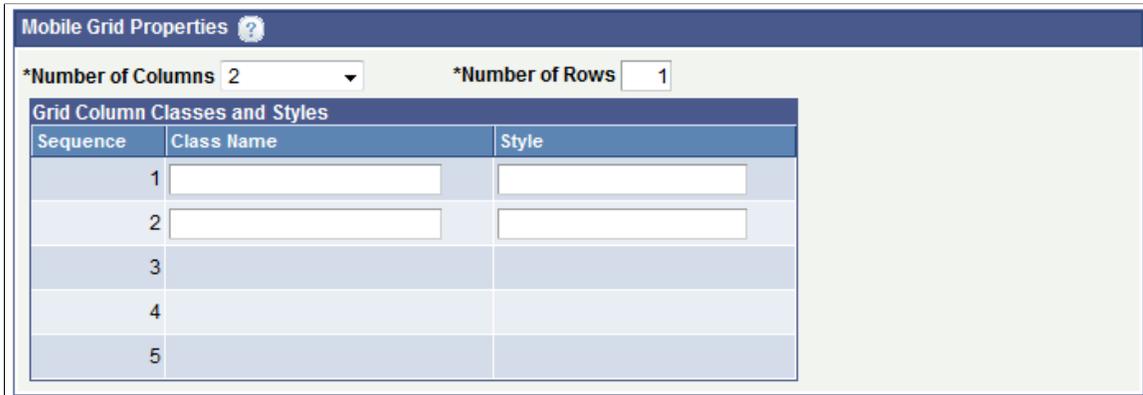
The Mobile Grid Properties section is where you define the number of columns and rows in the mobile grid and where you can apply CSS3 classes and HTML5 styles to each row in the grid.

By default, two columns and one row are defined in a mobile grid. A minimum of two columns must defined and a maximum of five columns is allowed. A minimum of one row must be defined and there is no maximum for rows.

The number or rows that appear in the Grid Column Classes and Styles grid is equal to the number of columns that you define for the grid.

After you define the columns, rows, and any optional styling, save the changes to return to the Layout grid to insert elements into the mobile grid.

Inserting Page Elements into Mobile Grids

When you insert page elements into a mobile grid page element in the Layout grid, you insert elements for each column in row one, followed by elements for each column in row two, and so on.

To illustrate how to insert rows and columns in a mobile grid and how they appear in the HTML consider the following grid:

Column 1	*Column 2*	*Column 3*	*Column 4*
Row 1/Column 1	Row 1/Column 2	Row 1/Column 3	Row 1/column 4
Row 2/Column 1	Row 2/Column 1	Row 2/Column 1	Row 2/Column 1
Row 3/ Column 1	Row 3/Column 1	Row 3/Column 1	Row 3/Column 1

Omitting the column headings, the example shows a grid with three (3) rows and four (4) columns.

The following example shows a mobile grid element populated with static text elements to create the grid in the previous example:

Image: Mobile grid element populated with static text elements for the example

This example illustrates a partial view of the Layout grid and shows a mobile grid page element inserted into the grid and populated with static text elements for the example.

In the example, Mobile Grid Example appears in the Label Text field for the *mapmobl_start_1* element ID. The label text serves as a reference only and does not appear in the generated HTML.

A total of 12 static text page elements are inserted between the *mapmobl_start_1* and *mapmobl_start_2* element IDs and are inserted in the following order:

1. Static text page elements entered sequentially for all the static text entries in row 1.

2. Static text page elements entered sequentially for all the static text entries in row 2.

3. Static text page elements entered sequentially for all the static text entries in row 3.

The text entered in the Label Text field for each static text element is the text that appears in the grid in the generated HTML.

The following example shows the generated HTML for the example:

Image: Generated HTML of the mobile grid example

This example illustrates the generated HTML of the mobile grid example.

Chapter 31

Managing Static Text Page Elements

Managing Static Text Page Elements

See the following topic for information on how to manage static text page elements:

Task	*Topic*
Define static text page element.	See <u>Defining Static Text and Image Page Elements</u>
Define static text page element properties.	See <u>Defining Static Text Properties</u>

Chapter 32

Managing Image/Static Text Page Elements

Defining Image/Static Text Page Elements

See the following topics for information on how to manage image/static text page elements:

Task	*Topic*
Define image/static text elements.	See <u>Defining Static Text and Image Page Elements</u>
Define image/static text page element properties.	See <u>Defining Image Properties</u>

Chapter 33

Managing List View Page Elements

Defining List View Page Elements

This topic provides an overview of list view page elements.

Understanding Dynamic and Static List View Page Elements

List view elements in the Mobile Application Platform can be dynamic or static.

Dynamic List View   A *dynamic list view* is one where the number of rows is determined at runtime based on the number of items in a PeopleSoft document collection element.

Note that a compound element wrapped in a collection element must be defined in the PeopleSoft document associated with the layout to define a dynamic list view page element.

Static List View   A *static list view* is where one row is displayed.

Adding List View Page Elements

To add a list view page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

    Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

    The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

        The Current Row button.

4. Click the Add Listview button on the Layout grid toolbar.

        The Add Listview button.

After you click the button the Listview page appears for you to define the element.

Viewing the Listview Page

The Listview Page appears after you add a list view page element to the layout grid as described in the previous section.

Important! To create a list view with rows of data, the PeopleSoft document linked to the layout must contain a collection element wrapped in a compound element. It is the collection element that provides *n* number of rows in the list view. You must also select the Dynamic Flag box on the Listview page to create a multi-row list view.

Image: Listview page (Default view)

This example illustrates the fields and controls on the default view of the Listview page. You can find definitions for the fields and controls later on this page.

Image: Listview page with field added to the Listview Composition grid

This example illustrates the fields and controls on the Listview page when you add a field to the Listview Composition grid. You can find definitions for the fields and controls later on this page.

Defining General List View Parameters

This table describes the options you can define at the top of the Listview page:

Ordered	Select the radio button to create a numbered list.
Unordered	Select the radio button to create an unnumbered list. (Default.)
Add URL	Select the box to add a URL to the row which will be fired off when selected.
	You will define the URL after you define the values on the Listview page.
Add Split Button	Select the box to create two different interactive actions using URLs on the same row.
	You will define the split button after you define the values on the Listview page.
JavaScript	This control appears if you add a field to the Listview Composition grid and define the field as any Type for which you can define a primitive element.
Image	Add an Image for the row from the image repository.

	You will select an image after you define the values on the Listview page.
Icon	Add an for the row from the image repository.
	You will select an icon after you define the values on the Listview page.
Dynamic Flag	Select the box to create a dynamic list view that will use a collection element to populate the list view composition arguments.

Defining List View Composition Parameters

This table describes the fields and controls in the List View Composition grid. See the previous table for a description of fields and values to define:

Add Field	Click the link to add an additional column to the list view. You can add up to 50 additional columns.
	The field appears at the bottom of the Listview Composition grid.
	See Defining Additional List View Fields later on this page for more information.
Title	Defines the text to show on the list view row.
	Define the source of the title text:
	Important! If the Dynamic Flag is selected, in this field you must define a primitive wrapped in a collection, as that determines the number of rows to add for the list view.
Description	Defines the text to show under the title on the list view row.
	Define the source of the description text.
Count bubble	Generates a circle with a number inside, rendered on the right side of the row. This should ideally be an integer value (primitive of type integer).
Aside	Insert a second column in a row for supplemental text. If a count bubble is used, an aside appears to the left of the count bubble.
Attributes	Add jQuery attributes to a row or rows.
Divider	Create a visual separator between rows. The divider can go above or below the current row.
Dynamic Image	Add a dynamic image to a row or rows.
Dynamic URL	Enter a value to dynamically add a URL to the list view.

A dynamic URL will override a URL defined using the Add URL control at the top of the page.

Alternate URL   Use an alternate URL when you want to dynamically change the element's mapped URL due to a form factor change. You can map a primitive which then can be populated with a Java Script function.

In PeopleCode the form factor can be identified and if applicable can set a property on the element (URLAlternativeURL).If the property is set to *True*, at runtime the JavaScript function will be called instead of the defined mapped URL.

Note: You can use an alternate URL on all URL types except the *Update* type.

Accessibility   Added alternate text to meet accessibility requirements.

Field#   This field appears after you click the Add Field link above the Listview Composition grid.

See Defining Additional List View Fields later on this page for more information.

Using Common Elements to Define List View Composition Parameters

The following table lists common values used to define parameters in the Listview Composition grid:

Type   Select the value type and source from the list.

Depending on the List View Item Type, the options are:

- *Label Text.* When you select this option, the system uses the text defined on the Layout Designer page in the Label Text field on the Layout grid as the label for the list view.

- *Msg Cat.* Select this option to use a message catalog entry. When you select this option, Msg Set, Msg Num, and Message Text fields appear in the Listview Composition grid. After you enter the appropriate values for the message set and message number, the message text appears in the grid.

- *Primitive.* Select this option to use a primitive element. When you select this option a Primitive link appears in the Listview Composition grid. Click the link to access the Select Primitive page to select a primitive element. After you select an element, it displays in the Primitive field in the Listview Composition grid.

Msg Set	This field appears when you select *Msg Cat* as a list view item type. Enter the message set number to use for the type.
Msg Num	This field appears when you select *Msg Cat* as a list view item type. Enter the message number to use for the type.
Message Text	After you define a message set and message number, the message text appears in this field.
Select Primitive	Click the link to access the Select Primitive page to select a primitive value.
Primitive	If you select a primitive value for the list view item type, the name of the primitive element displays in this field.
Class	Enter a CSS3 class name for list view item. The class must be defined in the Layout Designer – CSS page or in the layout template if a template is applied to the layout.
Style	Enter HTML5 styles for the list view item. Use a semi-colon (;) to separate multiple style entries.
Hidden	This control appears only when you add a field to the grid. Select the control to hide a column. This control is read-only for all item types except for the Field item type.
Divider	Use this control to insert a visual separator in the list view. The values are: • *Above.* Insert a divider above the row. • *Below.* Insert a divider below the row.

Defining Additional List View Fields

This topic discusses defining additional list view fields.

Understanding Adding List View Fields

Use the Add Field link above the Listview Composition grid on the Listview page to add additional columns to a list view.

Note the following points about adding additional columns to a list view page element:

• A PeopleSoft document element type must be selected for each row added.

• To remove any column, leave the Type field blank. The system will remove the column during save.

- To hide a column, select the Hidden box in the Listview Composition grid. The Hidden box appears in the grid only when you add an additional field to the grid.

Defining Element Types for Additional List View Fields

After you add an additional field to the Listview Composition grid, you must select an element type for the field and define the type. The following table lists the element types you can define for an additional field and the associated type.

Warning! If a dynamic URL is populated on the Listview Composition grid and during run time this primitive is actually populated with a value, then any fields defined as a Checkbox, Radio Button, Button, HTML or Input page element will not be displayed at runtime.

Element Type	Type	Comments
Button.	*Not Applicable*	None.
Check box.	*Primitive*	When you select the check box element type, the Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive element.
HTML.	*Primitive.*	When you select the check box element type, the Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive element.
Image.	*Primitive.*	When you select the check box element type, the Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive element.
Input.	*Primitive.*	When you select the check box element type, the Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive element.
Radio Btn. (Radio button.)	*Primitive.*	When you select the check box element type, the Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive element.

Element Type	Type	Comments
Text. (Default.)	• *Label Text.* • *Msg. Cat.* • *Primitive.*	See the Common Elements used to Define List View Composition Parameters earlier in this topic for more information about the type options.

You can also use the Listview properties page to add fields other than *Text* or *Image* under the following conditions:

• The Add URL box is not selected.

• At least one field element type is not defined as *Text* or *Image.*

Defining Primary URLs and Split Button URLs for List View Page Elements

If you select the Add URL box or the Add Split Button box on the Listview page, the Define Primary URL page appears when you click the OK button. And if you selected the Add Split Button box on the Listview page, the Select Split Button URL page appears after you've defined the primary URL.

Image: Select Primary URL page

This example illustrates the fields and controls on the Select Primary URL page.

Image: Select Split Button URL page

This example illustrates the fields and controls on the Select Split Button URL page.

You define the following URL types:

URL Type	Description	Comments
(External) *URL*	A URL to an external web site or resource.	NA
Call REST Service	Define a URL to a REST service.	NA
Call Layout	Define a URL to another Mobile Application Platform layout.	NA
Layout Page	Define a URL to a page in the current layout.	NA
Update	Define a URL to a PeopleCode Update event.	This URL type is not available to define for split buttons and does not appear on the Select Split Button URL page.

The fields and controls that you define depend on the URL type selected.

The documentation for Defining URL Page Elements applies to defining primary URLs and split button URLs for list view page elements. Please see that documentation for more information.

Defining Images for List View Page Elements

Use the Select Image page to select an image for the list view.

This page appears when you define an image for the list view definition. The page appears after you've defined the fields and controls on the Listview page.

Image: Select Image page

This example illustrates the fields and controls on the Select Image page. You can find definitions for the fields and controls later on this page.

Add Image	Select the box to display the Image Selector link.
Image Selector	Click the link to access the Image Selector page and to select an image.
	After you select an image, click the OK button to return to the Select Image page.
Image Name	Displays the name of the image selected.

Defining Icons for List View Page Elements

Use the Select Icon page to select an image for the list view.

This page appears when you define an icon for the list view definition and after you've defined the fields and controls on the Listview page.

The Mobile Application Platform framework delivers a number of icons from which to choose, however you can easily implement your own icons.

To implement your own icons create CSS on a layout template or on the Layout Designer — CSS page and enter the CSS class name in the Custom field on the Select Icon page. The framework will determine that this value is not one of the default values, and will apply the custom CSS class that will consist of a value for the data icon prefixed with *ui-icon-*. For example, if you add the value *happy* in the Custom field, then the value actually rendered will be *ui-icon-happy*. Icons should be 18 x 18 pixels square.

Image: Select Icon page

This example illustrates the fields and controls on the Select Icon page. You can find definitions for the fields and controls later on this page.

Icon	From the drop-down list select an icon for a defined split button. The values are:

- *Alert.*

- *Back.*

- *Bars.*

- *Check.*

- *Delete.*

- *Down Arrow.*

- *Edit.*

- *Forward.*

- *Gear.*

- *Grid.*

- *Home.*

- *Info.*

- *Left Arrow.*

- *Minus.*

- *Plus.*

- *Refresh.*

- *Right Arrow.*

- *Search.*

- *Star.*

- *Up Arrow.*

Custom   In lieu of selecting one of the delivered icons, enter a CSS class
                                       name in the field to use as the icon.

Location   Choose the placement of the icon in the list view row. The
                                       values are:

- *Bottom.*

- *Left.*

- *No text.*

- *Right.*

- *Top.*

Defining JavaScript for List View Field Elements

Use the Layout URL Creation Page to execute JavaScript for the following list view field elements:

- *Button.*

- *Check box.*

- *HTML.*

- *Input.*

- *Radio button.*

When adding a field to the Listview Composition grid, if you select any of these element types, a JavaScript box appears toward the top of the Listview page. If you check the box, after you finish defining field on the page and click the OK button, the Layout URL Creation Page appears.

Image: Layout URL Creation Page

This example illustrates the fields and controls on the Layout URL Creation Page. You can find definitions for the fields and controls later on this page.

In the JavaScript long edit box, enter the JavaScript to execute when the URL link is selected.

Viewing List View Page Elements in the Layout Grid

After you define a list view page element the Layout Designer – Layout page appears.

List view page elements are indicated in the Layout grid by the element ID *maplistview*

Image: List view page element inserted into the Layout grid

This example illustrates a list view page element inserted into the Layout grid.

The system generates default text that appears in the Label Text field, *LISTVIEW_TEXT.* This is the label that appears for the list view at runtime, so you should enter a more meaningful and appropriate label.

To define properties for a list view element click the Properties link to the right of the element ID.

Related Links
Defining List View Page Element Properties

Defining List View Page Element Properties

This topic describes how to define list view page element properties.

Understanding Defining List View Properties

This topic describes how to define list view page element properties, including defining properties on the Listview Properties page and on the Modify Listview Mappings page.

Many properties that appear on the Listview Properties page also appear on the properties pages for other page elements. These common properties are described in the topic Common Layout Page Element Properties . This topic discusses using the properties on the Listview Properties page that are unique to working with list view page elements only.

Related Links

Accessing Page Element Properties

Using Listview Section Properties

Use the Listview Properties page to set properties for list view page elements. To access the Listview Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the list view element for which you want to define properties.

The Listview Properties page features a Listview Properties section.

Image: Listview Properties section of the Listview Properties page

This example illustrates the fields and controls on the Listview Properties section of the Listview Properties page. You can find definitions for the fields and controls later on this page.

The following table lists the fields and controls that appear in the Listview Properties section of the Listview Properties page. The fields and controls that are available to select and edit depend on the initial definition of the list view.

The fields and controls are listed in alphabetical order.

Type	The values are:
	• *Unordered.* Create an unnumbered list or items.
	• *Ordered.* Create a numbered list of items.
Description	When this read-only field is selected a description is defined for the list view.
Inset	By default the lists are in full-page mode. Select this control to make inline lists.

Aside	When this read-only field is selected an aside is defined for the list view.
Count Bubble	When this read-only field is selected a count bubble is defined for the list view.
Count Theme	From the drop-down list select a jQuery theme for a defined count bubble.
Filter	Select this option to add a filter to the list view to filter names of list items.
Filter Theme	From the drop-down list select a jQuery Mobile theme for a defined filter.
Split Button	When this read-only field is selected a split button is defined for the list view.
Split Theme	From the drop-down list select a jQuery Mobile theme for a defined split button.
Split Icon	From the drop-down list select an icon for a defined split button. The values are:

- *Alert.*

- *Back.*

- *Bars.*

- *Check.*

- *Delete.*

- *Down Arrow.*

- *Edit.*

- *Forward.*

- *Gear.*

- *Grid.*

- *Home.*

- *Info.*

- *Left Arrow.*

- *Minus.*

- *Plus.*

- *Refresh.*

- *Right Arrow.*

- *Search.*

- *Star.*

- *Up Arrow.*

Place Holder	Select the box to define a short hint that describes the expected value of an input field.
Collapsible Options Section	The Collapsible Options subsection features these options: • Collapsible. Select the option to make the list view collapsible. • Initially Collapsible. Select the option to show the list view collapsed upon initialization of the page.
Placeholder Text Section	If you select the Place Holder field, the fields in this section become enabled. Enter information in the following fields to select placeholder text from the message catalog: • Msg Set. Enter the message set. • Msg Num. Enter the message set number. The Message field displays the message text.
Image Type **Icon**	From the drop-down list select an icon to appear. See the description for Split Button for a list of available icons. The Right Arrow icon is the default.
Theme Icon	From the drop-down list select a jQuery Mobile theme for a defined icon.
Modify Listview Mappings	Click the link to access the Modify Listview Mappings page to add or modify fields and containers for the list view. See Using the Modify Listview Mapping Page for more information.
Dynamic Flag	When this read-only field is selected the page element is defined as a dynamic list view.
List View Composition grid	This section shows the Listview Composition parameters defined. Click the Modify Listview Mappings link to access the Modify Listview Mappings page to add, modify, or delete parameters.

Related Links
Defining jQuery Mobile Theme Properties

Using the Modify Listview Mappings Page

Use the Modify Listview Field Mappings page to add, modify or remove items from a list view.

To access the page, click the Modify Listview Mappings link in the Listview Properties section of the Listview Properties page.

Image: Modify Listview Field Mappings page (Default view)

This example illustrates the fields and controls on the Modify Listview Field Mappings page.

Image: Modify Listview Field Mappings page (Additional fields and container added)

This example illustrates the fields and controls on the Modify Listview Field Mappings page. The Additional Fields grid shows that several additional fields and a container have been added.

Most of the fields and controls that appear on the page are described in the Defining List View Page Elements topic.

The following table describes fields and controls that are not described in that topic:

Add Field	Click the link to add additional fields to the list view.
	When you click the link an Additional Fields grid appears at the bottom of the page.
Add Container	Click the Add Container link to add a container to the list view.
	When you click the link the container appears in the Additional Fields grid appears at the bottom of the page, and features start and end container tags.
Alt ID	Add an alternate attribute for selection.
	Alternate IDs are populated in the HTML as *altid*.
Move Item Up	Click the link to move the selected item up in the grid.
Move Item Down	Click the link to move the selected item down in the grid.
Delete Item	Click the link to delete the selected item in the grid.
Layout Position	Indicates if a row is the current row or a non-current row.
	The following icons indicate position. Click an icon to toggle between making the row the current and non-current row.

Indicates the current row in the Additional Fields grid.

Indicates the non-current row in the Additional Fields grid.

Chapter 34

Managing Select Menu Page Elements

Defining Select Menu Page Elements

This topic discusses how to define select menu page elements.

Adding Select Menu Page Elements to Pages

To add a select menu page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

    Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

    The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

    The Current Row button.

4. Click the Add Select Menu button on the Layout grid toolbar.

    The Add Select Menu button.

After you insert a select menu element into the Layout grid the "Do you want to create a group?" page appears.

Image: "Do you want to create a group?" page

This example illustrates the "Do you want to create a group?" page.
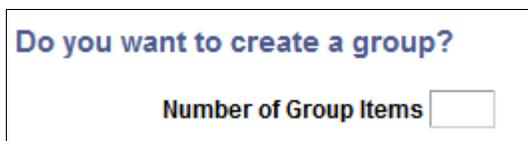
If you want to create a group of select menus, enter the number in the Number of Group Items field and click the Yes button. If you do not want to create a group of select menus click the No button.

After you define whether to create an element group, the Select Primitive for Select Menu page appears for you to define the select menu.

Related Links

Using the Select Primitive for Select Menu Page

Image: Select Primitive for Select Menu page

This example illustrates the fields and controls on the Select Primitive for Select Menu page. You can find definitions for the fields and controls later on this page.
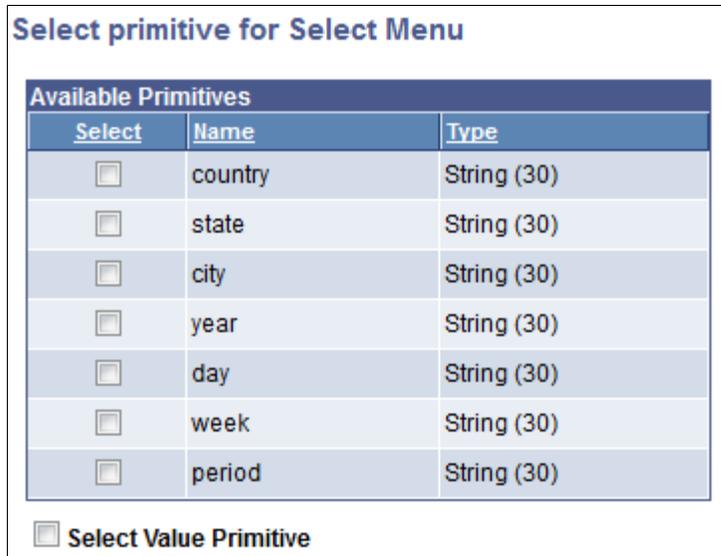
The grid displays primitive elements defined in the PeopleSoft document defined for the layout.

Select	Select the primitive for the select menu from the list.
	Typically the element to select is a primitive wrapped in a collection in order to display more than one menu item.
Select Value Primitive	Select the control to select a different primitive for the value. This value is passed back to the application server and subsequently read in PeopleCode.
	If you select this control the Select Primitive for Value page appears for you to select the primitive to use for the value.

Using the Select Primitive for Value Page

If you select the Select Value Primitive control on the Select Primitive for Select Menu page, this page appears.

Image: Select Primitive for Value page

This example illustrates the fields and controls on the Select Primitive for Value page. You can find definitions for the fields and controls later on this page.

In the grid, select the primitive to use for the select menu value.

Viewing Defined Select Menu Elements in the Layout Grid

After you define a select menu element by selecting a primitive value for it, the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for select menu page elements is the name of the primitive you defined for it., for example *<primitive_name>_n.*

Image: Select menu defined in the Layout grid

The example illustrates a select menu defined in the Layout grid.

In the previous example, *state_1* is the element ID.

By default the system populates the Label Text field with the name of the primitive defined for the element, *state* in this example. This is the same name defined in the Element Name field for the primitive in the PeopleSoft document defined for the layout. This value appears as the label for the select menu. You can keep the default value, enter a different value, or remove it.

Click the Properties link to access the Select Menu Properties page to define properties for the element.

Defining Select Menu Page Element Properties

This topic discusses how to define select menu page element properties.

Understanding Defining Select Menu Page Elements

Many properties that you can define for select menu page elements are also available to define for other page elements. These common properties are described in the topic Common Layout Page Element Properties

This topic discusses using the properties that are unique to select menu page elements only.

Defining Select Menu Properties

This topic describes using the Select Menu Properties section of the Select Menu Properties page.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid click the Properties link next to the element ID of the select menu element for which you want to define properties.

Image: Select Menu Properties section (Default view)

This example illustrates the fields and controls in the default view of the Select Menu Properties section on the Select Menu Properties page. You can find definitions for the fields and controls later on this page.

If you select the Non-native box addition properties appear in the section as shown in the following example.

Image: Select Menu Properties section (Non-Native Properties displaying)

When you select the Non-native property in the Select Menu Properties section, additional properties appear as shown in this example.
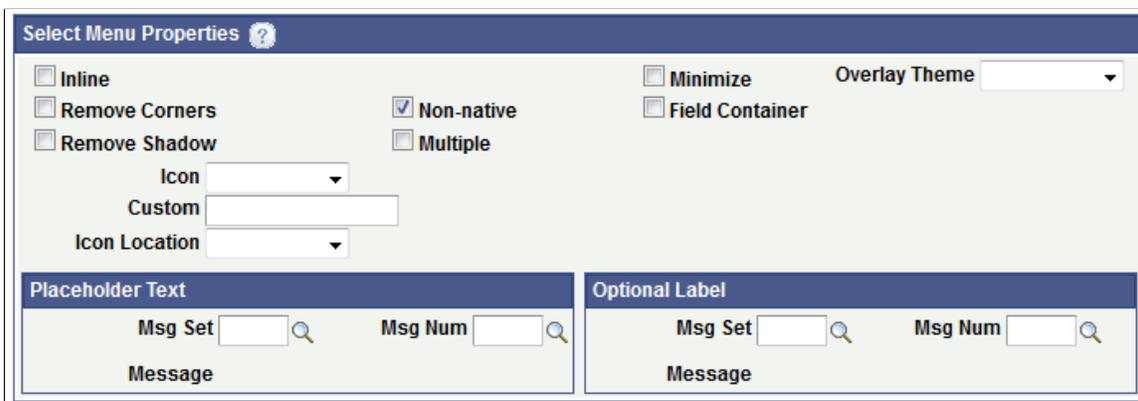
The following table describes the fields and controls in the Select Menu Properties section of the Select Menu Properties page. The fields and controls are listed in alphabetical order:

Custom   Enter the name of a custom CSS3 class for a custom icon.

	The class must be defined in the Layout Designer – CSS page or in the layout template if a template is applied to the layout.
Field Container	Select this option to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.
Icon	From the drop-down list select an icon for the select menu button. The values are:

- *Alert.*

- *Back.*

- *Bars.*

- *Check.*

- *Delete.*

- *Down Arrow.*

- *Edit.*

- *Forward.*

- *Gear.*

- *Grid.*

- *Home.*

- *Info.*

- *Left Arrow.*

- *Minus.*

- *Plus.*

- *Refresh.*

- *Right Arrow.* (Default.)

- *Search.*

- *Star.*

- *Up Arrow.*

Icon Location	From the drop-down list select the location for the icon. The values are:

- *Bottom.*

- *Left.*

- *No Text.*

- *Right.*

- *Top.*

Inline	Select the property so that the length of the button label determines the button width.
Minimize	Select the property to create a more compact version of the element.
Multiple	Select the property to allow users to select multiple options from a select menu.
Non-native	Select the property to provide an alternative to the native select element of the mobile browser. When you select this property the following additional fields and controls appear:

- Multiple.

- Overlay Theme.

- Optional Label.

- Placeholder Text.

Optional Label controls	Use these controls to define a group label for the menu options, The label appears just below the menu title.

- *Msg Set.* Select the message set of the message catalog entry that contains the label to use.

- *Msg Num.* Select the message number of the message catalog entry that contains the label to use.

- *Message.* Displays the text of the message catalog entry for the message set and message number defined.

Overlay Theme	From the drop-down list select a jQuery theme for the overlay layer of the select menu.
	For more information about jQuery themes in the Mobile Application Platform see <u>Defining jQuery Mobile Theme Properties</u>
Placeholder Text controls	Use these controls to define a non–selectable overlay title.
	When defined, placeholder text displays as the title of the menu instead of the first menu item.

- *Msg Set.* Select the message set of the message catalog entry that contains the placeholder text to use.

- *Msg Num.* Select the message number of the message catalog entry that contains the placeholder text to use.

- *Message.* Displays the text of the message catalog entry for the message set and message number defined.

Remove Corners   Select the property to remove the border radius around the button.

Remove Shadow   Select the property to remove the shadow around the button.

Chapter 35

Managing URL Page Elements

Defining URL Page Elements

This topic discusses how to define URL page elements.

Understanding Defining URL Page Elements in the Mobile Application Platform

In the Mobile Application Platform you can define URL page elements to link to:

- Standard URLs.

- REST services.

- Other Mobile Application Platform layouts.

- Pages within a Mobile Application Platform layout.

- Execute PeopleCode Update events.

You can create standard URL hyperlinks or define hot spot images.

This topic discusses how to define each of these URL types. When you are done defining a URL page element, you can define additional properties using the URL Properties page.

Related Links
Defining URL Properties

Adding URL Page Elements to Pages

To add a URL page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                     The Current Row button.

4. Click the Add URL button on the Layout grid toolbar.

                                          The Add URL button.

After you click the button the URL page appears to define the element.

Using the URL Page

Image: URL page (Default view)

This example illustrates the fields and controls on the default view of the URL page. You can find definitions for the fields and controls later on this page.

The fields and controls that you use to define a URL depend on the URL Type that you choose in the default view of the page.

Common Elements Used to Define URL Page Elements

URL   Click the radio button to define a standard URL, for example:

`http://www.example.com`

After you click the button, fields and controls specific to defining a standard URL appear.

Call REST Service   Click the radio button to define a URL to REST service.

After you click the button, fields and controls specific to defining an external URL appear.

Call Layout   Click the radio button to define a URL to another Mobile Application Platform layout.

After you click the button, fields and controls specific to defining an external URL appear.

Layout Page	Click the radio button to define a URL to a layout page in the current layout.
	After you click the button, fields and controls specific to defining an external URL appear.
Update	Click the radio button to define a URL that executes a PeopleCode Update event.
	After you click the button, fields and controls specific to defining an external URL appear.
Add Image	Select the control to add image that becomes the hot spot for the URL.
Image Selector	This link appears if you select the Add Image control.
	Click the link to access the Image Selector page to select an image.
Image Name	After you select an image using the Image Selector page, the image name appears in this field.
Select Primitive	Click the link to access the Select Primitive page to select an image defined as a primitive element to display dynamically.

Related Links

Using the Image Selector Page
Using the Select Primitive Page

Defining Standard URLs

When defining a standard URL you can specify a fully-qualified URL or an anchor using an id attribute (#<id>).

Note: For cases were an icon is desired within a list item an anchor must be present. In this case simply use a "#" which will render <href = #> which will be then ignored as a link.

To define a standard URL click the URL radio button on the URL page.

Image: URL page (Standard URL)

This example illustrates the fields and controls on the URL page when you are defining a standard URL. You can find definitions for the fields and controls later on this page.
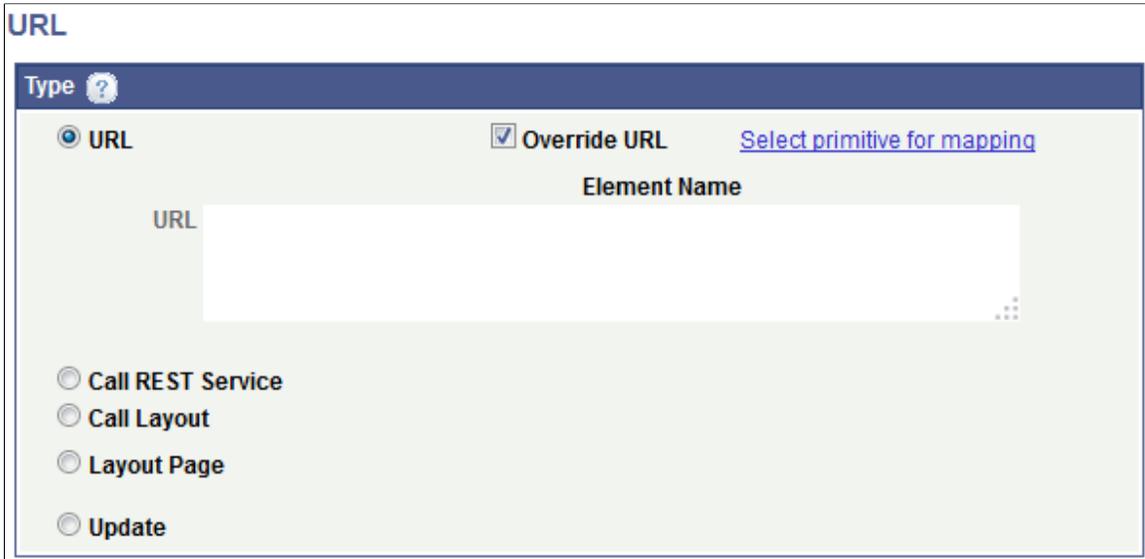
When defining a standard URL, you can define a static URL or define a dynamic URL. To define a static URL, enter a value in the URL long edit box. To define a dynamic URL, check the Override URL box and then click the Select primitive for mapping link to select a primitive value for the URL.

URL	Enter the URL in the long edit box.
Override URL	Select the box to define a dynamic URL using a primitive element defined in the PeopleSoft document associated with the layout.
	You cannot override a URL for a Listview page element, however when working with a Listview element type an option is available to add JavaScript.
Select primitive for mapping	This link appears if you select the Override URL box.
	Click the link to access the Select Primitive page to select a primitive.
Element Name	If you define a primitive for the URL, the primitive element name appears in this field.

Related Links
Using the Select Primitive Page

Defining URLs to REST Services

You can define a URL to call provider or consumer REST services. When defining this URL type, you select the REST service operation, select the appropriate URI template, and then identify the variables to pass.

To define a URL that calls a REST service click the Call REST Service radio button on the URL page

Image: URL page (Call REST Service)

This example illustrates the fields and controls on the URL page when you are defining a URL that calls a REST service.

Click the Select Rest Operation link to access the URI Selection page to build the REST URI.

See Using the URI Selection Page

After you build the URI an Operation field appears on the page that displays the name of the REST service operation.

Defining URLs to Other Layouts

When you define a URL to call another layout you must select the name of the layout to call, define the appropriate layout initialization to use, and identify the variables to pass.

To define a URL to call another layout click the Call Layout radio button on the URL page.

Image: URL page (Call Layout)

This example illustrates the fields and controls on the URL page when you are defining a URL to another Mobile Application Platform layout. You can find definitions for the fields and controls later on this page.
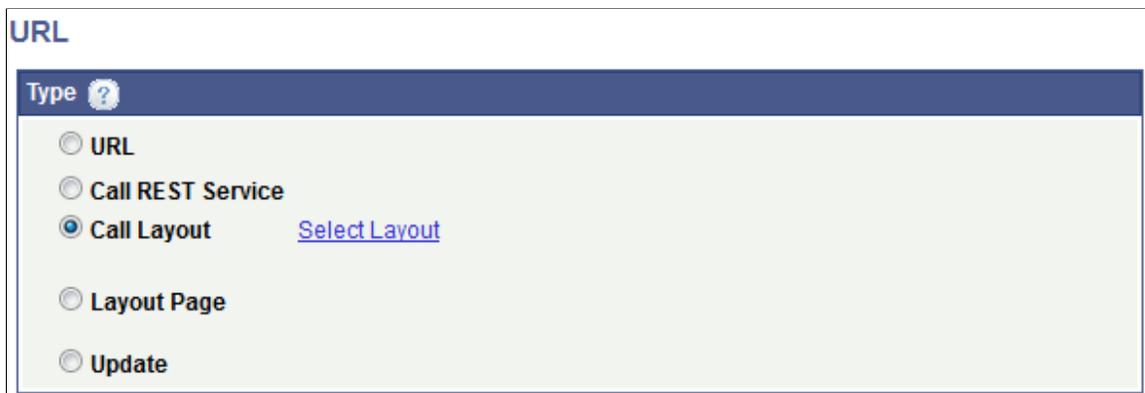
When you click the Call Layout radio button a Select Layout link appears. Click the link to access the Layout Selection page to select the layout to call.

See Using the Layout Selection Page

Defining URLs to Pages Within Layouts

Click the Layout Page radio button on the URL page to define a URL to call a page within the same layout.

The fields and controls that appear on the page depend on if you are defining a URL to a page defined as an Internal type page or an External type page. (You define the Internal and External page types using the Page Details page.)

Image: URL page (Internal Layout Page)

This example illustrates the fields and controls on the URL page when you are defining a URL that calls an Internal type layout page in the same layout. You can find definitions for the fields and controls later on this page.

Image: URL page (External Layout Page)

This example illustrates the fields and controls on the URL page when you are defining a URL that calls an External type page in the same layout. You can find definitions for the fields and controls later on this page.
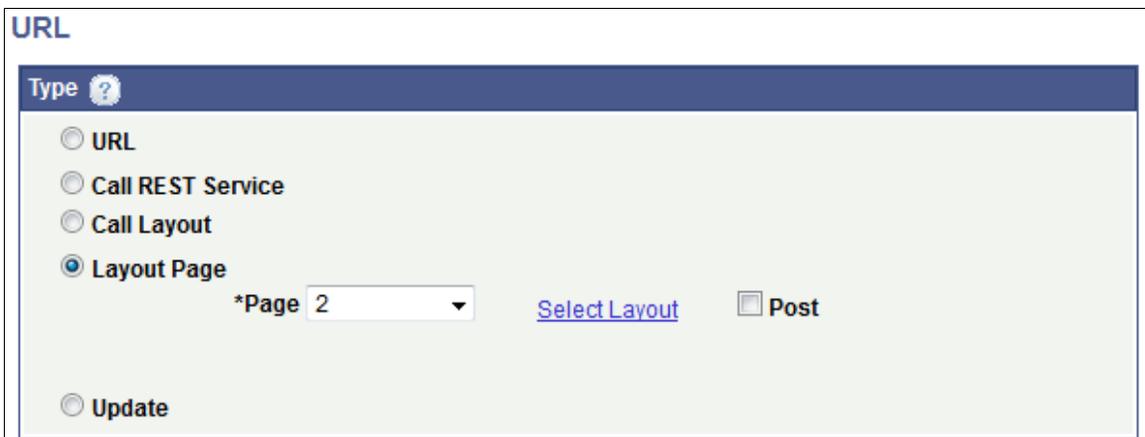
Page	From the drop-down list select the number of the layout page to call.
Select Layout	This link appears when you select an External type page from the Page drop-down list.
	Click the link to access the Layout Selection page to select the layout to call.

See Using the Layout Selection Page

Post   By default the external page is called via an HTTP GET. If data to be passed as arguments needs to be more secure, select the box to make the request via an HTTP POST.

Related Links
Using the Page Details Page
Defining Page and Layout Navigation

Defining URLs to Execute PeopleCode Update Events

Click the Update radio button on the URL page to define a PeopleCode Update event to call.

Image: URL page (Call PeopleCode Update event)

This example illustrates the fields and controls on the URL page when you are defining a URL to call an PeopleCode Update event. You can find definitions for the fields and controls later on this page.

Map to URI   Click the link to access theSelect URI page to select the URI parameters to pass to the Update event.

The Select URI page displays the URIs defined as initialization parameters defined for the layout.

See Using the Select URI Page

URI   After you define the URI parameters to pass using the Select URI page, the URI string displays in this field.

Package   Enter the name of the application package that contains the method to invoke.

Path	Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.
Class ID	Enter the name of the application class that contains the method to invoke.
Method	From the drop-down list select the method to invoke.

Using the Layout Selection Page

Use the Layout Selection page to:

- Select a layout to call when you are defining a URL to call another layout.

- Select a layout page to call when you are defining a URL to call another page within the same layout.

When you are working with either of these URL types, a Select Layout link appears on the URL page. Click the Select Layout link to access the page. The default view that appears depends on the URL type that you are defining.

Image: Layout Selection page (Default View for Call Layout URL)

This example illustrates the fields and controls on the default view of the Layout Selection page when you are defining a URL to another layout. You can find definitions for the fields and controls later on this page.
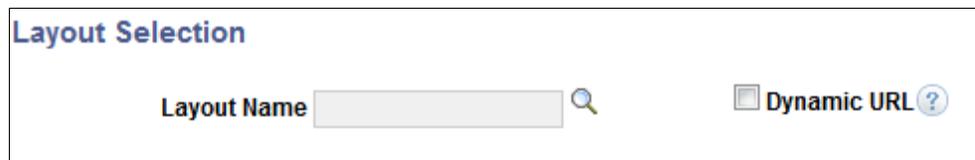
After you define a layout to call, the Layout URI List grid appears and is populated with the URIs defined as initialization parameters for the layout.

When you are defining a URL to call a page in the same layout, the Layout Selection page appears with the Layout URI List grid populated, as shown in the following example.

Image: Layout Selection page (Layout URI List grid populated)

This example illustrates the fields and controls on the Layout Selection page when the Layout URI List grid is populated. This is the default view of the page when you are defining a Layout Page URL. You can find definitions for the fields and controls later on this page.

For both Call Layout and Layout Page URL types, you must select a URI in the Layout URI List grid. If you select the default initialization URI, there are no additional configuration steps. When you select a URI other than the default initialization URI, the URI Parameters grid appears on the page, as shown in the following example.

Image: Layout Selection page (Layout URI List grid and URI Parameters grid populated)

This example illustrates the fields and controls on the Layout Selection page when the Layout URI List and URI Parameters grids are populated. You can find definitions for the fields and controls later on this page.

In the URI Parameters grid, define the parameter values for the URI.

You can define constant or primitive values for the URI parameters. When you select the Constant control, a Const Value field appears where you enter the constant value. When you select the Primitive control, a Select Primitive link appears to access the Select Primitive page to define a primitive for the parameter.

Image: Layout Selection page (Defining URI Parameters)

This example illustrates the fields and controls on the Layout Selection page when defining URI parameters in the URI Parameters grid. You can find definitions for the fields and controls later on this page.
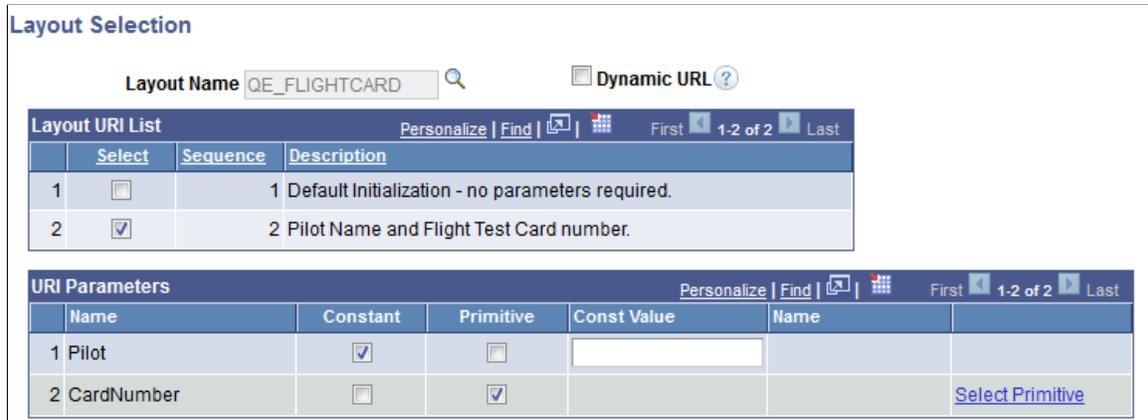
Layout Name	When defining a URL to another layout, from the drop-down list select the name of the layout to which to link. If you are defining a URL to a page within the same layout, this field displays the name of the current layout.
Dynamic URL	Select the box for the system to determine the complete URL when the user clicks the link instead of at page initialization. When you select this link, the framework treat the link as an absolute external link, therefore will hide the automated Back button and disable any links created on the Layout Designer – Layout page on the layout or layout page being called.
Select	Select the box to select the corresponding URI in the REST Operations grid.
Sequence	Indicates the sequence for the selected URI, as defined in the in the service operation named in theOperation field.
Description	Displays the description for the URI as defined in the initialization parameters for the layout defined in the Layout Name field.
URI Parameter **Name**	Displays the name of the URI parameter.
Constant	Select the control to define a constant value for the parameter. When you select the control, a Const Value field appears where you can enter the constant value.
Primitive	Select the control to define a primitive value for the parameter. When you select the control, a Select Primitive link appears. Click the link to access the Select Primitive page to select a primitive value.
Const Value	This field appears when you select the Constant box.

	Enter a constant value for the URI parameter.
(Primitive) **Name**	If you define a primitive value for a parameter, the name of the primitive appears in the field.
Select Primitive	This link appears when you select the Primitive control in the URI Parameters grid. Click the link to access the Select Primitive page to define a primitive value for the parameter.

Related Links

Using the Select Primitive Page

Viewing Defined URL Page Elements in the Layout Grid

URL page elements are indicated in the Layout grid by the *mapurl_n* element ID.

Image: URL element defined in the Layout grid

This example illustrates a URL page element defined in the Layout on the Layout Designer — Layouts page.

The system generates default text that appears in the Label Text field, *URL_LINK_TEXT*. This is the label text for the hyperlink at runtime, so you should enter a more meaningful and appropriate label.

If you have defined an image hot spot, leave the Label Text field blank unless you want a label to display for the image hot spot at runtime.

To define properties for a sidebar element click the Properties link to the right of the start element ID. Defining properties for sidebar elements is described in the next section.

Defining URL Page Element Properties

After you define a URL page element, there are a number of properties you can define to manipulate the characteristics of the URL.

See Accessing Page Element Properties and Common Layout Page Element Properties

Managing HTML Page Elements

Managing HTML Page Elements

This topic describes how to manage HTML page elements.

Defining HTML Page Elements

This topic discusses how to:

- Add HTML area page elements to pages.

- Use the HTML page to add HTML to pages.

- View defined HTML page elements in the layout grid.

Adding HTML Page Elements to Pages

To add an HTML area page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

        The Current Row button.

4. Click the Add HTML Area button on the Layout grid toolbar.

        The Add HTML Area button.

After you click the button the HTML Page appears where you enter code in HTML5 format that you want to appear on the layout page.

Using the HTML Page to Add HTML to Pages

Image: HTML page

This example illustrates the HTML page.
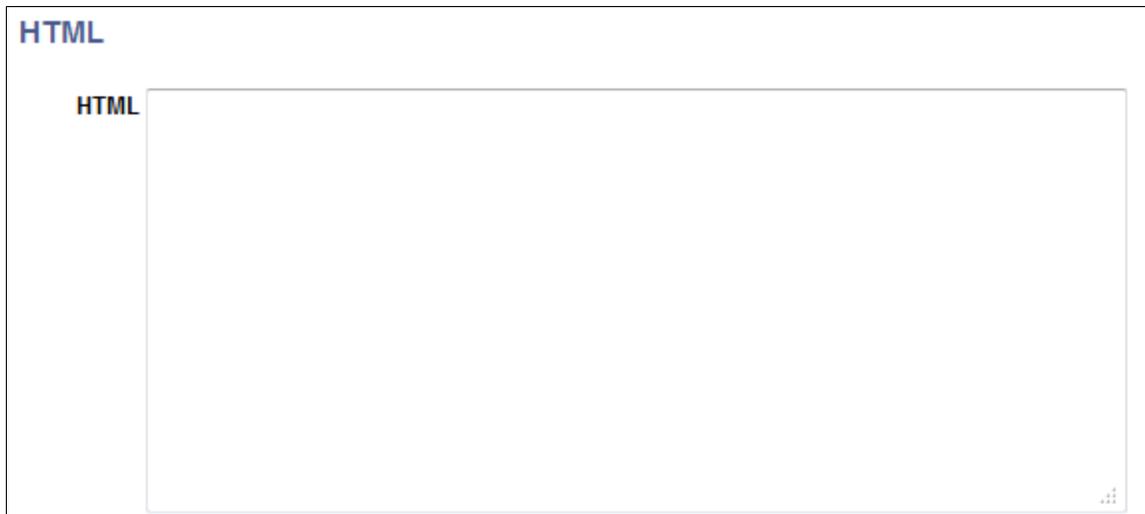
In the HTML long edit box, enter the HTML that you want to appear on the layout page.

Note that after you've defined an HTML page element you can define properties for it using the HTML Area Properties page.

Viewing Defined HTML Area Page Elements in the Layout Grid

HTML Area page elements are indicated in the Layout grid by the *maphtml* element ID.

Image: HTML Area page element defined in the Layout grid

This example illustrates an HTML Area element defined in the Layout grid.

The label text for HTML area elements does not appear on the user interface at runtime, however you may find it useful to enter a value in the field for reference during development.

To define properties for an HTML element click the Properties link. Defining properties for HTML area elements is described in the next section.

To apply custom styles to the element, use the Class, Style, Theme and No Style Overrides fields and controls to apply custom styles to the element. See Defining Styles and Themes for Page Element Labels

Defining HTML Page Element Properties

This section describes defining HTML page element properties.

Understanding Defining HTML Page Element Properties

To define properties for HTML elements use the HTML Properties page.

Many properties that appear on the HTML Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the HTML Properties page that are unique to working with HTML areas. These properties are located in the HTML Area section of the page.

Defining HTML Element Properties

To access the HTML Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the HTML element for which you want to define properties.

Image: HTML Area section

This example illustrates the fields and controls in the HTML Area section of the HTML Properties page. You can find definitions for the fields and controls later on this page.

HTML	The long edit box displays the HTML current defined for the element.
	Use the field to modify or update HTML for the element.
Change Mapping	Click the link to access the Change HTML Mapping page to change the HTML from HTML code to a primitive element or vice versa.
	See the next section for information about using the Change HTML Mapping page.
	Click the icon to view the HTML in a modal window.

Using the Change HTML Mapping Page

Image: Change HTML Mapping page (Default view)

This example illustrates the fields and controls on the default view of the Change HTML Mapping page, when HTML is selected as the HTML source. You can find definitions for the fields and controls later on this page.

Image: Change HTML Mapping page (Primitive)

This example illustrates the fields and controls on the Change HTML Mapping page, when the mapping source is a primitive element a PeopleSoft document. You can find definitions for the fields and controls later on this page.
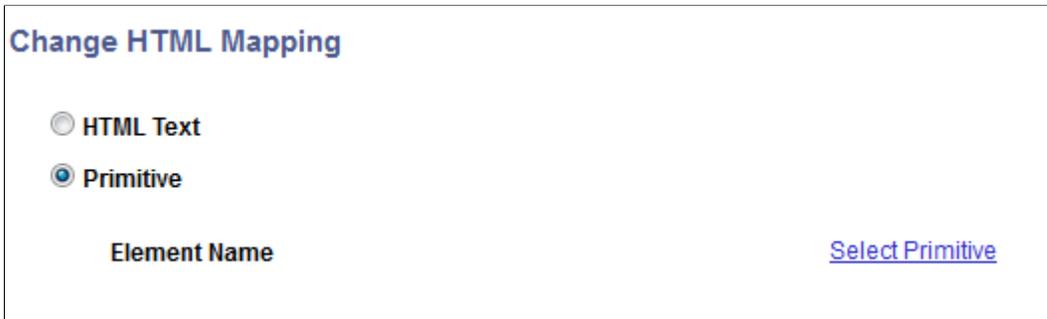
HTML Text	Select the radio button to change the HTML source to HTML code.
HTML	Enter HTML in HTML5 format that you want to appear on the layout page.
Primitive	Select the radio button to change the HTML source to a primitive defined in the PeopleSoft document associated with the layout.
Select Primitive	Click the link to access the Select Primitive page to select a primitive element for the HTML source.
Element Name	This field displays the name of the primitive element defined as the HTML source if applicable.

Related Links

Attaching Files in MAP Application

You can attach files from local drive (device folders) or from any configured cloud Provider repository, such as GoogleDrive, Oracle Document Cloud and so forth. On the MAP layout, the file attachment will be defined in HTML page element.

To attach files in MAP application:

1. Insert an HTML element in the MAP layout.

2. In Application Designer, create an HTML definition that includes an iframe to the required PIA page which has the attachment functionality.

    The attachment takes place in the PIA Page using File Attachment APIs.

    This is an example of the html area.

    ```
    html>
    <body>
    <h1></h1>
    <p></p>

    <iframe id="mapAttachPIAPage" type="text/html" src="ABC/EMPLOYEE/QEDMO/c/QE_FP⇒
    .QE_TESTUTIL.GBL" scrolling=auto style="width: 800px; height: 594px;"></iframe⇒
    >
    </iframe>
    </body>
    </html>
    ```

3. Use PeopleCode application class to dynamically load the HTML in MAP Layout.

    This is an example of the OnInitEvent for the MAP layout, which will load the HTML page, you will need to add code to retrieve the web server information to update the HTTP URL.

    ```
    method OnInitEvent
       /+ &Map as Map +/
       /+ Returns Map +/

       Local Compound &CUSTCOM, &COM, &URICOM, &COM2, &COM1;
       Local Document &Document, &CustDocument, &URIDocument, &DOC2, &DOC1;
       Local MapPage &Page;
       Local MapElement &Element;
       Local MapElement &ElementHeader, &ElementFooter;
       Local boolean &bRet;
       Local integer &i = 1;
       Local Primitive &one;
       Local Message &MSG, &RETMSG;

       &Document = &Map.GetDocument();
       &COM = &Document.DocumentElement;

       &URIDocument = &Map.GetURIDocument();
       &URICOM = &URIDocument.DocumentElement;

       Local string &str = GetHTMLText(HTML.QE_MAP_ATTACH);

       /* Update the HTML for the current env from above string*/
       /* create the code to retrieve web server and update &newstr with the HTTP ⇒
    ```

```
URL */
    Local string &newstr = Substitute(&str, "ABC", &server);

    &CustDocument = &Map.GetCustDocument();
    &CUSTCOM = &CustDocument.DocumentElement;
    &CUSTCOM.GetPropertyByName("prim01").value = &newstr;

    Return &Map;
end-method;
```

4. View the attached file or any uploaded file in MAP using MAP Ajax event and getAttachmentURL functionality.

   This is an example of PeopleCode with getAttachmentURL:

```
method OnAjaxEvent
    /+ &Map as Map +/
    /+ Returns Map +/

    Local Collection &DEMO_COLL, &COLL_ONE;
    Local Compound &DEMO_COMP, &COM, &URICOM, &CUSTCOM;
    Local Document &CustDocument, &Document, &URIDocument;
    Local MapPage &Page, &CustPage;
    Local MapElement &Element, &CustElement;
    Local MapElement &ElementHeader, &ElementFooter;
    Local Rowset &RS_EMPLOYEE;
    Local boolean &bRet;
    Local integer &i = 1;
    Local Primitive &one;

    &Document = &Map.GetCustDocument();
    &COM = &Document.DocumentElement;

    Local integer &RETCODE;
    Local string &URL_ID;
    Local string &ATTACHSYSFILENAME;
    Local string &ATTACHUSERFILE;
    Local string &ATTACHMENTURL;
    Local string &AUTHTOKEN;

    &URL_ID = "record://PSFILE_ATTDET";

    /*
    &ATTACHSYSFILENAME = &COM.GetPropertyByName("QE_PO_REF").Value;
    &ATTACHUSERFILE = &COM.GetPropertyByName("QE_PO_REF").Value;
    */

    &ATTACHSYSFILENAME = &Map.GetAJAXValueByName("mykey");
    &ATTACHUSERFILE = &Map.GetAJAXValueByName("mykey");


    &AUTHTOKEN = &Map.GetAuthToken();
    &RETCODE = GetAttachmentURL(&URL_ID, &ATTACHSYSFILENAME, &ATTACHUSERFILE, &⇒
ATTACHMENTURL, &AUTHTOKEN);

    &COM.GetPropertyByName("prim02").Value = &ATTACHMENTURL;

    Return &Map;
```

This is an example of the section of the MAP layout showing the HTML Area.

Image: MAP Layout with HTML area to launch the PIA page

This example illustrates the section of the MAP layout that contains the HTML area. It is in the Employee Photo container.
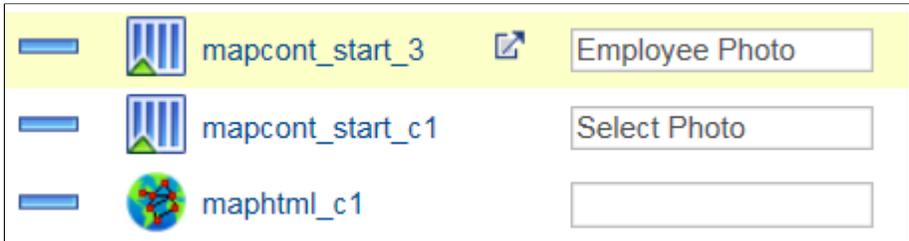
This is an example of the MAP application when the user selects the Select Photo button and then Attach.

Image: MAP application displaying File Attachment page

This example illustrates the MAP application when the user clicks the Select Photo button it will open the PIA page for file attachment. When user clicks Attach button, the file attachment page opens.

Chapter 37
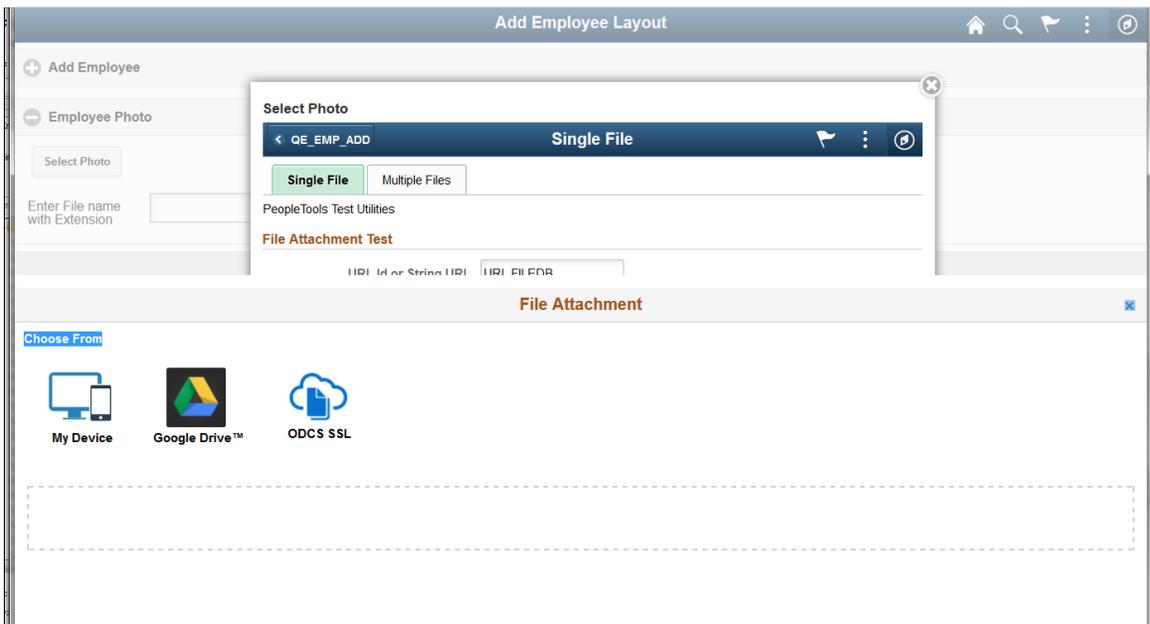
Managing Action Sheet Page Elements

Managing Action Sheet Page Elements

This topic describes how to manage action sheet page element.

Understanding Action Sheet Page Elements

An action sheet page element in a type of container that at runtime pop ups a layer of content on top of the main application.

You can insert the following elements into an action sheet element:

- Check box.

- Flip toggle.

- List view.

- Push button.

- Select box.

- Select menu.

- Static text.

- URL.

Defining Action Sheet Page Elements

This section describes how to:

- Add action sheet page elements to pages.

- Use the Select Icon for Action Sheet page.

- View action sheet elements in the Layout grid.

Adding Action Sheet Page Elements to Pages

To add an action sheet page element, in the Layout grid click the Add Action Sheet button on the Layout grid toolbar on the Layout Designer – Layouts page. To access the page select PeopleTools >Mobile Application Platform >Layout Designer.

    The Add Action Sheet button on the Layout grid toolbar.

After click the Add Action Sheet button, the Select Icon for Action Sheet page appears for you to define the icon or image for the action sheet.

Using the Select Icon for Action Sheet Page

Use the Select Icon for Action Sheet page to define an image or an icon for the action sheet and to define the location where it displays. You can define a static image or icon or define a dynamic one by defining a primitive value from the PeopleSoft document associated with the layout.

Image: Select Icon for Action Sheet page

This example illustrates the fields and controls on the Select Icon for Action Sheet page. You can find definitions for the fields and controls later on this page.

Add Image	Select the box to define an image for the icon.
	When you select this control the Image Selector link appears for you to select an image.
Image Selector	When you select the Add Image control an Image Selector link appears. Click the link to access the Image Selector page to select an image.
Image Name	After you select an image using the Image Selector page, the image name appears in this field.
Icon	Select an icon from the list. The values are:

- *Alert.*

- *Back.*

- *Bars.*

- *Check.*

- *Delete.*

- *Down Arrow.*

- *Edit.*

- *Forward.*

- *Gear.*

- *Grid.*

- *Home.*

- *Info.*

- *Left Arrow.*

- *Minus.*

- *Plus.*

- *Refresh.*

- *Right Arrow.* (Default.)

- *Search.*

- *Star.*

- *Up Arrow.*

Custom	Enter the name of a style class for a custom icon. The class must be defined on the Layout Designer – CSS page. See <u>Using the Layout Designer - CSS Page</u>
Location	From the drop-down list select the location for the icon. The values are: • *Bottom.* • *Left.* • *No Text.* • *Right.* • *Top.*
Select Primitive	Click the link to access the Select Primitive page to define a primitive value.
Element Name	If you have defined a primitive value for the icon, the name of the primitive element appears in the field.

Viewing Action Sheet Page Elements in the Layout Grid

After you define an icon or image for an action sheet the Layout Designer–Layout page appears.

Action sheet elements are indicated in the Layout grid on the Layout Designer –Layout page by the *mapactsht_start* and *mapactsht_end* element IDs.

Image: Action Sheet page element inserted into the Layout grid

This example illustrates the

Enter a value in the Label Text field for the action sheet start element ID to label the element at runtime. By default the field is blank.

To define properties for an action sheet element, click the Properties link to the right of the start ID. Defining properties for action sheet page elements is described in the next section.

To apply custom styles to the element label , use the Class, Style, Theme and No Style Overrides fields and controls. See Defining Styles and Themes for Page Element Labels for more information.

Defining Action Sheet Page Element Properties

This section describes how to define action sheet page element properties.

Understanding Defining Action Sheet Page Element Properties

This topic describes how to define action sheet page element properties using the Action Sheet Properties page.

Many properties that appear on the Action Sheet Properties page also appear on the properties pages for other page elements. These common properties are described in the topic Common Layout Page Element Properties .

This topic discusses using the properties on the Action Sheet Properties page that are unique to working with action sheet page elements only.

Defining Action Sheet Properties

Action sheet page element properties are located on the Action Sheet Properties page. To access the page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the action sheet element for which you want to define properties.

Image: Action Sheet Properties section

This example illustrates the fields and controls in the Action Sheet Properties section of the Action Sheet Properties page. You can find definitions for the fields and controls later on this page.

The following table describes the fields and controls in the Action Sheet Properties section of the Action Sheet Properties page. The fields and controls are listed alphabetically.

Button Location	From the drop-down list select the location for the icon. The values are:

- *Bottom.*

- *Left.*

- *No Text.*

- *Right.*

- *Top.*

Close Button	Select the property to add a Close button on the generated action sheet.
Inline	Select the property so that the length of the button label determines the button width.
Mini	Select the property to create a more compact version of the element.
Remove Corners	Select the property to remove the border radius around the button.
Remove Icon Shadow	Select the property to remove the shadow around the icon.
Remove Shadow	Select the property to remove the shadow around the button.

Chapter 38

Managing Flip Toggle Page Elements

Managing Flip Toggle Page Elements

This topic describes how to manage flip toggle page elements.

Understanding Defining Flip Toggle Page Elements

A flip toggle is a switch that provides a great solution for binary data, such as yes/no, on/off, true/false and so on.

When you define a flip toggle page element you map it to a boolean element defined in the PeopleSoft document associated with the layout. You then create the label for the element by assigning message catalog entries to the boolean.

Defining Flip Toggle Page Elements

This topic discusses how to:

- Add flip toggle elements to pages.

- Use the Select Boolean Primitive for Flip Toggle page.

- Use the Flip Toggle Option Selection page.

- View defined flip toggle page elements in the Layout grid.

Adding Flip Toggle Page Elements to Pages

To add a flip toggle page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

    Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

    The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                        The Current Row button.

4. Click the Add Flip Toggle button on the Layout grid toolbar.

The Add Flip Toggle button.

After you click the button the Select Boolean Primitive for Flip Toggle page appears.

Using the Select Boolean Primitive for Flip Toggle Page

Use the Select Boolean Primitive for Flip Toggle page to select the primitive element to map to the toggle.

Image: Select Boolean Primitive for Flip Toggle page

This example illustrates the fields and controls on the Select Boolean Primitive for Flip Toggle page. You can find definitions for the fields and controls later on this page.

The page displays all the boolean elements defined in the PeopleSoft document associate with the layout.

In the Available Primitive grid, select the boolean to map to the flip toggle element. After you select a primitive element, the Flip Toggle Option Selection page appears:

Using the Flip Toggle Option Selection Page

Use the Flip Toggle Option selection page to define the label text for each option of the toggle, where the value you define in the Option Value 1 section is for one of the toggle options, and the value you define in the Option Value 2 section is for the other toggle options.

Image: Flip Toggle Option Selection page

This example illustrates the fields and controls on the Flip Toggle Option Selection page. You can find definitions for the fields and controls later on this page.

Message Set (Number)	Enter or search for the message set that contains the message to use as a label for the option.
Message Number	Enter or search for the message number that contains the message to use as a label for the option.
Message Text	Displays the message text of the selected message set/message number defined.

Viewing Defined Flip Toggle Page Elements in the Layout Grid

Defined flip toggle elements appear in the Layout grid of the Layout Designer – Layouts page with an element ID that is the name of the boolean primitive element you selected for the flip toggle when you defined it

Image: Flip toggle element defined in the Layout grid

This example illustrates a flip toggle page element defined in the Layout grid.

By default the system also uses the name of the boolean primitive defined for the page element to populate the Label Text field, *APPROVAL_STATUS* in this example. This is the same name defined in the Element Name field for the primitive in the PeopleSoft document defined for the layout. This value appears as the label for the flip toggle at runtime. You can keep the default value, enter a different value, or remove it.

To apply custom styles to the element label , use the Class, Style, Theme and No Style Overrides fields and controls to apply custom styles to the element. See Defining Styles and Themes for Page Element Labels for more information.

Click the Properties link to access the Flip Toggle Properties page to define properties for the element. See the next section for more information about defining properties for flip toggles.

Defining Flip Toggle Page Element Properties

This section describes how to define flip toggle page element properties

Understanding Defining Flip Toggle Page Element Properties

To define properties for flip toggle elements use the Flip Toggle Properties page.

Many properties that appear on the Flip Toggle Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the Flip Toggle Properties page that are unique to working with flip toggle elements. These properties are located in the Flip Toggle Properties section of the page.

Defining Flip Toggle Properties

To access the Flip Toggle Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the flip toggle element for which you want to define properties.

Image: Flip Toggle Properties section

This example illustrates the fields and controls in the Flip Toggle Properties section of the Flip Toggle Properties page. You can find definitions for the fields and controls later on this page.
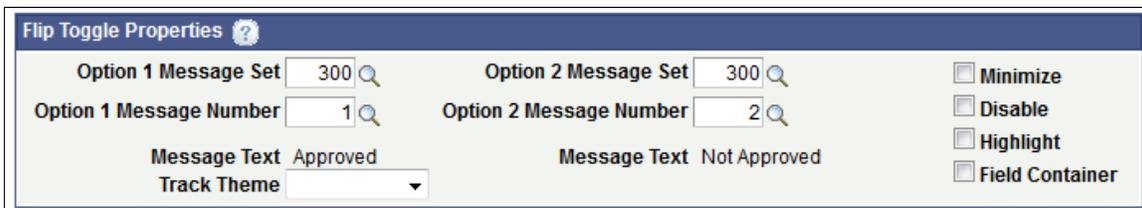
The Flip Toggle Properties section of the Flip Toggle Properties page displays the message catalog entries defined for the element label. You can change the values here as well as define other properties.

Option 1 Message Set / Option 2 Message Set	Enter or search for the message set that contains the message to use as a label for the option.
Option 1 Message Number / Option 2 Message Number	Enter or search for the message number that contains the message to use as a label for the option.
Message Text	Displays the message text of the selected message set/message number defined.
Track Theme	From the drop-down list select a jQuery theme to apply to the track of the flip toggle. See Defining jQuery Mobile Theme Properties for more information about jQuery mobile themes.

Minimize   Select the option to create a more compact version of the element.

Disable   Set this property so that the default state of the flip toggle is disabled.

Highlight   Set this property so that a highlight appears in the flip toggle track.

Field Container   Select this option to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.

Chapter 39

Managing Push Button Page Elements

Defining Push Button Page Elements

This topic discusses how to define push button page elements.

Understanding Defining Push Button Page Elements

This topic discusses how to define push button page elements. When you are done defining a push button page element, you can define additional properties using the Button Properties page.

Push Button Types

The Mobile Application Platform supports two push button types:

Save
When you define a Save type push button you define a PeopleCode Save event to execute when the user selects a push button.

URL
When you define a URL type push button you define one a URL to invoke when the user selects a push button. You can define push button elements to invoke:

- Standard URLs.

- REST services.

- Other Mobile Application Platform layouts.

- Pages within a Mobile Application Platform layout.

- PeopleCode Update events.

The process for defining URLs for push button page elements is very similar to the process for defining URLs for URL page elements.

Grouping Push Button Elements

You can define groups of push button elements in the Mobile Application Platform.

When defining push buttons you are presented with the "Do you want to create a group?" page, which enables you to define an element group and also define the number of push buttons in the group.

Save Type Push Buttons in Pop-Up Container Elements

With one exception, PeopleCode Save event processing via Save type push buttons does not return input data to the application server when defined in pop-up containers. (Pop-up containers are those container elements where the Enable Popup option is enabled in the container's properties.) In these cases use an Ajax event for save processing.

The exception is when pop-up containers are defined in header or footer elements. When pop-up containers are defined in header or footer page elements, PeopleCode Save event processing via Save type push buttons returns data to the application server.

Adding Push Button Page Elements to Pages

To add a push button page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

    Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

    The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                        The Current Row button.

4. Click the Add Pushbutton button on the Layout grid toolbar.

                                        The Add Pushbutton button.

After you click the Add Pushbutton button the "Do you want to create a group?" page appears.

Adding Groups of Push Buttons to Pages

After you click the Add Pushbutton control on the Layout grid toolbar on the Layout Designer – Layout page, the "Do you want to create a group?" page appears, giving you the option to create a group push buttons

Image: "Do you want to create a group?" page

This example illustrates the fields and controls on the "Do you want to create a group?" page. You can find definitions for the fields and controls later on this page.

If you want to create a group of push buttons, enter the number of group items to create in the Number of Group Items field and click the Yes button. If you do not want to create a group of radio button elements click the No button.

After you click Yes to create a group or push button or click No to not create a group of push buttons, the Select Button Type page appears. See the next section for information about how to use the page.

Related Links
Managing Element Groups

Using the Select Button Type Page

Use the Select Button Type page to select the button type.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer, and add a push button element to the Layout grid.

Image: Select Button Type page (Default view)

This example illustrates the fields and controls in the default view of the Select Button Type page. You can find definitions for the fields and controls later on this page.

Save	Select the radio button to define a Save button.
	When you select this button type, additional parameters appear on the page to define the PeopleCode Save event to invoke when the user selects the button.
URL	Select the radio button to define a button that invokes a URL.
	When you select this button type, the Select Button URL page appears for you to select the type of URL type invoke when the users selects the button.

Related Links
Understanding Defining Push Button Page Elements

Defining Push Buttons to Invoke PeopleCode Save Events

When you click the Save radio button on the Select Button Type page additional fields and controls appear to define the PeopleCode Save event to invoke when the user selects the button. Additional fields and controls also appear to define the push button control.

Image: Select Button Type page (Save type)

This example illustrates the fields and controls on the Select Button Type page after you select Save as the button type. You can find definitions for the fields and controls later on this page.

Package	Enter the name of the application package that contains the method to invoke.
Path	Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.
Class ID	Enter the name of the application class that contains the method to invoke.
Method	From the drop-down list select the method to invoke.
Save Redirect	By default, save messages appear in a pop-up message.
	Select this control to display the save message on another page or update the same page, instead of appearing as a pop-up message.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links
Defining Push Button Controls

Using the Select Button URL Page

Use the Select Button URL page to select the URL type to invoke when the user selects a push button.

To access the page click the URL radio button on the Select Button Type page.

Image: Select Button URL page (Default view)

This example illustrates the fields and controls on the default view of the Select Button URL page.

The fields and controls that you use to define a URL depend on the URL type that you choose in the default view of the page.

URL	Click the radio button to define a standard URL to invoke when a user click the push button, for example: `http://www.example.com`
Call REST Service	Click the radio button to define a URL to invoke a REST service when a user clicks the push button.
Call Layout	Click the radio button to define a URL to invoke another Mobile Application Platform layout when a user clicks the push button.
Layout Page	Click the radio button to define a URL to invoke a another page in the current layout when a user clicks the push button.
Update	Click the radio button to define a URL that executes a PeopleCode Update event when a user clicks the push button.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links
Defining Push Button Controls

Defining Push Buttons to Invoke Standard URLs

When defining a push button to invoke a standard URL, you can specify a fully-qualified URL or an anchor using an id attribute (#<id>).

To define a standard URL click the URL radio button on the Select Button URL page.

Image: Select Button URL page (Call standard URL)

This example illustrates the fields and controls on the Select Button URL page when you are defining a push button that invokes a standard URL. You can find definitions for the fields and controls later on this page.

URL	Enter the fully-qualified URL or anchor in the long edit box.
Override URL	Select the box to define a dynamic URL using a primitive element defined in the PeopleSoft document associated with the layout.
Select primitive for mapping	This link appears if you select the Override URL box.
	Click the link to access the Select Primitive page to select a primitive.
Element Name	If you define a primitive for the URL, the primitive element name appears in this field.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links
Using the Select Primitive Page
Defining Push Button Controls

Defining Push Buttons to Invoke REST Services

You can define a push button to call provider or consumer REST services. When defining this URL type, you select the REST service operation, select the appropriate URI template, and then identify the variables to pass.

To define a push button that calls a REST service click the Call REST Service radio button on the Select Button URL page.

Image: Select Button URL page (Call REST service)

This example illustrates the fields and controls on the Select Button URL page when you are defining a push button that calls a REST service.

Click the Select Rest Operation link to access the URI Selection page to build the REST URI.

After you build the URI an Operation field appears on the page that displays the name of the REST service operation.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links
Using the URI Selection Page
Defining Push Button Controls

Defining Push Buttons to Invoke Layouts

When you define a push button to call another layout you must select the name of the layout to call, define the appropriate layout initialization to use, and identify the variables to pass.

To define a push button to call another layout click the Call Layout radio button on the Select Button URL page.

Image: Select Button URL page (Call Layout)

This example illustrates the fields and controls on the Select Button URL page when you are defining a push button to call another Mobile Application Platform layout. You can find definitions for the fields and controls later on this page.

When you click the Call Layout radio button a Select Layout link appears. Click the link to access the Layout Selection page to select the layout to call.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links

Using the Layout Selection Page
Defining Push Button Controls

Defining Push Buttons to Invoke Pages Within Layouts

Click the Layout Page radio button on the Select Button URL page to define a push button to call a page within the same layout.

The fields and controls that appear on the page depend on if you are defining a URL to a page defined as an Internal type page or an External type page.

Image: Select Button URL page (Call Internal type page in layout)

This example illustrates the fields and controls on the Select Button URL page when are defining a push button to call an Internal type page in the current layout. You can find definitions for the fields and controls later on this page.

Image: Select Button URL page (Call External type page in same layout)

This example illustrates the fields and controls on the Select Button URL page when are defining a push button to call an External type page in the current layout. You can find definitions for the fields and controls later on this page.

Page	From the drop-down list select the number of the layout page to call.
Select Layout	This link appears when you select an External type page from the Page drop-down list.
	Click the link to access the Layout Selection page to select the layout to call.
Post	This control appears when you select an External type page from the Page drop-down list.
	By default the external page is called via an HTTP GET. If data to be passed as arguments needs to be more secure, select the box to make the request via an HTTP POST.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links

Using the Layout Selection Page
Defining Page and Layout Navigation
Defining Push Button Controls

Defining Push Buttons to Invoke PeopleCode Update Events

Click the Update radio button on the Select Button URL page to define a PeopleCode Update event to call when a user clicks a push button.

Image: Select Button URL page (Call PeopleCode Update event)

This example illustrates the fields and controls on the Select Button URL page when you are defining a push button to call a PeopleCode update event. You can find definitions for the fields and controls later on this page.

Map to URI	Click the link to access the Select URI page to select the URI parameters to pass to the Update event.
	The Select URI page displays the URIs defined for the current layout, as defined in the initialization parameters for the layout.
URI	After you define the URI parameters to pass using the Select URI page, the URI string displays in this field.
Root Package ID	Enter the name of the application package that contains the method to invoke.
Path	Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.
Class ID	Enter the name of the application class that contains the method to invoke.
Method	From the drop-down list select the method to invoke.

The remaining controls on the page are used to define the push button control. These fields are described later in this topic.

Related Links
Using the Select URI Page
Defining Push Button Controls

Defining Push Button Controls

At the bottom of the Select Button URL page are fields and controls to define push button control that the user selects on the user interface.

To define the control you can upload an image from the database, use one of the delivered icons, or define a control using CSS3 style or class. You also have the option of defining a dynamic image by defining a primitive value for image.

Image: Select Button URL page (Button and image fields and controls)

This example illustrates the fields and controls on the Select Button URL page for defining push button controls. You can find definitions for the fields and controls later on this page.

Add Image	Select the control to add image that becomes the hot spot for the push button control.
Image Selector	This link appears if you select the Add Image control.
	Click the link to access the Image Selector page to select an image.
Image Name	After you select an image using the Image Selector page, the image name appears in this field.
Icon	From the drop-down list select an icon for the push button. The values are:

- *Alert.*

- *Back.*

- *Bars.*

- *Check.*

- *Delete.*

- *Down Arrow.*

- *Edit.*

- *Forward.*

- *Gear.*

- *Grid.*

- *Home.*

- *Info.*

- *Left Arrow.*

- *Minus.*

- *Plus.*

- *Refresh.*

- *Right Arrow.* (Default.)

- *Search.*

- *Star.*

- *Up Arrow.*

Custom	Enter the class or style name of a custom icon defined on the Layout Designer – CSS page, or in the Template Designer – CSS page if a layout template is applied to the layout.
Location	From the drop-down list select the location for the icon relative to any icon label text defined. The values are:

- *Bottom.*

- *Left.*

- *No Text.*

- *Right.*

- *Top.*

Select Primitive	Click the link to access the Select Primitive page to select an image defined as a primitive element to display dynamically. In PeopleCode you must populate the primitive element with the fully qualified URL of the image to display.

Related Links

Using the Image Selector Page

Viewing Defined Push Button Elements in the Layout Grid

After you define a push button element using the Select Button URL page the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for push button page elements is *mapbutton.*

Image: Push button element defined in the Layout grid

This example illustrates a partial view of the Layout grid showing a push button defined.

The value that you enter in the Label Text field becomes the button label in the generated HTML. In this example *Expense Report* is the label that appears on the button at runtime.

Click the Properties link to access the Button Properties page to define properties for the element.

Use the other fields to apply CSS3 classes and styles to the button label, as well as apply jQuery themes.

Related Links
Setting Alternate Page Element IDs
Defining Display Labels for Page Elements
Defining Styles and Themes for Page Element Labels

Defining Push Button Page Element Properties

This topic discusses how to define properties for push button page elements.

Understanding Defining Push Button Page Element Properties

To define properties for push button page elements, use the Button Properties page. To access the page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid click the Properties link for the push button element for which you want to define properties.

Many properties that appear on the Button Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

Define Properties for Save Type Push Buttons

The properties for push buttons that invoke PeopleCode events are consolidated in the Button Properties section of the Button Properties page.

Use the section to define button control properties and to modify PeopleCode Save events defined for a push button.

See Defining Properties for Push Button Controls later in this topic.

Defining Properties for URL Type Push Buttons

The fields and controls for defining properties for URL type push button page elements are identical to those for defining properties for URL page elements. See the topic Defining URL Properties for Page Elements for more information.

Defining Properties for Push Button Controls

Use the Button Properties section of the Button Properties page to define properties for the push button control.

The properties that are available to set for a push button control depend on the type of push button for which you are defining properties, URL type or Save type.

URL type push buttons are those that call standard URLs, REST services, other layouts, pages within a layout, and PeopleCode Update events. Save type push buttons are those that call PeopleCode Save events.

Image: Button Properties section (URL type push button)

This example illustrates the fields and controls in the Button Properties section of the Button Properties page when you are defining properties for a URL type push button. You can find definitions for the fields and controls later on this page.

When working with the properties for a Save type button, a Save Event section appears that enables you to modify the PeopleCode Save event defined for the button.

Image: Button Properties section (Save type push button)

This example illustrates the fields and controls in the Button Properties section of the Button Properties page when you are defining properties for a Save type push button that invokes a PeopleCode Save event. You can find definitions for the fields and controls later on this page.
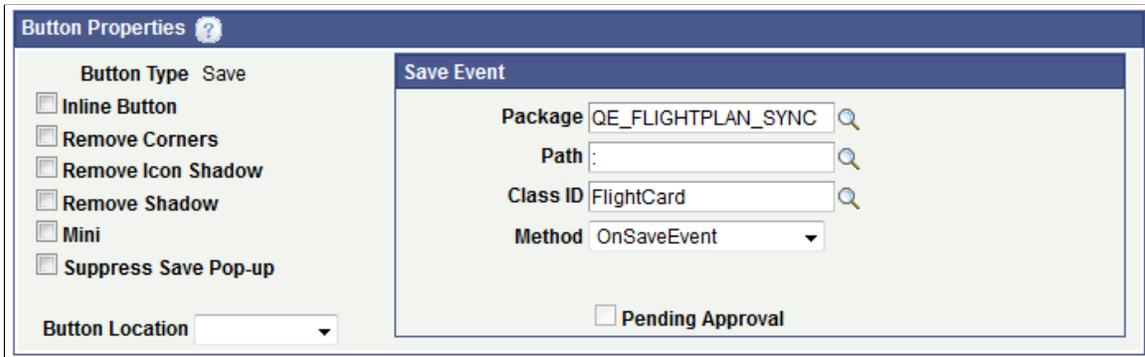
Button Type	Displays the type of push button. The values are:

- *Save.* Push button that invokes a PeopleCode Save event.

- *URL.* Push button that invokes a standard URL, REST service, layout, page, or PeopleCode Update event.

Inline Button	Select the property so that the length of the button label determines the button width.
Remove Corners	Select the property to remove the border radius around the button.
Remove Icon Shadow	Select the property to remove the shadow around the icon.
Remove Shadow	Select the property to remove the shadow around the button.
Mini	Select the property to create a more compact version of the button.
Suppress Save Pop-Up	This property appears in the Button Properties section only when you are defining properties for a Save type push button.

Select the property to suppress the display of the save pop-up box.

Location	From the drop-down list select the location of the button on the page. The values are:

- *Bottom.*

- *Left.*

- *No Text.*

- *Right.*

- *Top.*

Save Event section

This section appears in the Button Properties section only when you are defining properties for a Save type push button.

Use the following fields to modify the PeopleCode Save event defined for the push button:

- Package. Enter the name of the application package that contains the method to invoke.

- Path. Enter the names of each sub-package in the application class hierarchy that define the location of the application class. Separate sub-package names by a colon. If the class is defined in the top-level application package, enter or select the colon.

- Class ID. Enter the name of the application class that contains the method to invoke.

- Method. From the drop-down list select the method to invoke.

Chapter 40

Managing Input Page Elements

Managing Input Page Elements

This topic describes how to manage input page elements.

Defining Input Page Elements

This section describes how to:

- Add input page elements to pages.

- Use the Input Selection page.

- View defined input page elements in the Layout grid.

Adding Input Page Elements to Pages

To add an input page element:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                       The Current Row button.

4. Click the Add Input button on the Layout grid toolbar.

                                       The Add Input button.

After you click the Add Input button, the Input Selection page appears.

Using the Input Selection Page

Image: Input Selection page

This example illustrates the fields and controls on the Input Selection page. You can find definitions for the fields and controls later on this page.

Use the Input Selection page to select the primitive for the input field.

Select	Select the primitive for the input element from the list.

After you select the value for the element the Layout Designer – Layout page appears.

Viewing Defined Input Page Elements in the Layout Grid

After you define an input element by selecting a primitive value for it, the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for input page elements is the name of the primitive you defined for it., for example *<primitive_name>_n*.

Image: Input element defined in the Layout grid

This example illustrates an input element defined in the Layout grid.

In the example, *WEEK_1* is the element ID.

By default the system populates the Label Text field with the name of the primitive that you defined for the element, *week* in the example. This is the same name defined in the Element Name field for the primitive in the PeopleSoft document. This value appears as the label for the input element. You can keep the default value, define a different value, or remove it.

Click the Properties link to access the Input Properties page to define properties for the element.

Use the other fields to apply CSS3 classes and styles to the button label, as well as apply jQuery themes.

Related Links

Setting Alternate Page Element IDs

Defining Input Page Element Properties

Understanding Defining Input Page Element Properties

This topic describes how to define input page element properties using the Input Properties page.

Many properties that appear on the Input Properties page also appear on the properties pages for other page elements. These common properties are described in the topic Common Layout Page Element Properties. This topic discusses using the properties on the Input Properties page that are unique to working with input page elements only.

Understanding Input Types

One of the properties you can define for an input page element is the input type.

By default the framework defaults the input type to the data type of the primitive you selected when you defined the element. You can select a different input type to correspond to the primitive type. For example, a primitive type String defaults in as Text. But you can select another input type.

The Mobile Application Platform supports the basic HTML5 inputs shown in the following table:

HTML Inputs	*Considerations*
• Date. • DateTime. • Month. • Time. • Week.	Keep in mind that HTML5 date fields are not supported on every browser, whether mobile or desktop.
• Email. • Number. • Telephone. • URL.	When any of these input types are defined, the appropriate optimized virtual keyboard will be displayed.

Defining Input Properties

Input page element properties are located on the Input Properties page. To access the page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the input element for which you want to define properties.

Image: Input Properties section

This example illustrates the fields and controls in the Input Properties section of the Input Properties page. You can find definitions for the fields and controls later on this page.

The following table describes the fields and controls in the Input Properties section of the Input Properties page. The fields and controls are listed alphabetically.

Auto Focus	Select the property for the input element to automatically get focus when the page loads.
Class	Enter the name of a custom CSS3 class for the element.
	The class must be defined on the Layout Designer – CSS page.
Field Container	Select this property to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.
Input Type section	Select the type of input desired.
	By default the system populates the input type with the data type of the primitive element defined when the element is created.
	If no input is selected the default value (*Text*) is used.
	The valid values are:
	• *Date.* A date selector (day, month, and year).
	• *DateTime.* Complete date selector (day, month, year, hour, and minutes) using standard syntax including GMT time zone.
	• *Time.*
	• *Number.*
	• *Text.* (Default.)
	• *Password.*

- *Input Range.* When you select this input type the Max and Min fields become enabled for you to define the minimum and maximum range.

- *Month.*

- *URL.*

- *Telephone.*

- *Week.* For a week of the year selector.

- *Email.*

- *Hide Input Field.*

- *Text Area.* If you choose this input type the primitive type must be *Text* (unbounded). The text area will automatically grow based on the amount of data entered.

Minimize	Select the option to create a more compact version of the element.
Pattern section	Use the properties in this section to define a message catalog entry to describe the pattern expected for data input. For example, if you specify a date, you may require that users enter it in a *mm-dd-yyyy* format.

- *Msg Set.* Select the message set of the message catalog entry that contains the label to use.

- *Msg Num.* Select the message number of the message catalog entry that contains the label to use.

- *Message Text.* Displays the text of the message catalog entry for the message set and message number defined.

Place Holder	Select the property to define a short hint of the expected value of the input field.
Place Holder **Msg Set** and **Msg Num**	These field become enabled when you select the placeholder property. Define the hint message using a message catalog entry by defining a message set and message number.

- *Msg Set.* Select the message set of the message catalog entry that contains the label to use.

- *Msg Num.* Select the message number of the message catalog entry that contains the label to use.

Required	Specify that an input field must be filled out before submitting the form.
Search	Select this property to define the input with a search field.

Search fields are styled inputs. They start out with a search icon (magnifying glass) in the left of the input field. As the user begins to type, a clear icon (an "x") will appear in the right f the input field. Tapping on the stop icon will clear the user's input.

Style   Enter the name of a custom style for the element.

The class must be defined on the Layout Designer – CSS page.

Type   Displays the data type of the primitive element defined for the element.

Chapter 41

Managing Select Box Page Elements

Managing Select Box Page Elements

This topic discusses how to manage select box page elements.

Understanding Select Box Page Elements

A select box page element is equivalent to drop-down list box. The content is typically populated using a primitive element wrapped in a collection.

Defining Select Box Page Elements

This section discusses how to:

- Add select box page elements to pages.

- Use the Select Primitive for Select Box page.

- Use the Select Primitive for Value page.

- View defined select box page elements in the Layout grid.

Adding Select Box Page Elements to Pages

To add a select box page element:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

    The Current Row button.

4. Click the Add Select Box button on the Layout grid toolbar.

    The Add Select Box button.

After you click the Add Select Box button, the Select Primitive for Select Box page appears for you to define the element.

Using the Select Primitive for Select Box Page

Use the Select Primitive for Select Box Page to define the value for the select box.

Image: Select Primitive for Select Box

This example illustrates the fields and controls on the Select Primitive for Select Box. You can find definitions for the fields and controls later on this page.
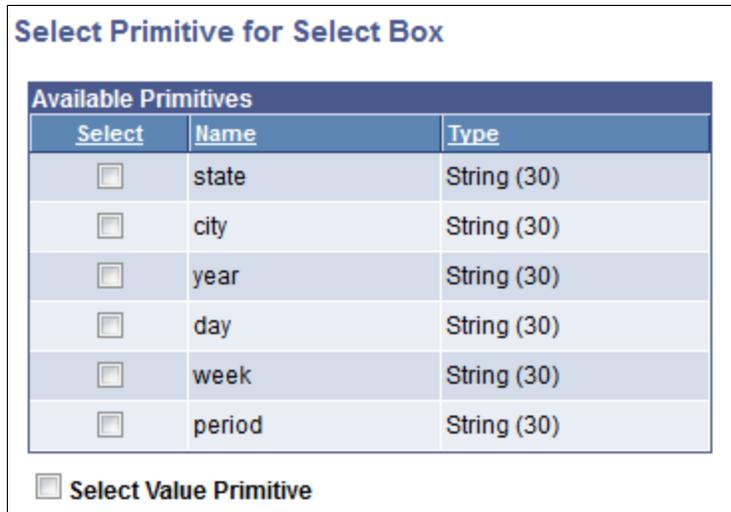
The primitive elements defined in the PeopleSoft document defined for the layout appear on the page.

Select	Select the primitive for the select box from the list.
Select Value Primitive	Select the control to use a different primitive for the value. This value is passed back to the application server and subsequently read in PeopleCode.
	If you select this control the Select Primitive for Value page appears for you to select the primitive to use for the value.

Using the Select Value for Primitive Page

If you select the Select Value Primitive control on the Select Primitive for Select Box page, the Select Primitive for Value page appears.

Image: Select Primitive for Value page

This example illustrates the fields and controls on the Select Primitive for Value page. You can find definitions for the fields and controls later on this page.

In the grid, select the primitive to use for the select box value.

Viewing Defined Select Box Page Elements in the Layout Grid

After you define a primitive value for the select box, the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for select box page elements is the name of the primitive you define for it., for example *<primitive_name>_n.*

Image: Select box page element defined in the Layout grid

This example illustrates a select box page element defined in the Layout grid.

By default the system populates the Label Text field with the name of the primitive that you defined for the element. This is the same name defined in the Element Name field for the primitive in the PeopleSoft document. This value appears as the label for the select box at runtime. You can keep the default value, define a different value, or remove it.

Click the Properties link to access the Select Box Properties page to define properties for the element.

Use the other fields to apply CSS3 classes and styles to the button label, as well as apply jQuery themes.

Related Links

Setting Alternate Page Element IDs
Defining Display Labels for Page Elements
Defining Styles and Themes for Page Element Labels

Defining Select Box Page Element Properties

This section discusses defining select box page element properties.

Understanding Defining Select Box Page Element Properties

To define properties for select box page elements, use the Select Box Properties page.

Many properties that appear on the Select Box Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the Select Box Properties page that are unique to working with select box page elements. These properties are located in the Select Box Properties section of the page.

Defining Select Box Properties

To access the Select Box Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the select box element for which you want to define properties.

Image: Select Box Properties section

This example illustrates the Select Box Properties section of the Select Box Properties page.

Field Container	Select this property to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.

Chapter 42

Managing Radio Button Page Elements

Managing Radio Button Page Elements

This topic discusses how to manage radio button page elements.

Defining Radio Button Page Elements

This section discusses how to:

- Add radio button page elements to pages.

- Add groups of radio buttons to pages.

- Use the Select Boolean Primitive for Radio Button page.

- View defined radio button page elements in the Layout grid.

- Define display labels for radio button page elements.

Understanding Defining Radio Page Elements

To define a radio button page element you specify a boolean primitive element defined in the PeopleSoft document associated with the layout definition.

Adding Radio Button Page Elements to Pages

To add a radio button page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                       The Current Row button

4. Click the Add Radio Button button.

                       The Add Radio Button button.

After you select a position for the element in the Layout grid and click the Add Radio Button button , the "Do you want to create a group?" page appears:

Adding Groups of Radio Buttons to Pages

After you click the Add Radio Button control on the Layout grid toolbar on the Layout Designer – Layout page, the "Do you want to create a group?" page appears, giving you the option to create a group of radio buttons.

Image: "Do you want to create a group?" page

This example illustrates the fields and controls on the "Do you want to create a group?" page. You can find definitions for the fields and controls later on this page.

If you want to create a group of radio buttons, enter the number of group items to create in the Number of Group Items field and click the Yes button. If you do not want to create a group of radio button elements click the No button.

For more information about creating element groups, see Managing Element Groups

After you click Yes to create a group or radio buttons or click No to not create a group of radio buttons, the Select Boolean Primitive for Radio Button page appears. See the next section for information about how to use the page.

Using the Select Boolean Primitive for Radio Button Page

Use the Select Boolean Primitive for Radio Button page to select a value for the element.

Image: Select Boolean Primitive for Radio Button page

This example illustrates the fields and controls on the Select Boolean Primitive for Radio Button page. You can find definitions for the fields and controls later on this page.
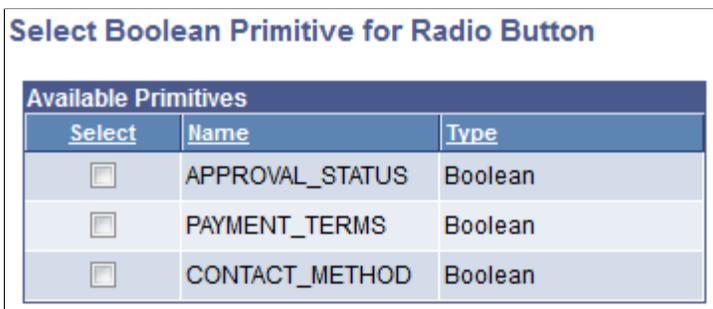
The boolean primitive elements defined in the PeopleSoft document defined for the layout appear on the page. The page displays only those boolean primitive elements that are not already defined for other elements in the layout.

Select   Select the boolean primitive for the radio button from the list.

After you select the value for the element the Layout Designer–Layout page appears.

Viewing Defined Radio Button Page Elements in the Layout Grid

After you define a radio button using the Select Boolean Primitive for Radio Button page, the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for a radio button page element is the name of the primitive you defined for it., for example *<primitive_name>_n.*

By default the system populates the Label Text field with the name defined in the Element Name field in the PeopleSoft document.

Image: Radio button element defined in the Layout grid

This example illustrates a radio button element defined in the Layout grid.

In the example, *APPROVAL_STATUS_1* is the element ID and *APPROVAL_STATUS* is defined as the label text. These values are defaults from the primitive element defined for the radio button.

Defining Display Labels for Radio Button Page Elements

The value in the Label Text field for a radio button element is the label that appears for the button at runtime. And as described previously, the default value is the value defined in the Element Name field for the primitive in the PeopleSoft document.

You can keep the default value, define a different value, or remove it.

In the same row as the radio button element are class, style, and theme fields. Use these fields to define a custom CSS3 class name and custom CSS3 style name for the display label. (Note that any class or style you enter must be defined in the Layout Designer – CSS page.) Use the theme field to define a jQuery theme for the display label.

Related Links

Defining jQuery Mobile Theme Properties
Setting Alternate Page Element IDs
Defining Display Labels for Page Elements

Defining Radio Button Page Element Properties

This section discusses defining radio button page element properties.

Understanding Defining Radio Button Page Element Properties

To define properties for radio button page elements, use the Radio Button Properties page.

Many properties that appear on the Radio Button Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the Radio Button Properties page that are unique to working with radio button page elements. These properties are located in the Radio Button Properties section of the page.

Defining Radio Button Properties

To access the Radio Button Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the radio button element for which you want to define properties.

Image: Radio Button Properties section

This example illustrates the fields and controls in the Radio Button Properties section of the Radio Button Properties page. You can find definitions for the fields and controls later on this page.

Minimize	Select the property to create a more compact version of the element.
Field Container	Select the property to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.

Chapter 43

Managing Check Box Page Elements

Managing Check Box Page Elements

This topic discusses how to manage check box page elements.

Understanding Defining Check Box Page Elements

To define a check box page element you specify a boolean primitive element defined in the PeopleSoft document associated with the layout definition.

Defining Check Box Page Elements

This section discusses how to:

- Add check box page elements to pages.

- Use the Select Boolean Primitive for Checkbox page.

- View defined check box page elements in the Layout grid.

- Define check box display labels.

Adding Check Box Page Elements to Pages

To add a check box page element:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

    The Current Row button.

4. Click the Add Checkbox button on the Layout grid toolbar.

    The Add Checkbox button.

After you select a position for the element in the Layout grid and click the Add Checkbox button, the "Do you want to create a group?" page appears:

Image: "Do you want to create a group?" page

This example illustrates the fields and controls on the "Do you want to create a group?" page. You can find definitions for the fields and controls later on this page.

Do you want to create a group?
Number of Group Items [ ]

If you want to create a group of radio buttons, enter the number in the Number of Group Items field and click the Yes button. If you do not want to create a group of check box element click the No button. For more information about creating element groups, see Managing Element Groups

After you determine whether to create a group of check box elements, the Select Boolean Primitive for Checkbox page appears.

Using the Select Boolean Primitive for Checkbox Page

Use the Select Boolean Primitive for Checkbox page to select a value for the element.

Image: Select Boolean Primitive for Checkbox page

This example illustrates the fields and controls on the Select Boolean Primitive for Checkbox page. You can find definitions for the fields and controls later on this page.
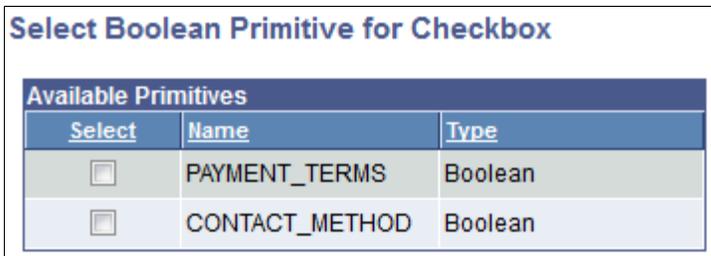
Select Boolean Primitive for Checkbox

Available Primitives		
Select	Name	Type
☐	PAYMENT_TERMS	Boolean
☐	CONTACT_METHOD	Boolean

The boolean primitive elements defined in the PeopleSoft document defined for the layout appear on the page. The page displays only displays those elements that are not already defined for an element.

Select   Select the boolean primitive for the radio button from the list.

After you select the value for the element the Layout Designer – Layout page appears.

Viewing Defined Check Box Elements in the Layout Grid

After you define a radio button by selecting a primitive value for it, the Layout Designer – Layout page appears and the element appears in the Layout grid.

The default element ID for radio button page elements is the name of the primitive you defined for it., for example *<primitive_name>_n*.

Image: Check box element defined in the Layout grid

This example illustrates a check box element defined in the Layout grid.

In the example, *PAYMENT_TERMS_1* is the element ID.

By default the system populates the Label Text field with the name of the primitive that you defined for the element, *PAYMENT_TERMS* in the example.

Click the Properties link to access the Checkbox Properties page to define properties for the element.

Defining Display Labels for Checkbox Page Elements

The value in the Label Text field for a check box element is the label that appears for the element at runtime. And as described previously, the default value is the value defined in the Element Name field for the primitive in the PeopleSoft document.

You can keep the default value, define a different value, or remove it.

In the same row as the element are class, style, and theme fields. Use these fields to apply a CSS3 classes and styles to the button label, as well as apply a jQuery theme.

Related Links
Setting Alternate Page Element IDs
Defining Display Labels for Page Elements
Defining Styles and Themes for Page Element Labels

Defining Check Box Page Element Properties

This section discusses defining radio button page element properties.

Understanding Defining Check Box Page Element Properties

To define properties for check box page elements, use the Checkbox Properties page.

Many properties that appear on the Checkbox Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the Checkbox Properties page that are unique to working with check box page elements. These properties are located in the Checkbox Properties section of the page.

Defining Check Box Properties

To access the Checkbox Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the check box element for which you want to define properties.

Image: Checkbox Properties section

This example illustrates the fields and controls in the Checkbox Properties section of the Checkbox Properties page. You can find definitions for the fields and controls later on this page.

Minimize	Select the property to create a more compact version of the element.
Field Container	Select the property to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.

Chapter 44

Managing Slider Page Elements

Managing Slider Page Elements

This topic provides an overview of managing slider page elements.

Understanding Managing Slide Page Element

To define a slider page element you specify a integer primitive element defined in the PeopleSoft document associated with the layout definition.

Defining Slider Page Elements

This section discusses how to:

- Add slider page elements to pages.

- Use the Select Integer Primitive for Slider page.

- Use the Slider Properties page.

- View defined slider page elements in the Layout grid.

- Define display labels for slider page elements.

Adding Slider Page Elements to Pages

To add a slider page element:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                   The Current Row button.

4. Click the Add Slider button on the Layout grid toolbar.

The Add Slider button.

After you select a position for the element in the Layout grid and click the Add Slider button, the Select Integer Primitive for Slider page appears for you to define the element.

Using the Select Integer for Slider Page

Image: Select Integer for Slider page

This example illustrates the fields and controls on the Select Integer for Slider page. You can find definitions for the fields and controls later on this page.

The integer primitive elements defined in the PeopleSoft document defined for the layout appear on the page. The page displays only displays those elements that are not already defined for other elements in the layout.

Select   Select the integer primitive for the slider from the list.

After you select the value for the element the Slider Properties page appears for you to define the beginning and end ranges for the slider, as well as the (incremental) steps.

Using the Slider Properties Page

Use the Slider Properties page to define the minimum range, maximum range, and step value for the slider.

Image: Slider Properties page

This example illustrates the fields and controls on the Slider Properties page. You can find definitions for the fields and controls later on this page.

For example, if you set the Slider Minimum to *0,* the Slider Maximum to *1000,* and the Slider Step to *50,* at runtime the user will be able to move the slider from *0* to *1000* in increments of *50* units.

Slider Minimum	Enter an integer value for the slider minimum.
Slider Maximum	Enter an integer value for the slider maximum.
Slider Step	Enter an integer value for the slider step.

After you define the values on this page and click the OK button, the Layout Designer – Layout page appears.

Viewing Defined Slider Page Elements in the Layout Grid

Slide page elements are initially indicated in the Layout grid on the Layout Designer – Layout page by the name of the primitive you defined for it, for example *<primitive_name>_n*.

Image: Slider element defined in the Layout grid

This example illustrates a slider element defined in the Layout grid.

In the example, the element ID is *Slider_1_1*. To define an alternate label for the ID click the element ID to access the Add Alternate ID

By default the system populates the Label Text field with the name of the primitive that you defined for the element using the Select Integer Primitive for Slider page, *Slider* in the example. This is the same name defined in the Element Name field for the primitive in the PeopleSoft document. This value appears as the label for the slider at runtime. You can keep the default value, define a different value, or remove it. You can also apply styles, classes and themes to the label, using the fields to the right of the Properties link. See

Click the Properties link to access the Slider Properties page to define properties for the element. Defining slider page element properties is described in the next section.

Defining Display Labels for Slide Page Elements

The value in the Label Text field

The value in the Label Text field for a slider element is the label that appears for the element at runtime. And as described previously, the default value is the value defined in the Element Name field for the primitive in the PeopleSoft document.

You can keep the default value, define a different value, or remove it.

In the same row as the element are class, style, and theme fields. Use these fields to define a custom CSS3 class name and custom CSS3 style name for the display label. (Note that any class name you enter must be defined in the Layout Designer – CSS page.) Use the theme field to define a jQuery theme for the display label.

Related Links

Setting Alternate Page Element IDs
Defining Display Labels for Page Elements
Defining Styles and Themes for Page Element Labels

Defining Slider Page Element Properties

Understanding Defining Slider Page Element Properties

To define properties for slide page elements, use the Slider Properties page.

Many properties that appear on the Radio Button Properties page also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on the Slider Properties page that are unique to working with slider page elements. These properties are located in the Slider Properties section of the page.

Defining Slider Properties

To access the Slider Properties page, in the Layout grid on the Layout Designer – Layout page click the Properties link next to the slider element for which you want to define properties.

Image: Slide Properties page – Slider Properties section

This example illustrates the fields and controls in the Slider Properties section of the Slider Properties page.

Slider Minimum	Enter an integer value for the slider minimum.
Slider Maximum	Enter an integer value for the slider maximum.
Slider Step	Enter an integer value for the slider step.
Default Value on Track	Enter an integer value to set the default position of the slider handle.
Track Theme	From the drop-down list select a jQuery theme to apply to the slider track. For more information about jQuery themes in the Mobile Application Platform, see Defining jQuery Mobile Theme Properties
Minimize	Select the property to create a more compact version of the element.
Disable	Select the property to disable the slider.
Highlight	Select the property to highlight the track left of the handle.
Field Container	Select the property to enable a field container for the element to assist in rendering the element on wider screens, such as in table devices.

Chapter 45

Managing Table Page Elements

Managing Table Page Elements

This topic discusses how to manage table page elements.

Defining Table Page Elements

This section discusses how to:

- Add table page elements to pages.

- Use the Compound Selection page.

- View defined table page elements in the Layout grid.

- Define display labels for table page elements.

Understanding Defining Table Page Elements

Tables in the Mobile Application Platform consist of columns of data.

To define a table, the PeopleSoft document associated with the layout must contain a compound element that contains primitive elements. And that compound element must be defined in a collection element. This type of nesting is referred to as *a compound of primitives wrapped in a collection.*

Image: Document structure for a table element

This example illustrates the document structure for a table element in the Mobile Application Platform.

Note: The highlighted areas in the image are system-generated and do not indicate emphasis for documentation purposes.

The document is structured as follows:

1. *Sales* is the root element of the PeopleSoft document and was added to the system when the document was added to the system.

2. Added to the root element is a collection child element named *SalesCollection.*

3. Added to the collection element is a compound child element named *SalesCompound.*

4. Added to the compound element are a number of primitive child elements. In this example, the primitive elements are fields from a PeopleSoft record.

   The primitives (or PeopleSoft record fields) will be the columns in the table in the Mobile Application Platform application.

Adding Table Page Elements to Pages

To add a table page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                           The Current Row button.

4. Click the Add Table button in the Layout grid toolbar.

                                           The Add Table button.

See <u>Inserting Page Elements into the Layout Grid</u> for more information about positioning and inserting elements into the Layout grid.

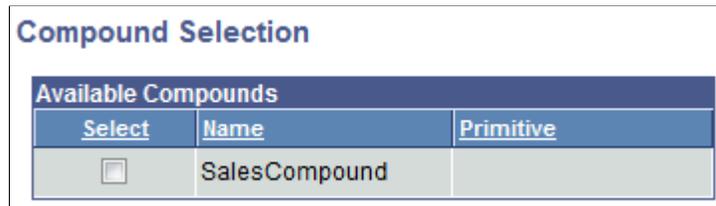After you click the Add Table button the Compound Selection page appears to select the compound element for the table.

Using the Compound Selection Page

Use the Compound Selection page to define the compound for the table. As described previously in this topic, the compound element must contain the primitive elements to use as the table columns and must be wrapped/defined in a collection element

Image: Compound Selection page

This example illustrates the fields and controls on the Compound Selection page. You can find definitions for the fields and controls later on this page.

The compound elements defined in the PeopleSoft document defined for the layout appear on the page. The page displays only those compound elements that are not already defined for other elements in the layout.

Select   Select the compound element to use for the table.

After you select the value for the element the Layout Designer–Layout page appears.

Viewing Defined Table Page Elements in the Layout Grid

After you define a table using the Compound Selection page the Layout Designer – Layout page appears and the element appears in the Layout grid.

Image: Table page element defined in the Layout grid

This example illustrates a table page element defined in the layout grid.

There are two types of icons that appear in the grid for a table element:

                                        Indicates the table structure.

                                        Indicates a table column.

The default element ID for a table element is the name of the compound document element you defined for it, for example *<compound_name>_n*. The grid also displays each primitive document element defined in compound element, using the name of the primitive element as defined in the document, for example *<primitive_name>_n*.

By default the system populates the Label Text field for the compound element and primitive elements in the grid with the value defined in the Element Name field for each element in the PeopleSoft document.

In the example, *SalesCompoud_1* is the element ID for the table and *SalesCompound* is the label text for the table. Both values are defaults from the compound defined for the element.

CUST_ID_1, CUSTOMER_TYPE_1, ITEM_1 and the remaining values in the first column in the grid are element IDs for the primitive elements defined in the compound element. *CUST_ID, CUSTOMER_TYPE, ITEM* and the remaining values in the second column in the grid are the label text values. These are default values from the primitive elements defined for the compound.

Defining Table Display Labels

The values in the Label Text fields for a table element are the labels that appear for the table at runtime.

The label text value for the compound is the table label and the value for a primitive element is a column label.

And as described previously, the default values are those defined in the Element Name fields for the compound element and primitive elements in the PeopleSoft document.

You can keep the default values, define different values, or remove them.

In the same row for each element ID are class, style, and theme fields. Use these fields to define custom CSS3 class names and custom CSS3 style names for the display labels. (Note that any classes or styles you enter must be defined in the Layout Designer – CSS page.) Use the theme fields to define jQuery themes for the display labels.

Related Links
Defining jQuery Mobile Theme Properties
Setting Alternate Page Element IDs
Defining Display Labels for Page Elements

Defining Table Page Element Properties

This section discusses how to:

- Use the Table Properties page.

- Use the Table Element Properties page.

Understanding Defining Table Page Element Properties

To define properties for table page elements, use the Table Properties page and the Table Element Properties page.

Many properties that appear these pages also appear on the properties pages for other page elements. For more information and links to topics about these common properties, see the topic Common Layout Page Element Properties

This topic discusses using the properties on Table Properties page and the Table Element Properties page the that are unique to working with table page elements.

Using the Table Properties Page

Use the Table Properties page to define properties for the compound element of the table.

To access the page:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

    The Layout Designer – Layout page appears.

2. In the Layout grid, locate the row that contains the compound element for the table.

3. Click the Properties link.

Image: Table Properties page — Header section

This example illustrates the fields and controls on the Table Properties page — Header section. You can find definitions for the fields and controls later on this page.
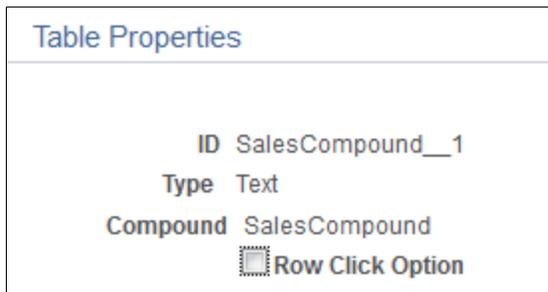
The compound name is displayed header section of the table properties.

Row Click Option	Selecting this checkbox provides the ability to make the table or grid act like a List view where a row can be selected and data from that row can be passed to another page, call another layout, perform an update and so on.
	After selecting this checkbox, the Change Type link will appear. Click the Change Type link the Select URL Type page is displayed. Select the appropriate option and complete the configuration.
	In order to disable the row click option deselect the checkbox or select the Reset Type to Text button.
	See Enabling Row Click on Table

Image: Table Properties page — Table Properties section

This example illustrates the fields and controls in the Table Properties section of the Table properties page. You can find definitions for the fields and controls later on this page.

Column Toggle	Select this property to hide less important columns and provide a button that opens a menu, allowing the user to choose to increase or decrease the number of viewable columns.
	The column chooser pop-up contains a dynamically generated list of columns based on the table markup that the user can select and deselect to adjust the visible columns.
	When you enable this property, use the Table Element Properties page to define a priority for each table element for columns to appear. This unique priority is used to generate a class name (ui-table-priority-<priority number selected>) which one can then modify via CSS. (Using the Table Element Properties page is discussed in the next section.)
Column Stripe	Select this property for the background to be shaded in alternate rows in the table.
Responsive	Select this property to add your own customizations via CSS3 and media queries.
Row Count	Select this property to include the number of rows in the table.
Column Button Theme	From the drop-down list select a jQuery theme to apply to the column pop-up button.
Column Popup Theme	From the drop-down list select a jQuery theme to apply to the pop-up column.
Column Button Text section	The default text that displays in the column pop-up button is *Columns*.
	Define a message from the message catalog to define a different label.

- *Message Set.* Select the message set of the message catalog entry that contains the label to use.

- *Msg Num.* Select the message number of the message catalog entry that contains the label to use.

- *Message.* Displays the text of the message catalog entry for the message set and message number defined.

Using the Table Element Properties Page

Use the Table Element Properties page to define properties for primitive elements/columns of the table.

To access the page:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

   The Layout Designer – Layout page appears.

2. In the Layout grid, locate the row that contains a primitive element for the table.

3. Click the Properties link.

Image: Table Element Properties page

This example illustrates the fields and controls on the Table Element Properties page. You can find definitions for the fields and controls later on this page.

Change Type	Click this link to change the element type for the primitive. See Changing Element Type
Hide column from table display	Select the property to not display the column in the table.
Table Column Priority	This property determines the order and priority of how the column displays.
	From the drop-down list, select a priority, *1* through *10,* with *1* defining the column as the first to appear in the table with the highest priority.
Table Class	Define the name of a CSS3 class to apply to the column.
Table Style	Define the name of a CSS3 style to apply to the column.

If you define a class or style name for a column, you must define the class or style in the Layout Designer – CSS page.

For information on the fields and controls on the Table Element Properties page not described in this topic, see Common Layout Page Element Properties.

Enabling Row Click on Table

You can make a table act like a List view where a row can be selected and data from that row can be passed to another page, call another layout, perform an update and so on.

To enable row click on a table or grid:

1. In the Layout grid, click on the Properties link for the table or grid element.

                             Table Element

2. Select the Row Click checkbox.

3. After selecting this checkbox, the Change Type link will appear. Click the Change Type link.

4. On the Select URL Type page, select the appropriate option and complete the configuration.

In order to disable the row click option deselect the checkbox or select the Reset Type to Text button.

Using the Select URL Page

Use the Select URL page (IB_DOCLOURL2_SEC) to select the URL type for the row click.

Navigation

Click the Change Type link on the Table Properties page.

Image: Select URL Type page

This example illustrates the fields and controls on the Select URL Type page. You can find definitions for the fields and controls later on this page.

URL	See <u>Defining Standard URLs</u>
Call REST Service	See <u>Defining URLs to REST Services</u>
Call Layout	See <u>Defining URLs to Other Layouts</u>

| **Layout Page** | See <u>Defining URLs to Pages Within Layouts</u> |
| **Update** | See <u>Defining URLs to Execute PeopleCode Update Events</u> |

Changing Element Type

On a table, you have the option to change the primitive type for Input, Image, URL, Checkbox Radio, HTML or Button.

The following restrictions apply:

• Only Text primitives can be changed to HTML.

• Only Boolean primitives can be changed to Check Box or Radio.

To change the primitive type:

1. Click on the Properties link for the element in the table.

2. Select the Change Type link.

3. The Select New Type page will appear with the available options.

4. Perform the same steps as you would when adding that specific primitive type element directly.

Using the Select New Type Page

Use the Select New Type page (IB_DOCTABTYPE_SEC) to select the new primitive type.

Navigation

Click the Change Type link on the Table Element Properties page.

Image: Select New Type page

This example illustrates the Select New Type page.
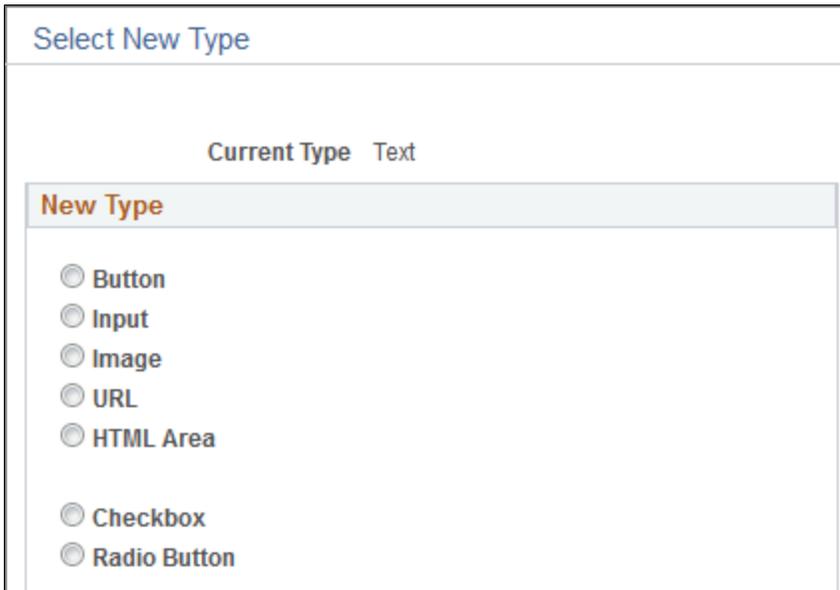
The new selection will appear on the properties page of the element. All types selected other than the default Text will allow the user to add Class, Style, Alt ID and Javascript. Label Text will be shown for Check Box, Radio and Input.

Image: Example Table Element Properties where the element type was changed to URL

This example illustrates the fields and controls on the Example Table Element Properties where the element type was changed to URL.
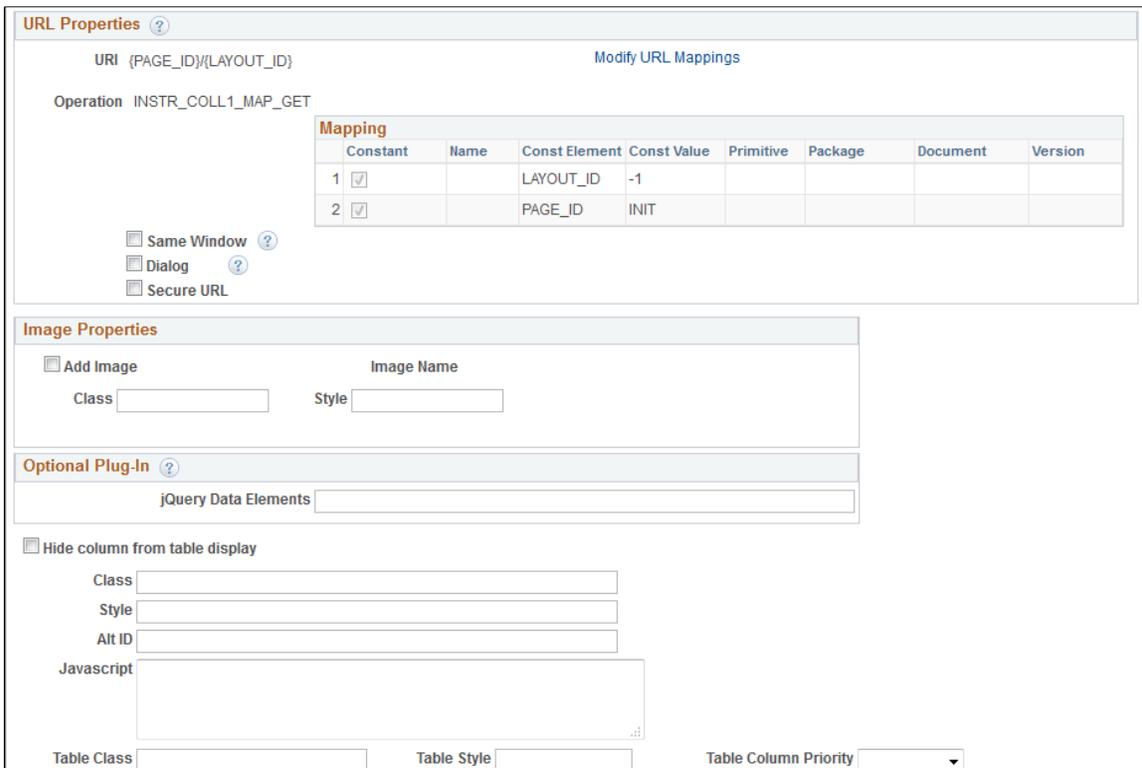
Resetting Type to Text

To reset the primitive to text:

1. Click on the Properties link for the element in the grid or table.

2. Click the Reset to Text button.

3. Click OK.

Image: Example Table Element Properties after changing type

This example illustrates the fields and controls on the Example Table Element Properties after changing type.

Chapter 46

Managing Sidebar Page Elements

Managing Sidebar Page Elements

This topic discusses how to manage sidebar page elements.

Understanding Defining and Developing Sidebar Page Elements

Sidebars are content areas that enter the viewport from the right or left side.

Similar to container elements, when you define a sidebar the framework inserts start and end element IDs into the Layout grid. You then define additional page elements inside those element IDs.

When working with sidebar elements consider the following guidelines:

- You can position and define sidebar elements only after the *mapfooter_end* element ID on a page.

- You can define multiple sidebar elements on a page, as long as they are all inserted after the *mapfooter_end* element ID.

- You can insert and define any page element in a sidebar element except for another sidebar element; you cannot nest sidebar elements.

- You must define a URL type push button or URL page element to open the sidebar.

- You can optionally define a URL type push button or standard URL page element to close the sidebar. If you do not define a close control, users can tap outside of the sidebar content area to close the sidebar.
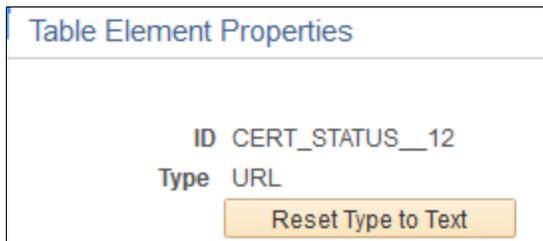
Defining Sidebar Page Elements

This section discusses how to:

- Add sidebar page elements to pages.

- View defined sidebar page elements in the Layout grid.

Understanding Defining Sidebar Page Elements

To define a sidebar element you use the Layout grid toolbar on the Layout grid on the Layout Designer – Layouts page to insert it into the layout page.

Note: Sidebar page elements can be defined only after the *mapfooter_end* element.

At the time you define/add a sidebar page element to your application, there are no other tasks to perform or parameters to specify. After you add the element to a layout page, use the Sidebar Properties page to define styles and options for the sidebar.

Adding Sidebar Page Elements to Pages

To add a sidebar page element to a page:

1. Access the Layout grid on the Layout Designer – Layout page.

   Select PeopleTools >Mobile Application Platform >Layout Designer. to access the page.

2. Click the *mapfooter_end* element ID in the Layout grid to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                       The Current Row button.

3. Click the Add Sidebar button on the Layout grid toolbar.

                                       The Add Sidebar button.

After you click the button the sidebar element appears in the Layout grid.

Viewing Defined Sidebar Page Elements in the Layout Grid

Sidebar page elements are indicated in the Layout grid by the *mapsidebar_start* and *mapsidebar_end* element IDs.

Image: Sidebar element defined in the Layout grid

This example illustrates a sidebar element defined in the Layout grid.

The label text for sidebar elements does not appear on the user interface at runtime, however you may find it useful to enter values in these fields for reference during development.

To define properties for a sidebar element click the Properties link to the right of the start element ID. Defining properties for sidebar elements is described in the next section.

Defining Sidebar Page Element Properties

Understanding Defining Sidebar Page Element Properties

Many properties that you can define for sidebar page elements are also available to define for other page elements. These common properties are described in the topic Common Layout Page Element Properties

This topic discusses using the properties that are unique to working with sidebar page elements.

Defining Sidebar Element Properties

Use the Sidebar Properties section of the Sidebar Properties page to define properties for this element.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer and in the Layout grid click the Properties link next to the *mapsidebar_start* element ID.

Image: Sidebar Properties section

This example illustrates the fields and controls in the Sidebar Properties section of the Sidebar Properties page. You can find definitions for the fields and controls later on this page.
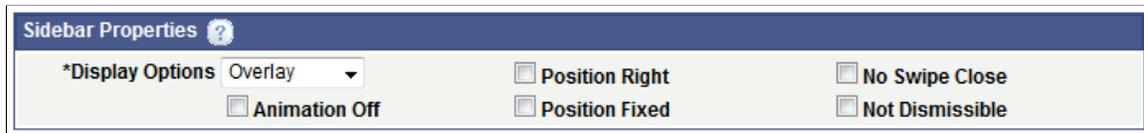
Display Options	From the drop-down list select how the sidebar appears on the page. The values are:

- *Reveal.* The sidebar sits under the page and is revealed as the page slides away.

- Overlay. (Default.) The sidebar appears on top of the page content.

- *Push.* Animates both the Sidebar and page at the same time.

Animation Off	Select the property to turn off animations.
Position Right	By default a sidebar appears on the left side of the viewport.
	Select this property to present the sidebar on the right-side of the viewport.
Position Fixed	Select this property so that sidebar contents appear no matter how far down the page the user scrolls.
No Swipe Close	By default swiping wiping left or right, or tapping the Esc key closes the sidebar.
	Select the property to turn off the swipe-to-close behavior.
Not Dismissible	By default, users can close a sidebar by selecting or tapping the page content outside the sidebar.
	Set this property to prevent this default behavior.

In addition, there is a Sidebar Responsive property that you can set on the Page Details page that sets a breakpoint in the sidebar widget to facilitate responsive behavior. See Using the Page Details Page

Developing Controls to Open and Close Sidebars at Runtime

You must create a button or URL for users to open a sidebar at runtime.

Users can select or tap an area outside of the sidebar content area to close a sidebar, or you can create a button or URL that closes it.

To create a stand-alone icon to open or close a sidebar, use the information in the following sections to create the push button or URL. When defining the control, choose an icon for the image and in the Layout grid leave the Label Text field blank for the element.

Creating Controls to Open Sidebar Content

To create a button or URL to open sidebar content:

1. Insert a push button or URL element somewhere on the page other than the footer or sidebar.

2. Define a standard URL for the element.

   When defining the URL to open the content, enter it in the following format:

   ```
   #mapsidebar_start_1
   ```

   Where *mapsidebar_start_1* is the starting element ID for the sidebar.

3. If you want to add a label to the button or URL, in the Layout grid enter a value in the Label Text field for the button or URL element .

Creating Controls to Close Sidebar Content

When you define the URL for the push button or URL element to close a sidebar, you define the page to appear when the user closes the sidebar.

After you create the desired page element, you must go to the associated properties page for the element and enable the Sidebar Close property.

Image: Sidebar Close property

This example illustrates the Sidebar Close property.

The property appears at the bottom of the Button Properties page and URL Properties page, only when you define one of these elements inside a sidebar element.

To create a button or URL to close sidebar content:

1. Insert a push button or URL element somewhere in the sidebar element.

2. Define a standard URL for the element.

   When defining the URL to close the content, specify the layout page to appear when the user closes the sidebar. Specify the URL in the following format:

   ```
   #page3
   ```

   Where *page3* is the name defined in the Page Name field on the Page Details page.

3. If you want to add a label to the button or URL, in the Layout grid enter a value in the Label Text field for the element.

4. Enable the Sidebar Close property.

   a. In the Layout grid, click the Properties link next to the button or URL element.

The properties page for the element appears.

b.   Select the Sidebar Close property at the bottom of the page.

c.   Click the Save button.

Related Links
Using the Page Details Page

Chapter 47

Defining Layout Page Navigation

Defining Page and Layout Navigation

This topic describes how to define page navigation in mobile applications you create using the Mobile Application Platform.

Understanding Defining Page and Layout Navigation

Typically, in a mobile application built on the Mobile Application Platform framework, users navigate to other pages within a layout, to pages in other layouts, and to other resources not defined as layouts, such as external web sites. In most cases, users click a button, image, hyperlink, or some other page element to navigate to a different page. And the navigation logic is defined in the properties page of the element.

Note: In the discussion about page navigation in this topic the terms *navigate, link* and *call* are synonymous

Calling Internal Page Types

Internal page types enable you to link to other pages in the same layout.

The system nests the internal page types within the same HTML document, and they are added as children to the body so that they can get accessed without any server call.

The advantage of using internal pages is that after the HTML document is loaded, no additional application server calls are necessary to navigate through the pages.

The disadvantage is that all the content data needs to be loaded as part of initialization. Moreover, if there are many pages the size of the HTML document can be quite large. If you think that some pages would not be frequently selected than one of the other options below might be more desirable.

Calling External Page Types

If you don't want to include a page inside the same HTML document as the first one (initial call), or if you need to dynamically create the content, then the best choice is defining navigation to external page types.

When navigating to external page types the framework makes the request, gets the external page, parses the content, adds the page to the current DOM, and makes a smooth transition to the new page. This is the typical selection for mobile applications.

For example, on a page you can call an external page and pass parameters as part of the request using layout initialization functionality.

If you have a list of items on a page and each item has a link to the same external page, the framework makes a call to the application server when the user selects an item, which in turn runs the initialization event. In the initialization event, the code checks the primitive(s) for each item and pulls the value(s) out. PeopleCode can then get information for the item(s) and populate the PeopleSoft Document object accordingly.

The framework builds the external page based on this data and the data presented back to the user is the information for the item(s). The user can select other items, which results in displaying the information for the item(s) on the external page.

Calling Other Mobile Application Platform Layouts

The process to navigate to other layouts is similar to calling external pages within layouts.

Add a link on the properties page for a URL or button element. And instead of calling an external page, you call another layout. Based on the information defined in the Layout Initialization Parameter page, you can select the URI to use to build the link.

Calling REST Services

This section describes calling REST services from Mobile Application Platform applications.

Use PeopleSoft Integration Broker to build and manage REST services. See *PeopleTools 8.55: PeopleSoft Integration Broker, "Managing REST Services"* for more information.

Understanding Calling REST Services from Mobile Application Platform Applications

You can call REST services (provider or consumer) defined in your application database. When defining the page element properties for a URL or button, you can select a provider or consumer REST service.

REST Provider Services

The REST service provider builds the REST service using the Integration Broker framework and specifies the service in the properties page for the page element. The REST service provider can generate the WADL to furnish to the service consumer for use in building the consumer service operation. Using the Provide Web Service wizard, select to generate the WADL for the system-generated service name associated with the requested layout. To access the Provide Web Service wizard, select PeopleTools >Integration Broker >Web Services >Provide Web Services.

Note that the system-generated REST service name displays on the Layout Designer – Security page.

REST Consumer Services

Since all Mobile Application Platform layouts are essentially REST services, you can request the WADL for a layout from the REST service provider, build the consumer REST service and the GET REST service operation based on the WADL. The GET REST service operation, must include a non-rowset based response message in html format. You can then call the service operation from the properties page for the page element.

Calling PeopleSoft Fluid and PeopleSoft Classic Applications

In situations where a PeopleSoft fluid page or a PeopleSoft classic application page calls a Mobile Application Platform application you can use the several PeopleCode properties to provide navigation back to the calling fluid or classic application page:

MAP Class Property	Description
PSBackURL	Use this property to obtain the URL that made the request (fluid, PeopleSoft classic, and Mobile Application Platform). Create a button and assign the URL as a button property to provide navigation back to the page.
PSBackLabel	Use this property to obtain the PeopleSoft component page name or MAP application/page that made the request and use the return value as a button label. Moreover, set this property to provide to the target system the proper label name to use for its back button to call the MAP application/page.
PSBackOrigin	Use this property to obtain the origin of the calling application, Mobile Application Platform, fluid or classic PIA.
PIAOriginURL	Use this property to obtain the last fully qualified PIA URL from the requesting page.
PIAOriginlabel	Use this property to obtain the label associated with the PIA URL.

Calling Other Documents and Sites

To call other documents and sites not defined as layouts, add the fully-qualified standard URL on the properties page for the element.

Configuring Related Actions

Related actions can be added to Static Text, Input, and Listview elements. Defaults for the Related Action Class, Style, Popup Class and Image are defined on the Default Layout Options page. See Defining Default Layout Options.

Setting Related Action Defaults in the Layout Designer

Use the Related Actions page to override the default Related Action Class, Style, Popup Class and Image.

1. Open the Layout where you are adding the related action. (select PeopleTools, Mobile Application Platform, Layout Designer)

2. Expand the Layout Details section.

3. Click the Related Actions link.

Image: Related Actions page

This example illustrates the fields and controls on the Related Actions page. You can find definitions for the fields and controls later on this page.

Class	Enter the class name to apply to the Related Action.
Style	Enter the style name to apply to the Related Action.
Popup Class	Enter the popup class name to apply to the Related Action.
Image	Image used to denote a related action for an element.
Select an image	Opens the Image Selector page to select an image from the database. The default image PTS_ACTIONS is already registered under People Tools for Dynamic Image Registration.

Note: You can also override any of properties using People Code. Properties of the MAP Object are RAClass, RAPopupClass, RAStyle, and RAImage.

Adding Related Actions to MAP Page

To add related actions to a MAP page:

1. Select PeopleTools, Portal, Manage Related Content Services.

2. Select the MAP layouts tab.

3. Select the Assign Related Actions to a MAP Layout link.

4. Select the Layout and click Add.

5. Click the Select icon for the page and select the page where the Related Action will appear.

6. Click the Select icon for the Service ID and select the service.

7. Select the service target location.

8. Click the Configure button.

9. Select the Mapping type for each parameter displayed.

10. Click the Select icon to select the Mapping Data.

11. Click the Select button in the menu Options section and select the Page Field menu.

12. Click OK.

13. Click Save.

See "Managing Related Actions for MAP Layouts" (PeopleTools 8.57: Portal Technology)

Chapter 48

Invoking Mobile Application Platform Layouts

Prerequisites for Invoking Layouts

Before you can invoke a Mobile Application Platform layout you must complete the tasks described in the topic <u>Prerequisites for Using the Layout Designer</u> which are to configure the following components in PeopleSoft Integration Broker:

- Define the Web service and REST target locations.

- Configure the Integration gateway, including set the gateway URL, load target connectors, define the default local node on the PeopleSoft Node Configuration page, and define the Java keystore password.

In addition, you must set permission list security for the layout using the Layout Designer – Security page.

Invoking Layouts from the Layout Designer

To invoke a layout from the layout designer:

1. Select PeopleTools >Mobile Application Platform >Layout Designer.

2. Click the View Initialization URL link.

3. Copy and paste the link into the URL field of a browser.

4. Hit the Enter key.

353

Chapter 49

Developing PeopleCode for the Mobile Application Platform

Understanding PeopleCode for the Mobile Application Platform

The Mobile Application Platform framework completely separates the business logic from the presentation layer.

Application classes are the primary methodology to use to add business logic and transaction data to a mobile application and to dynamically change the layout.

All metadata is available in the application class PeopleCode layer and is not tied to any mobile application page or other metadata, allowing you to override the defaults defined on layouts.

PeopleSoft delivers several objects and events for use in the application classes you build.

Use PeopleSoft Application Designer to create application classes for a Mobile Application Platform application.

This topic provides overview information and several examples of Mobile Application Platform PeopleCode. See the *PeopleTools 8.55: PeopleCode API Reference, "MAP Classes"* for detailed information about the MAP API.

Mobile Application Platform Application Class Interface

All Mobile Application Platform application classes must be derived from the IDoclayoutHandler interface, located in the PS_PT application package in PeopleSoft Application Designer.

Mobile Application Platform Objects

Three objects are delivered for the Mobile Application Platform framework:

Map	High-level object as input for the MAP PeopleCode events. The Document object, which contains the business logic shape definition and the URI document can be accessed from this object.
MapPage	Page object which can be retrieved from the Map object.
MapElement	Element object defined on the page. The MapElement object can be retrieved via the Map object or MapPage object.

In addition, the IntBroker object features several methods and properties to support the framework.

Mobile Application Platform Events

The four events currently supported by the framework are described in the following table:

Event	*Event Name*	*Description*
Initialization	OnInitEvent	This event populates the application with data when the application or external page is initialized. Specify this event in the Base Event Method section of the Initialization page. (PeopleTools, Mobile Application Platform, Layout Designer and click the Initialization link.)
Update	OnUpdateEvent	You can use the Update event and pass optional parameters to update the page The Update event is invoked every time an element is selected that has an Update event declared via the Get HTTP method. The current elements that can have an Update event are: URL, List view and Button. Specify this event in the properties page for an element.
Ajax	NA	You can invoke an Ajax event on any page element and you specify the even on the properties page for an element. In the element properties you provide the JavaScript event and function, and the application class to invoke. Use the Layout Designer – JavaScript page to code the JavaScript. See the Using Ajax Events in the Mobile Application Platform.
Save	OnSaveEvent.	You can invoke the Save event on the Button element. The Save event gets fired every time the application is invoked via the POST HTTP method when the Save option is selected. Specify this event in the properties page for the Button element.

Examples: Mobile Application Platform PeopleCode

This topic contains examples that illustrate using PeopleCode in the Mobile Application Platform.

Some of the example reference transactional data and application metadata.

Transactional Data	Data passed from the PeopleSoft document and can include things like customer ID, customer name, purchase order number, and so on.
Application Metadata	Metadata that determines how things are going to be rendered in the application – essentially layout metadata, like classes,

page elements, and so on. Application metadata also defines the
behavior of page controls.

Mobile Application Platform Application Class

This example shows a sample Mobile Application Platform application class.

In the example you'll notice that the Map object does not get instantiated. Instead, the Map object is built
based on the layout metadata, which is the layout itself.

The layout in this example is FlightCard. So the framework creates the Map object to pass into the event,
and if the event method returns a Map object, then it simply returns it.

```
import PS_PT:Integration:IDocLayoutHandler;

class FlightCard implements PS_PT:Integration:IDocLayoutHandler
 method FlightCard();
 method OnInitEvent(&Map As Map) Returns Map;
 method OnUpdateEvent(&Map As Map) Returns Map;
 method OnSaveEvent(&Map As Map) Returns string;
 property integer SaveStatus;
end-class

/* constructor */
method FlightCard
end-method;

method OnInitEvent
 Local Compound &COM;
 Local Document &Document;

 &Document = &Map.GetDocument();
 &COM = &Document.DocumentElement;

 /* run business logic and return Map object */
 Return &Map;
end-method;

method OnUpdateEvent
 Local Compound &COM;
 Local Document &Document;

 &Document = &Map.GetDocument();
 &COM = &Document.DocumentElement;

 /* run business logic and return Map object */
 Return &Map;
end-method;

method OnSaveEvent
 Local Compound &COM;
 Local Document &Document;

 &Document = &Map.GetDocument();
 &COM = &Document.DocumentElement;

 /* run business logic */
 Return "Success";
end-method;
```

Get Document Object to Read/Write Transactional Data

```
Local Compound &COM;
Local Document &Document;
```

```
&Document = &Map.GetDocument();
&COM = &Document.DocumentElement;
&COM.GetPropertyByName("Pilot").Value = "data";
```

Get MapPage Object and Read/Write Application Metadata

```
Local Compound &COM;
Local Document &Document;
Local MapPage &Page;
Local String &ClassName;
&Document = &Map.GetDocument();
&COM = &Document.DocumentElement;
&COM.GetPropertyByName("Pilot").Value = "data";

&Page = &Map.GetPage(1);

/* Read and write out Main Class Name */
&ClassName = &Page.MainClassName;
&Page.MainClassName = "input-wrapper";
```

Get MapElement Object and Read/Write Application Metadata

```
Local Compound &COM;
Local Document &Document;
Local MapPage &Page;
Local MapElement &Element;
Local string &ClassName, &LabelText;

&Document = &Map.GetDocument();
&COM = &Document.DocumentElement;
&COM.GetPropertyByName("Pilot").Value = "data";

&Page = &Map.GetPage(1);

/* Read and write out Main Class Name */
&ClassName = &Page.MainClassName;
&Page.MainClassName = "input-wrapper";

/* MapElement object via MapPage object */
&Element = &Page.GetElement("mapbutton_1");
&LabelText = &Element.LabelText;
&Element.LabelText = "Update";

/* MapElement object via Map object */
&Element = &Map.GetElement("mapbutton_1");
&LabelText = &Element.LabelText;
&Element.LabelText = "Update";
```

Get MapElement Object and Read/Write Application Metadata

```
Local Compound &COM;
Local Document &Document;
Local MapPage &Page;
Local MapElement &Element;
Local string &ClassName, &LabelText;
Local integer &Control;

&Document = &Map.GetDocument();
&COM = &Document.DocumentElement;
&COM.GetPropertyByName("Pilot").Value = "data";

&Page = &Map.GetPage(1);
```

```
                /* MapElement object via MapPage object */
                &Element = &Page.GetElement("mapbutton_1");
                &LabelText = &Element.LabelText;
                &Element.LabelText = "Update";

                /* disable Button */
                &Control = &Element.Controls;
                &Element.Controls = %MAP_Disable;
```

Change JavaScript Event and Function Call for a Mobile Application Page Element

```
                Local Compound &COM;
                Local Document &Document;
                Local MapPage &Page;
                Local MapElement &Element;
                Local string &ClassName, &LabelText;
                Local integer &Control;

                &Document = &Map.GetDocument();
                &COM = &Document.DocumentElement;
                &COM.GetPropertyByName("Pilot").Value = "data";

                &Page = &Map.GetPage(1);

                /* Change JavaScript Event and Function*/
                &Element = &Page.GetElement("mapbutton_1");
                &Element.SetJavaScript("onclick", "changedata");
```

Send Back JSON Data for Ajax Call from Different Document

```
                Local Compound &COM, &URICOM;
                Local boolean &bRet;
                Local Document &Document, &URIDocument, &Doc

                &Document = &Map.GetDocument();
                &COM = &Document.DocumentElement;

                &URIDocument = &Map.GetURIDocument();
                &URICOM = &URIDocument.DocumentElement;

                &Doc = CreateDocument("MAP_LAYOUT", "FLIGHTCARD1_MAP", "v1");
                &COM = &Doc.DocumentElement;
                &COM.GetPropertyByName("PAGE_ID").Value = "INIT";
                &COM.GetPropertyByName("LAYOUT_ID").Value = - 1;

                &bRet = &Map.SetOvrdAJAXDocument(&Doc);

                Return &Map;
```

Get Generated HTML for a Container in a Layout

```
                &Doc = CreateDocument("MAP_LAYOUT", "FLIGHTCARD1_MAP", "v1");
                &COM = &Doc.DocumentElement;
                &COM.GetPropertyByName("PAGE_ID").Value = "INIT";
                &COM.GetPropertyByName("LAYOUT_ID").Value = - 1;
                &string = %IntBroker.GenLayoutHTML("FLIGHTCARD", 1, &Doc, "mapcont_start_2");
```

Chapter 50

Managing Layout Cascading Style Sheets

Understanding Managing Layout Cascading Style Sheets

Cascading style sheets (CSS) enable you to programmatically describe how application are to appear and display when rendered in HTML and XML. And through the use of media queries, you can influence how they appear on different devices.

Use the Layout Designer - CSS page (IB_DOCLOCSS) to create CSS3-compliant style sheets, media queries and themes in your application. Note that you are responsible for the correctness and completeness of your code, as the system does not perform any validation on cascading style sheet, media query or theme code you create and define on this page.

If you link one or more templates to the pages of a layout, the system-generated cascading style sheet(s) appear on the page in read-only format. To make changes to cascading style sheets defined in a template and assigned to a page or layout, you must open the layout template, and make the necessary changes on the Layout Designer – CSS page. Then on the Layout-Designer-Layout page click the Sync Element Types button to synchronize the template modifications with the layout.

The Mobile Application Platform framework applies cascading style sheets following a specific order of precedence. It's important to familiarize yourself with the order of precedence to ensure that the correct styles are applied to your mobile application pages.

Related Links
Applying and Synchronizing Templates in Layouts
Cascading Style Sheets (CSS)

Using the Layout Designer - CSS Page

To access the CSS page (IB_DOCLOCSS) select PeopleTools >Mobile Application Platform >Layout Designer and click the CSS tab.

Image: Layout Designer – CSS page

This example illustrates the fields and controls on the Layout Designer – CSS page. You can find definitions for the fields and controls later on this page.
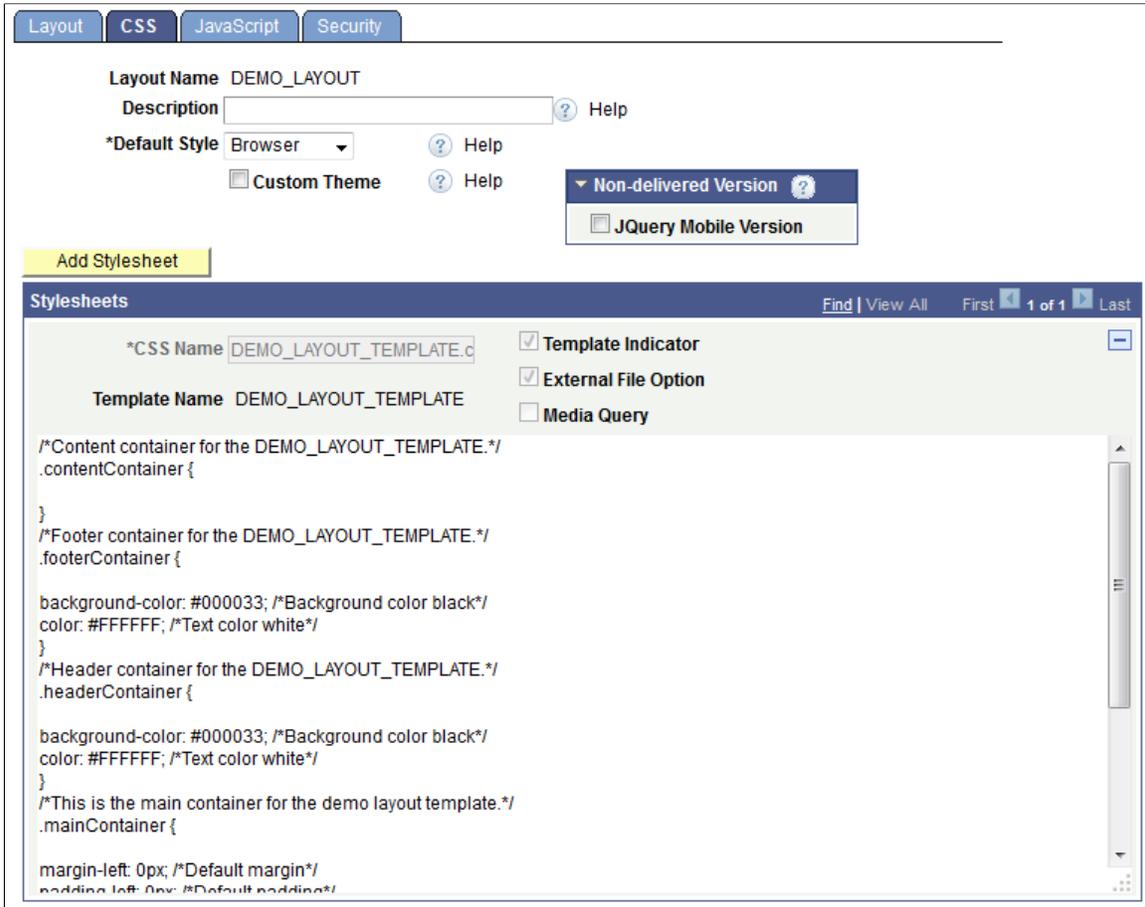
This example shows a system-generated style sheet for the *DEMO_LAYOUT_TEMPLATE* in the Layout Designer – CSS page.

The following example shows the page when you add a new style sheet to a layout:

Image: Layout Designer – CSS page

This example illustrates the fields and controls on the Layout Designer – CSS page when add a new cascading style sheet to a layout. You can find definitions for the fields and controls later on this page.

Layout Name	Displays the name of the layout.
Description	Displays the description of the layout if defined.
Default Style	Set the HTML default style rendering. The valid values are:

- *Browser.* (Default.) Use default browser styles.

- *Reset .* Set the default browser element styles to 0 (zero).

- *Normalize.* Adjust the default styles so that they look very similar across browsers.

Custom Theme	Check the box to apply a custom jQuery Mobile theme.
Custom Swatches	This field appears only if you check the Custom Theme box.
	Enter one or more swatch values. Enter multiple values separated by spaces.
JQuery Mobile Version	To view this field, expand the Non-delivered Version section.

	Check the box to use CSS that adheres to a non-supported version of jQuery Mobile.
Add Stylesheet	Click the link to add a new page for enter a cascading style sheet.
CSS Name	Enter the name for the style sheet, including the .css extension.
Template Name	If the style sheet displaying in on the page is system-generated from a layout template, this field displays the template name on which the style sheet is based.
Template Indicator	When checked, this read-only field indicates that the source of the CSS is from a layout template.
External File Option	Select the box to save a copy of the style sheet on the web server at the following location:

```
<PIA_HOME>\webserv\<PIA_DOMAIN>\applications\peoplesoft⇒
\PSIGW.war\map\<NODE_NAME>\layouts\<LAYOUT_NAME>\css
```

Media Query	Select the box to add a media query to the cascading style sheet.
Add/View Media Query	This link appears only when you select the Media Query box.
	Click the link to access the Layout Media Query page to add or a view a media query to the layout.
	Using the Layout Media Query page to add and view media queries is discussed elsewhere in the documentation for this subject.
Stylesheets	Use the long edit box to write the CSS.

Note: The system does not perform any validation checks on the CSS code.

Chapter 51

Managing Layout JavaScript

Understanding Managing Layout JavaScript

You can invoke JavaScript events and jQuery Mobile events on many page elements in the Mobile Application Platform. In most cases, you define the JavaScript on the JavaScript page and then reference the JavaScript event and function on specific page element properties.

If you have assigned one or more templates to a layout, any JavaScript that you included in the templates appears on the JavaScript page in read-only format. To make changes to JavaScript defined in a template, you must open the template, make the necessary changes, and then use the Sync Element Types button on the Layout page to synchronize the modifications with the layout.

Related Links
Applying and Synchronizing Templates in Layouts

Using the Layout Designer - JavaScript Page

To access the JavaScript page (IB_DOCLOJS) select PeopleTools >Mobile Application Platform >Layout Designer and click the JavaScript tab.

Image: Layout Designer – JavaScript page

This example illustrates the fields and controls on the Layout Designer – JavaScript page. You can find definitions for the fields and controls later on this page.
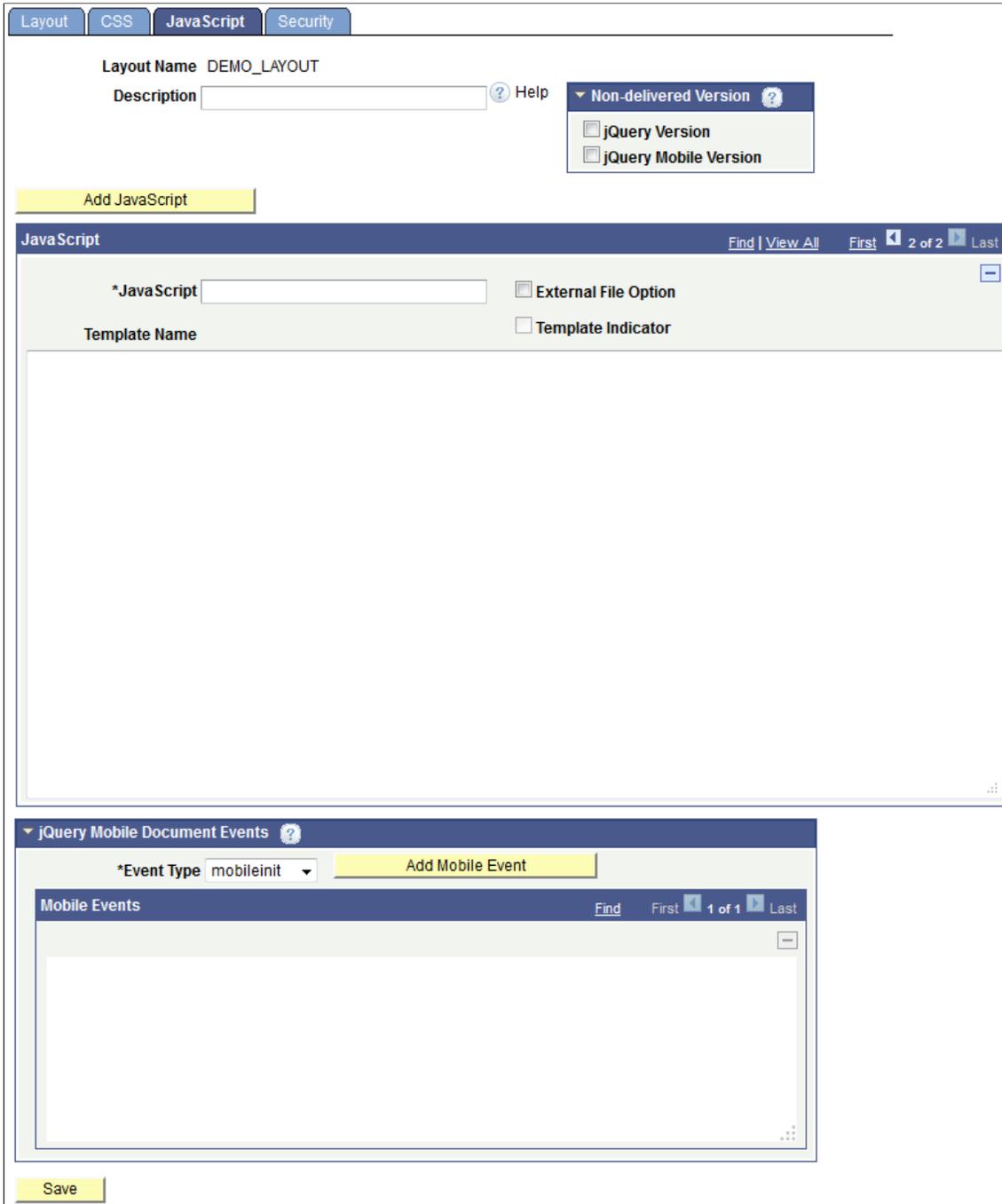
Layout Name	Displays the name of the layout.
Description	Displays the description of the layout.
jQuery Version	To view this field, expand the Non-delivered Version section.
	Check the box to use jQuery code that adheres to a non-supported version of jQuery.

jQuery Mobile Version	To view this field, expand the Non-delivered Version section.
	Check the box to use jQuery Mobile code that adheres to a non-supported version of jQuery Mobile.
Add JavaScript	Click the button to add a new page for creating JavaScript.
JavaScript Name	Enter a name for the JavaScript file, including the .js extension.
Template Name	If the JavaScript displaying on the page being read from a layout template, this field displays the template name where the JavaScript is defined.
Template Indicator	When checked, this read-only field indicates that the source of the JavaScript is from a layout template.
External File Option	Check the box to save a copy of the JavaScript on the web server at the following location:

```
<PIA_HOME>\webserv\<PIA_DOMAIN>\applications\peoplesoft⇒
\PSIGW.war\map\<NODE_NAME>\
layouts\<LAYOUT_NAME>\css
```

JavaScript	Use the long edit box to code the JavaScript. Note that the system does not perform any validation checks on the JavaScript code.
Event Type	Select a jQuery Mobile event from the drop-down list. The options are:

- *mobileinit.* (Default.)

- *pageinit.*

- *ready.*

Add Mobile Event	After you select a jQuery Mobile event type, click the button to add the event.
Mobile Events	Use the long edit box to code the jQuery Mobile event. The framework automatically defines the method syntax and places the method within the HTML markup. You need only add the contents of the appropriate event.

Note: No validation is performed on the code you enter.

Save	Click the button to save the additions and changes to this page.

Chapter 52

Debugging in the Mobile Application Platform

Debugging Mobile Application Platform Applications

This topic describes using the Layout Status page to set system settings that impact debugging Mobile Application Platform applications.

Understanding Debugging Mobile Application Platform Applications

In the Mobile Application Platform there is a system status setting that you define at the layout level that impacts the information detail that the system displays if an error occurs. The setting, System Status, is located on the Layout Status – Layout Status page.

Note: To debug Mobile Application Platform applications make sure that the system status for the layout is set to *Development*.

The following table describes the setting values:

Development	If an error occurs in *Development* mode:
	• The error is written to the application server log.
	• The error is displayed in the browser window.
	• Any JavaScript files do not use the minimized (compressed) version to facilitate debugging in the browser.
	The following example shows a sample error when the system status is set to *Development* :

```
The length of value: 114 is larger than what is defined⇒
 for
the document primitive: Tgt_ac_ppfor value: 114. (2,101⇒
0)
QE_FLIGHTPLAN_SYNC.FlightData.OnExecute Name:OnInitEven⇒
t
PCPC:10446 Statement:70
```

Production	If an error occurs in *Production* mode:
	• The error is written to the application server log.
	• The browser returns the HTTP status code, with no error message, except for login failures which display the proper exception.

The following example shows a sample error when the system status is set to *Production:*

```
500 Internal Server Error
The server encountered an unexpected condition which
prevented it from fulfilling the request.
```

Note that in either mode you can view the HTML generated for the browser by setting the gateway logging to *5* or by viewing the page source in the browser.

System status is a runtime setting and has no impact on the ability to delete layouts, rename layouts, and so on.

Changing from one mode or the other takes immediate effect upon Save on the Layout or going through the application.

Using the Layout Status Page

Use the Layout Status page (IB_DOCLOSTAT) to set the system status for a layout.

To access the page select PeopleTools >Mobile Application Platform >MAP Administration >Layout Status.

Image: Layout Status page

This example illustrates the fields and controls on the Layout Status page. You can find definitions for the fields and controls later on this page.
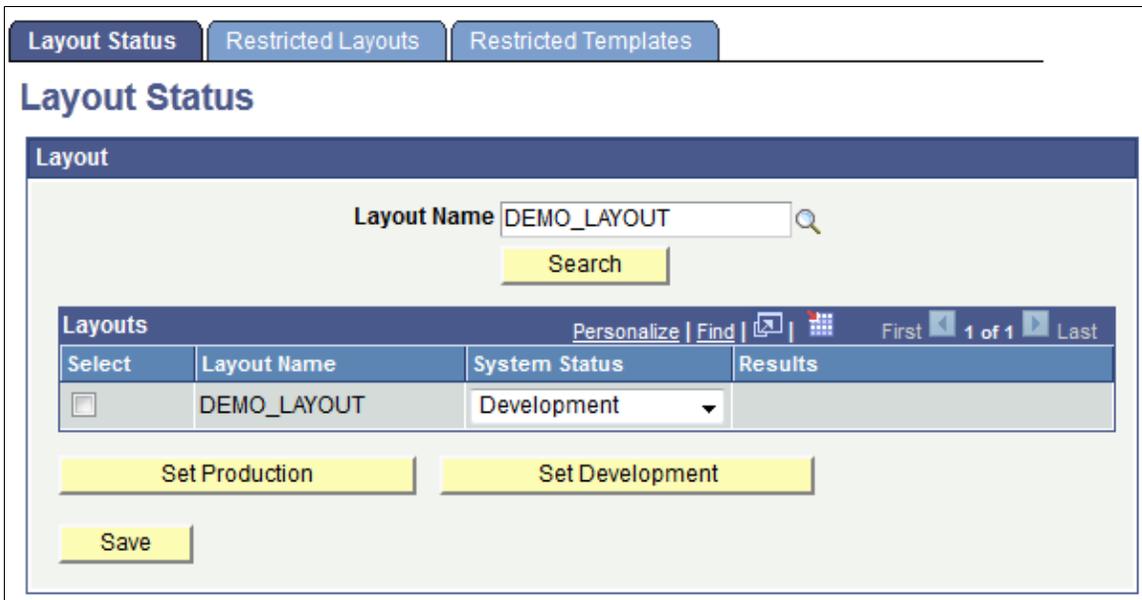
Layout Name	Enter the full or partial name of the layout for which to set the system status.
Search	Click the button to search the database for a layout based on the information entered in the Layout Name field.
	The search results appear in the Layouts grid.

Select   Select the field to select a layout for which to set the system
                                    status.

System Status   From the drop-down list select a status. The values are:

  • *Development.* (Default.)

  • *Production.*

Set Production   Click the button set the status of the selected layouts to
                                    Production.

Set Development   Click the button set the status of the selected layouts to
                                    Development.

Save   Click the button to save the changes.

Chapter 53

Managing Layout Security

Securing Layouts

Use the Security page (IB_DOCLOSECURE) to set and define security-related options for a layout, including authentication, grant permission-list access to a layout, and setting the system status for a layout.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer and click the Security tab.

Image: Layout Designer – Security page

This example illustrates the fields and controls on the Layout Designer – Security page. You can find definitions for the fields and controls later on this page.

Layout Name	Displays the name of the layout.
Description	Displays the description of the layout.
Service	Displays the system-generated REST service for the layout.
Req Verification	Use the drop-down list to set the required security level for requests. The options are:

- *Basic Authentication.*

- *Basic Authentication and SSL.*

- *None.* (Default.)

- *PeopleSoft Token.*

- *PeopleSoft Token and SSL.*

• SSL.

System Status

Use the system status level for debugging applications. The options are:

• *Development.*(Default.)

The system displays any errors in the browser and writes the errors to the application server log. Any JavaScript files will not use the minimized (compressed) version for potential debugging on the browser.

• *Production.*

If any error occurs the browser displays the appropriate HTTP status code only, with no error messages. The exception is in the case of login failures, in which case the system displays the exception. In this mode the system writes the errors to the application server log.

See Debugging Mobile Application Platform Applications for more information about debugging Mobile Application Platform applications.

Layout Permission

Click the link to access the Layout Permissions to page to grant permission-list access to the layout.

Chapter 54

Defining Custom Layouts

Understanding Custom Layout

The custom layout feature in MAP provides the ability to customize a delivered MAP layout. Typically when a PeopleSoft delivered application is customized, these customizations will present issues when the customer performs an upgrade or patch as the customized application will be overridden. Customers would need to run compare reports and determine from the compare reports how to merge the data. The MAP data model is designed to handle these types of merges as the customer can make simple to very complex customizations which as a result of a PeopleSoft update or patch will not override any of the customized metadata.

Creating a Custom Layout

To create a custom layout:

1. Select PeopleTools, Mobile Application Platform, Layout Designer

2. Select Add a New Value.

3. Enter the layout name, custom layout checkbox and the parent layout name.

4. Click Add.

5. (Optional) Add an optional document.

6. Click Create Custom layout.

7. You will be prompted to save the layout.

8. Use the Custom Layout to customize the delivered MAP layout.

Adding a New Custom Layout

Use the Add Layout page (IB_DOCLO_ADD) to add a new layout definition to the system.

To access the page select PeopleTools >Mobile Application Platform >Layout Designer. Click the Add a New Value tab. Select the Custom Layout checkbox.

Image: Add New Layout for Custom Layout

This example illustrates the fields and controls on the Add Layout page. You can find definitions for the fields and controls later on this page.

Layout Name	Layout Name
Custom Layout	Selecting this checkbox indicates that this will be a custom layout and therefore displays the Parent Layout Name prompt edit field.
Parent Layout Name	All layouts that can be customized will show in the prompt list. Select the layout to customize.

Excluded from the list are:

•   Layouts that have been assigned to a Custom layout.

•   Layouts built using Rapid App Designer.

After clicking Add, the page to select an optional document to link to the custom layout is displayed.

Image: Select an optional document to link to the custom layout.

This example illustrates the fields and controls on the Select an optional document to link to the custom layout page.

The user can assign an additional Document to this Customized Layout. The Document assigned to the Parent Layout Name should not be modified. Modifications needed to perform the customization should be assigned to this Document.

Note: This assignment of a Document is optional and thus is not required in order to create a Customized Layout.

Using the Custom Layout

The Custom Layout displays the actual assigned Parent Document. The number of pages will correspond to the number of pages of that on the Parent layout.

Image: Custom Layout page

This example illustrates the fields and controls on the Custom Layout page. You can find definitions for the fields and controls later on this page.
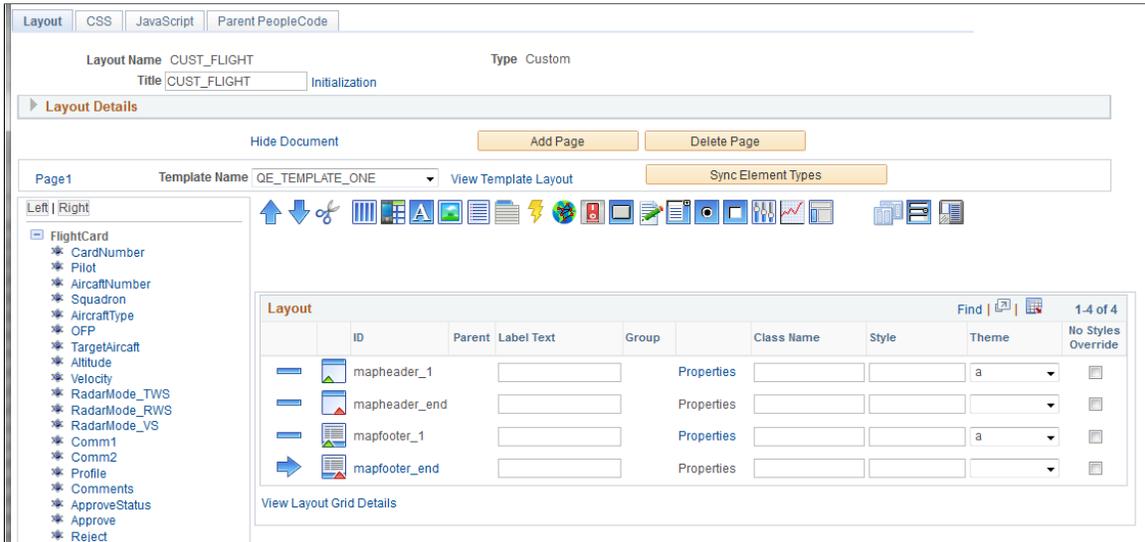
Note the following on this page:

- The Type is defined as Custom.

- The customized Document assigned (if assigned) will be below the Parent Document.

- There is no *View Initialization URL*. The Parent URL needs to be invoked to kick off the application.

- There is no Security page. Security is set from the Parent.

- Parent name is displayed to indicate the Layout associated with this Custom Layout.

Page Details will be read only for any page that was created upon initialization of the Layout, as these pages correspond to the pages from the Parent.

The user can add CSS/ Java Script to add/override styles or Java Script methods.

Layout Details

Expand the Layout Details section.

Image: Layout Details section

This example illustrates the fields and controls on the Layout Details section. You can find definitions for the fields and controls later on this page.

Package	Displays the package name that contains the PeopleSoft document linked to the layout.
Document	Displays the name of the PeopleSoft document linked to the layout.
Version	Displays the version of the PeopleSoft document linked to the layout.
Description	Enter a description of the layout.
Details	Enter a more lengthy description of the layout.
Service	Displays the name of the system-generated REST service for the layout.
Related Actions	Defaults from the parent layout.
Group	Enter a group name to group layouts together. Layout search pages can search by group name to show a logical grouping of layouts.
Owner ID	From the drop-down list, select an owner for the layout.
Validation	Defaults from the parent layout.
Raphael	Defaults from the parent layout.
Document Dump	Defaults from the parent layout.
Bookmark	Defaults from the parent layout.
Retain State	Defaults from the parent layout.
Touch Image	If you are implementing an application for the Apple framework, select an image for the tile.
Exception Layout Name	Defaults from the parent layout.
Parent Layout Name	Name of the parent layout.
No Custom	Select this checkbox if you do not want to use any of these customizations in order to generate the Parent application.

This is useful when a change occurs to the Parent (for example applying an update) and you want to see the modifications made to the parent.

Initialization

The Initialization link will contain all the Layout Initialization information from the Parent. The user can add or modify any of these links for the purpose of adding additional pages to the application. The user can choose to add a Base Event Method if needed for these additional pages.

The Initialization parameters are copied over from the Parent. The developer can add additional parameters as the Document URI is built from what is defined on the Custom Layout. Events can be defined to override the Parent on the Parent PeopleCode page, the Document URI will be that of the Parent. The Custom Document URI is used where any additional Page(s) created on the Custom Layout are called from the appropriate element such as URL, Button, Listview , etc.

AJAX Event:

If a Custom Document was defined then if XML or JSON is selected for data return a checkbox called *Use Custom Document* will be displayed. This allows the return data to be either from the Parent or Custom defined Document.

Parent PeopleCode page

All People Code events defined on the Parent will be displayed on this page. The user can elect to add a Custom event to any or all these events. At runtime the Parent event will first get executed, the returned MAP object will then be the input MAP object to the subsequent defined Custom event. The user can then modify anything on the MAP object (document data, element styles etc.). This will be all wrapped in the same transactional boundary so any database updates can also be overridden.

Image: Parent PeopleCode page

This example illustrates the fields and controls on the Parent PeopleCode page. You can find definitions for the fields and controls later on this page.

The same edits apply based on Event Type. Therefore the Application Class selected must be implemented by the proper interface IDocLayoutHandler in order for the Class ID to be available in the selection prompt. The Method(s) available in the prompt would then correspond to the proper Event Type signature: method xxxx(&Map As Map) Returns string.

If the developer wishes to only invoke the Custom defined event (instead of the Parent event first being executed), the Custom Event Only check box should be selected. At runtime only the Custom event will be executed.

Note: Adding additional Events is implemented the same way as a typical Layout. For example if an additional page was added to the Custom Document, then an OnInit method can be defined on the Base Event Method by selecting the Initialization link. In the case of OnInit Event Type the following execution order of methods would take place: Parent OnInitEvent, Custom OnInitEvent as defined on the Parent People Code page, and finally the OnInitEvent defined on the Custom Base Event Method. If additional AJAX events are required for the customization, these can be added under the applicable element's property page defined on the Custom Layout.

Customizing the Layout

Copying Containers from Parent

For customizations where the actual elements defined on a page need to be modified, the user can copy over the container and make the changes. At runtime the Container defined on the Custom Layout would be used instead of what is defined on the Parent.

Image: Select a page/container to copy

This example illustrates the fields and controls on the Parent PeopleCode page.

For Custom layouts , Parent Layout is available.

Depending on the page shown and where the cursor is placed at the time of selection, the dropdown list will show all containers that can copied over to the associated Custom Page. Therefore, if the cursor was defined within the Header Section then only Containers defined on that page's header section will be available to be copied.

Note: You cannot select a Container for customization if it contains a Save or Update Event. These events should be in their own Container or outside a Container. If a Save or Update is needed within a Container then an AJAX event should be used instead. Customizations for *Update* or *Save* events can still occur via People Code.

In Container Properties, under Optional Plugins, there is a property Use Parent.

Image: Container Properties – Optional Plug-In

This example illustrates the fields and controls on the Container Properties – Optional Plug-In page.
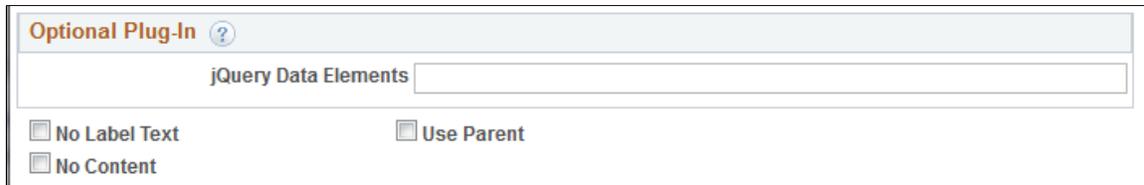
Select the Use Parent checkbox to use the Parent Container (and its elements defined within) instead of what is defined on the Custom Container. This property can be also set via People Code for the Custom defined Container.

Adding Elements

All elements are except UPDATE or SAVE events can be added to a Custom Layout. The user can always add an AJAX event.

Additional Edits

You cannot delete a page on the Custom Layout if it was generated at intialization (associated Parent page). If the Parent page is subsequently deleted then the user can delete the associated Custom page

Validation occurs when the Custom Layout is first selected in the Layout Desgner. Validation will check for associated pages with the Parent, document references, initialization parameters from Parent and so forth.

Containers defined on Sub Page cannot be deleted if referenced on a Custom Sub Container. If a container copied from the parent contains sub containers then these containers will also be copied over to the correct Sub Page.

Chapter 55

Administering Mobile Application Platform Applications

Defining Default Layout Options

This topic describes using the Layout Default Options page to define defaults for Mobile Application Platform layouts.

Using the Layout Default Options Page

To access the Layout Default Options page (IB_DOCLODFLTS) select PeopleTools >Mobile Application Platform >MAP Administration >Default Layout Properties.

Image: Layout Default Options page

This example illustrates the fields and controls on the Layout Default Options page. You can find definitions for the fields and controls later on this page.

Template Name	Enter or use the Lookup button to search for a layout template to apply by default to the first page in the layout.
Content Theme	From the drop-down list, select the default jQuery theme to use for content if the selected template does not define one.
Add Default Label Text	Select Header and/or Footer to add default text in the Header/Footer as a reminder to populate with the appropriate text.
Header Footer Positioning	Select the default position for the header and footer toolbars. The values are:

	• *Fixed.* The header and footer will appear within the viewport and stay present as the user scrolls. Tapping on the screen will hide them.
	• *Full Screen.* The header and footer will appear within the viewport and stay present as the user scrolls, regardless of where the user is in the content.
	• *Standard.* The toolbars are presented according to the document flow, scrolling into and out of the viewport as the user scrolls through the data.
Button	The default position of buttons when assigned to a group. The values are:
	• *Horz.* Horizontal.
	• *Vert.* Vertical.
Transition	Define the transition from this page to other pages in the layout. The options are:
	• *Fade.* Fade the page or dialog in over the previous content.
	• *Flip.* An animated page flip, rotating the current view out with the other view on the reverse side.
	• *Flow.* Throws the current page away and comes in with the next page.
	• *Pop.* The page springs into view from the center of the screen.
	• *Slide.* Slide in from the left or right, pushing the previous content out of the way.
	• *Slidedown.* Slide down from the top, over the top of the current container.
	• *Slidefade.* Slide in from the left or right then fade the page.
	• *Slideup.* Slide up to the top, revealing the next content below.
	• *Turn.* The page turns as in turning a page in a book.
Event PeopleCode Package Name	This field is reserved for future use.

Deleting Layouts

This topic describes using the Delete Layouts page to delete Mobile Application Platform layouts.

Using the Delete Layouts Page

To access the Delete Layouts page (IB_DOCLODEL) select PeopleTools >Mobile Application Platform >MAP Administration >Delete Layouts.

Image: Delete Layouts page

This example illustrates the fields and controls on the Delete Layouts page. You can find definitions for the fields and controls later on this page.

Layout Name	Enter the name of the layout you want to delete or click the Lookup button to search for it.
Search	Click the button to search the database for the layout you entered in the Layout Name field.
	The search results appear in the Layouts grid.
Select	In the Layouts grid select the box next to the layout that you want to delete.
Delete	Click the button to deleted the selected layout in the Layouts grid.

Renaming Layouts

This topic describes using the Rename Layouts page to rename a layout.

Using the Rename Layouts Page

To access the Rename Layouts page (IB_DOCLOREN) select PeopleTools >Mobile Application Platform >MAP Administration >Rename Layouts.

Image: Rename Layouts page

This example illustrates the fields and controls on the Rename Layouts page. You can find definitions for the fields and controls later on this page.

Layout Name	Enter the current name of the layout.
New Layout Name	Enter the new name of the layout.
Rename	Click the button to rename the layout.

Deleting Layout Templates

This topic describes using the Delete Layout Templates page to delete a layout template.

Using the Delete Templates Page

To access the Delete Templates page (IB_DOCTEMDEL) select PeopleTools >Mobile Application Platform >MAP Administration >Delete Templates.

Image: Delete Templates page

This example illustrates the fields and controls on the Delete Templates page. You can find definitions for the fields and controls later on this page.

Template Name	Enter the name of the layout template that you want to delete or click the Lookup button to search for it.
Search	Click the button to search the database for the layout template that you entered in the Layout Name field.
	The results appear in the Templates grid.
Select	In the Templates grid select the box next to the layout template that you want to delete.
Delete	Click the button to deleted the selected layout template in the Templates grid.

Administering Security

This topic provides an overview of the 4 MAP Security components — General, Permissions, Request Verification, and Security.

Using MAP Security General Page

Use the MAP Security general page to select from the available security options.

Navigation

PeopleTools >Mobile Application Platform >MAP Administration >MAP Security

Image: MAP Security — General page

This example illustrates the fields and controls on the MAP Security — General page. You can find definitions for the fields and controls later on this page.
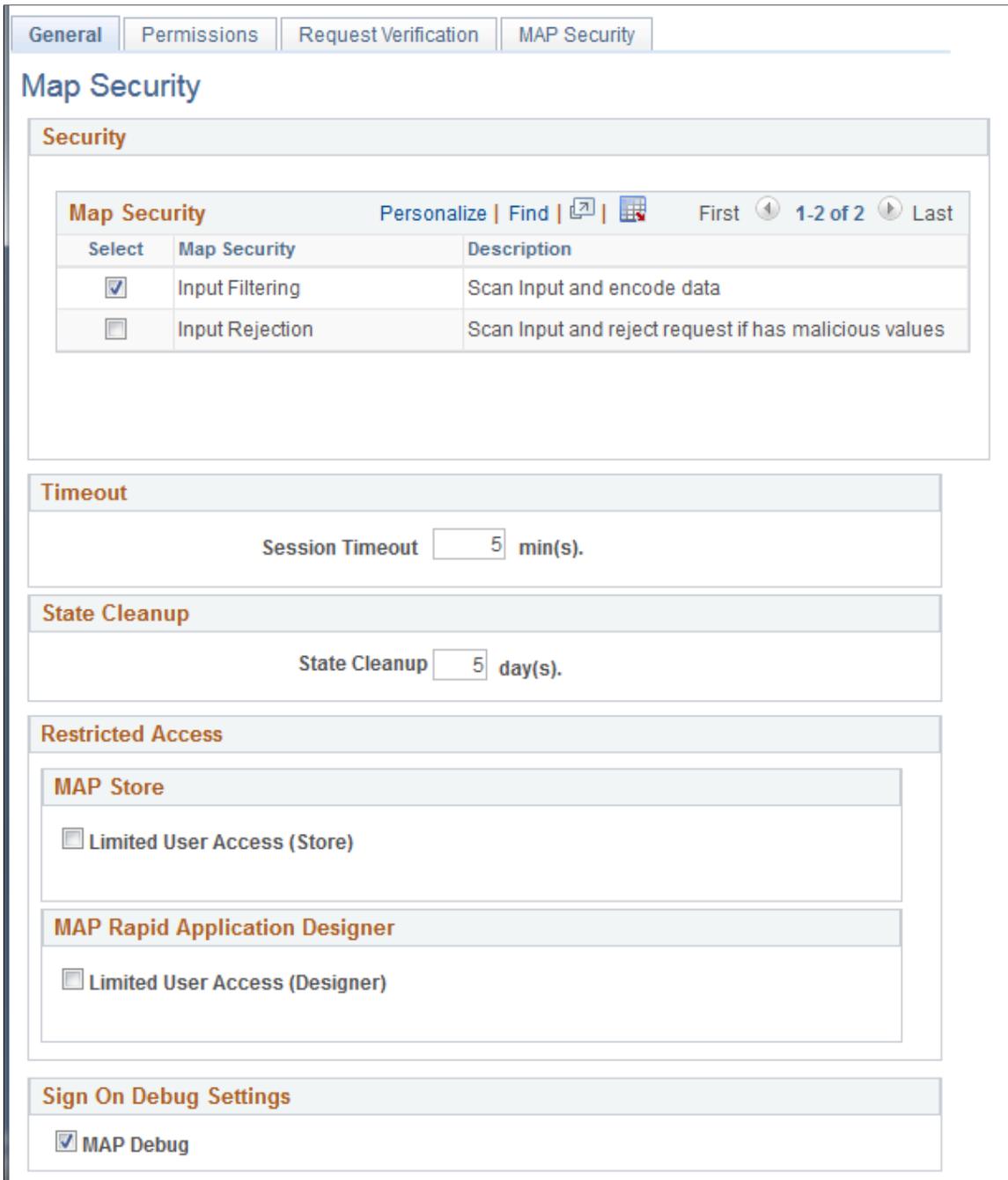
Input Filtering	Scan input data sent from the client and encode the data prior to populating the Document within the MAP object.
Input Rejection	This option is currently not implemented within the MAP framework.

Session Timeout	When the timeout is changed the update to the gateway will occur on the next File Synchronization, however it will not take effect until the Gateway is re-cycled.
State Cleanup	Set the number of days for applications that have the State functionality enabled to remove the old state. The default is 5 days.
Limited User Access (Store)	Select this checkbox to limit the applications and layouts available in the MAP Store. This option will only display layouts that the user has permission to access based on the permission list assigned to the layout or application.
Limited User Access (Designer)	Select this checkbox to limit the applications and layouts available in the Layout Rapid App Designer. This option will only display layouts that the user has permission to access based on the permission list assigned to the layout or application.
Map Debug	Select this checkbox to enable debugging from MAP application. When selected, the MAP Sign-On page will display a Set Trace link.

> **Note:** If the MAP application is launched from PIA, PIA tracing is not applicable to MAP. In order to enable MAP tracing, the user would need to set the tracing options using MAP application PT_ADMINISTRATION. The PT_ADMINISTRATION MAP application can be launched directly from the MAP store or as part of another MAP application. See Using PeopleSoft Delivered MAP Layouts and Applications for an example of creating a MAP application to set the debugging flags.

Using MAP Security — Permissions Page

Use the MAP Security — Permissions (IB_DOCLOPERM) page to update permission lists on selected layouts

Navigation

PeopleTools >Mobile Application Platform >MAP Administration >MAP Security

Image: MAP Security — Permissions page

This example illustrates the fields and controls on the MAP Security — Permissions page. You can find definitions for the fields and controls later on this page.

To update permission lists for a layout or layouts:

1. Optionally enter a Layout Name.

2. Optionally select the No permissions checkbox to list any layouts that do not belong to any permission lists.

3. Click Search and the layouts will be listed in the layouts grid.

4. Select the Layouts you wish to update Permissions.

5. Click Bulk Update.

6. Update the Permission list and set the Access.

7. Click Update.

Using MAP Security — Request Verification Page

Use the MAP Security — Request Verification page (IB_DOCLOREQVER) to update the Request Verification on selected layouts

Navigation

PeopleTools >Mobile Application Platform >MAP Administration >MAP Security

Image: MAP Security — Request Verification page

This example illustrates the fields and controls on the MAP Security — Request Verification page. You can find definitions for the fields and controls later on this page.

To update request verification for a layout or layouts:

1. Optionally enter a Layout Name.

2. Click Search and the layouts will be listed in the layouts grid.

3. Select the Layouts you wish to update the request verification.

4. Click Bulk Update.

5. Select the request verification to use.

6. Click Update.

Using the MAP Security — Security Page

Use the MAP Security — Security (IB_MAPRASEC) page to limit access to the MAP Store and Layout Rapid App Designer.

Navigation

PeopleTools >Mobile Application Platform >MAP Administration >MAP Security

Image: MAP Security — Security page

This example illustrates the fields and controls on the MAP Security — Security page. You can find definitions for the fields and controls later on this page.

MAP Store — Limit Access to Logged-In User	Select this checkbox to limit the applications available in the MAP Store based on the logged on user. This allows the customer to show only the applications that can be invoked by the logged on user accessing the MAP Store.
MAP Rapid Application Designer — Limit Access to Logged in User	Select this checkbox to limit the applications available in the Layout Rapid App Designer based on the logged on user.

Synchronizing MAP Files

MAP File synchronization is essential for proper execution of any MAP application. The MAP File synchronization process pushes the Templates, MAP Layouts and Images to the gateway with the latest update version. The View Gateway Files link can be used to verify that all files have been synchronized to the Gateway.

Typically when a Template or Layout is saved, synchronization occurs to update the files to the web server, however there are instances where manual synchronization is necessary. You should manually synchronize the files:

• After a software upgrade.

• Adding adding another Gateway for failover.

Note: For images to be properly uploaded to the gateway, the node ANONOYMOUS must have security access to the service operation PTIMAGE_GET.

Using the MAP File Synchronization Page

Use the MAP File Synchronization page to set up automatic synchronization and to manually synchronize file to the gateway.

Navigation

To access the MAP File Synchronization page (IB_DOCLOSYNC) select PeopleTools >Mobile Application Platform >MAP Administration >MAP File Synchronization.

or

Select MAP File Synchronization from the MAP WorkCenter.

Image: MAP File Synchronization page

This example illustrates the fields and controls on the MAP File Synchronization page. You can find definitions for the fields and controls later on this page.

Enable Autosync	To set up Auto Synchronization, Select the Enable Autosync checkbox to setup auto synchronization. You will need to provide a Sync Interval in minutes.
	Note: Select the Save button to properly enable auto synchronization
Sync Interval	Provide the Sync Interval in minutes. The minimum time is 10 minutes.
View Gateway Files group	For each gateway that is defined on your system, a View Gateway Files link is displayed.
	Select the View Gateway Files link to display all the objects returned from the gateway with the Web Server version and the Database version.
Synch	Click the Synch button to To perform a manual synchronization. If the synchronization is successful a message box will appear, otherwise an error will be displayed.

Cleanup	Click the Cleanup button to perform a cleanup of images, CSS and Java Scripts files on the Gateway when the files are no longer used or applicable. All images not used by any layouts will be removed along with any CSS and Java Script files.

Using the View Gateway Files Page

Use the View Gateway Files page (IB_DOCLOSYNC_SEC) to verify that all of your MAP files have been uploaded to the Web Server.

Navigation

Select the View Gateway Files link from the MAP File Synchronization page.

Image: View Gateway Files page

This example illustrates the fields and controls on the View Gateway Files page. You can find definitions for the fields and controls later on this page.

The file objects that have been uploaded to the Web Server will be displayed in 3 separate grids: Layouts, Templates, and Images. The Web Server version and the Database version should be at the same version number.

Reviewing Map Device Information

Use the Device Configuration Information page to review the devices that support MAP Applications.

Navigation

Select PeopleTools >Mobile Application Platform >MAP Administration >MAP Device Information

or

Select PeopleTools >Mobile Application Platform >MAP WorkCenter >MAP Device Information

Image: MAP Device Configuration page

This example illustrates the fields and controls on the MAP Device Configuration page. You can find definitions for the fields and controls later on this page.

This page identifies the device information for supported devices. This information can be used with PeopleCode in your MAP applications.

Type	Identifies the device type, such as phone, tablet, or desktop.
Platform	Identifies the device platform.
Keys	Identifies the keys for the device.

New devices can be added by creating a media query.

See Using the Media Queries Page

Configuring MAP for Fluid

MAP applications are executed outside of PIA (as REST service operations), however MAP applications can be called from PIA pages and return to the PIA page after execution. To the user, it may not be apparent if they are on a MAP page or a Fluid page as the look will be the same.

Multiple web servers and multiple domains can be set up for PIA and users will access the database based on the URL , domain, and signon provided. MAP automatically uses the default local gateway for inbound request processing, if additional inbound gateways are defined, they can be used when configured under a Load Balancer. The Gateway Page should be configured to use a Load Balancer and these Physical Gateway URLs defined accordingly. The Load Balancer URL will then be used to distribute processing. Additional domains may also be added, such as one domain used for internal users and another for external users. For REST Services, there is only one target location defined for the database, therefore Integration Broker needs to know how to route a MAP request from a PIA page that could be coming from a different gateway or domain, as well as how to return to the PIA page.

By default, MAP applications use the URL specified in the REST Target Locations. The Gateway defined for the REST Target Location URL also needs to be registered on the Inbound Gateway page as part of the Integration Broker Gateway setup. If a Load Balancer is not defined for the Gateway then the defined Gateway URL will automatically register itself on the Inbound Gateway Page. The administrator should verify that the machine and port used to identify the location of the Gateway is the same as defined for the REST Target Location.

The MAP application URL being invoked from PIA must be defined within the same domain as PIA, therefore when PIA has multiple domains defined, the MAP URL must be defined based on PIA and not what is defined on the REST Target Location.

See "Using the Target Locations Page to Set Target Locations for Services" (PeopleTools 8.57: Integration Broker Administration)

In order for MAP applications to use the Domain Name and Site Name (same webserver instance) from the actual PIA request, the PIA domains must be defined in the Multi Domain section of the MAP Fluid Configuration page. This allows MAP applications to be run under different domain configurations for multiple PIA installations that have different Domain and Site Names.

To configure MAP for Fluid:

1. Select PeopleTools, Mobile Application Platform, Mobile Administration, MAP Fluid Configuration.

   Alternatively, select PeopleTools, Mobile Application Platform, MAP WorkCenter, MAP Fluid Configuration.

2. The Node Details are presented as read-only and are populated based on the default Portal node. The context URI Text and the Portal URI Text are used to provide PIA navigation. Integration Broker is not aware of this information, however MAP applications need this information in order to provide a seamless flow between MAP applications and Fluid pages. The top of the MAP Fluid Configuration page is populated with the information for the default local node used by the Portal. To change the information provided, select PeopleTools, Portal, Node Definitions.

   See "Defining Portal Nodes" (PeopleTools 8.57: Portal Technology)

3. The Common Gateway (Physical Gateway URL) for each Domain must then be properly registered for each domain. This will allow for a MAP File Synchronization to push all the necessary files out

to each registered Gateway. By default, MAP applications use the URL specified in the REST Target Locations, this URL will be automatically populated into the Physical Gateways.

a. Domain Token. The Domain name is part of the Gateway URL that will have all PeopleSoft nodes defined within that Domain.

b. Load Balancer. If a Load Balancer is used enter Load Balancing URL. The physical gateways used for that load balancer must still be registered under the Physical Gateway section for that Domain.

c. Physical Gateways. Enter the physical gateways used for the Domain.

d. Use the Properties link for each Gateway URL to open the Integration Properties file for that Gateway in order to add the Nodes and Keystore Password.

Image: MAP Fluid Configuration page

This example illustrates the fields and controls on the MAP Fluid Configuration page. You can find definitions for the fields and controls later on this page.

Enable Multiple Domain	Select this check box to enable multiple domain configurations.
Domain Token	Enter the Domain Token name used for each PIA defined domain.
Load Balancing URL	Optional. If a Load Balancer is used enter the Load Balancing URL. The Physical Gateways used for that load balancer must still be registered under the Physical Gateway section for that Domain.
Gateway URL	Enter the URL for the physical gateway.
Ping	Use this button to ping the gateway to see if it is active.

Properties   Select this link to set up the properties for the physical gateway.
                                                    For any additional inbound gateways, you must define:

- Default local node

- Keystore password

Validating Custom Layouts

To validate a custom layout:

1. Select PeopleTools, Mobile Application Platform, MAP Administration, Validate Custom Layout.

2. On the search page click Search to display a list of all Custom Layouts, or select a specific Custom
   Layout name.

3. Select the layout names to validate and click Validate.

   Image: Validate Custom Layouts page

   This is an example of the Validate Custom Layouts page.

4. If the validation is successful, you will get a message *Successfully Validated*. If the validation fails,
   you will get a message *Failed Validation* and a link to the layout in the Layout Designer. The error(s)
   will then be displayed, as validation gets performed by default for a Custom Type Layout when the
   Layout Designer is rendered.

Chapter 56

Using Mobile Application Platform Utilities

Uploading and Updating Images for the Mobile Application Platform

This topic describes how to upload and update images for the Mobile Application Platform.

Understanding Uploading and Updating Images for the Mobile Application Platform

The Mobile Application Platform provides two pages for uploading and updating images for use in layouts, the Layout Image Upload page and the Upload Image page.

You can also upload and update images in PeopleSoft Application Designer, as these pages simply provide an alternative.

Note that neither the Layout Image Upload page nor the Upload Image page provide any image editing features.

To add a new image to the database you name the image using the Layout Image Upload page. After naming the image, the Image Upload page appears where you can upload the image.

To update an image already in the database or in use in a Mobile Application Platform application, you use the Image Upload page. This page enables you to replace an existing image with a new one.

Using the Layout Image Upload Page

Use the Layout Image Upload page to add the image name to the system. To access the page select PeopleTools >Mobile Application Platform >Utilities >Image Upload and click the Add a New Value tab.

Image: Layout Image Upload page

This example illustrates the fields and controls on the Layout Image Upload page. You can find definitions for the fields and controls later on this page.

In the Content Name field, enter the name for the image.

After you name the image, the Image Upload page appears that you use to upload the image to the system.

Using the Upload Image Page

Use the Upload Image page to upload a new image to the system or to update/replace an existing image in the system.

When you are adding a new image to the system, the page automatically appears after you name the image using the Layout Image Upload page.

When you want to update an image access the page by selecting PeopleTools >Mobile Application Platform >Utilities >Image Upload and search for and select the image to update.

Image: Image Upload page

This example illustrates the fields and controls on the Image Upload page when adding a new image to the system. You can find definitions for the fields and controls later on this page.
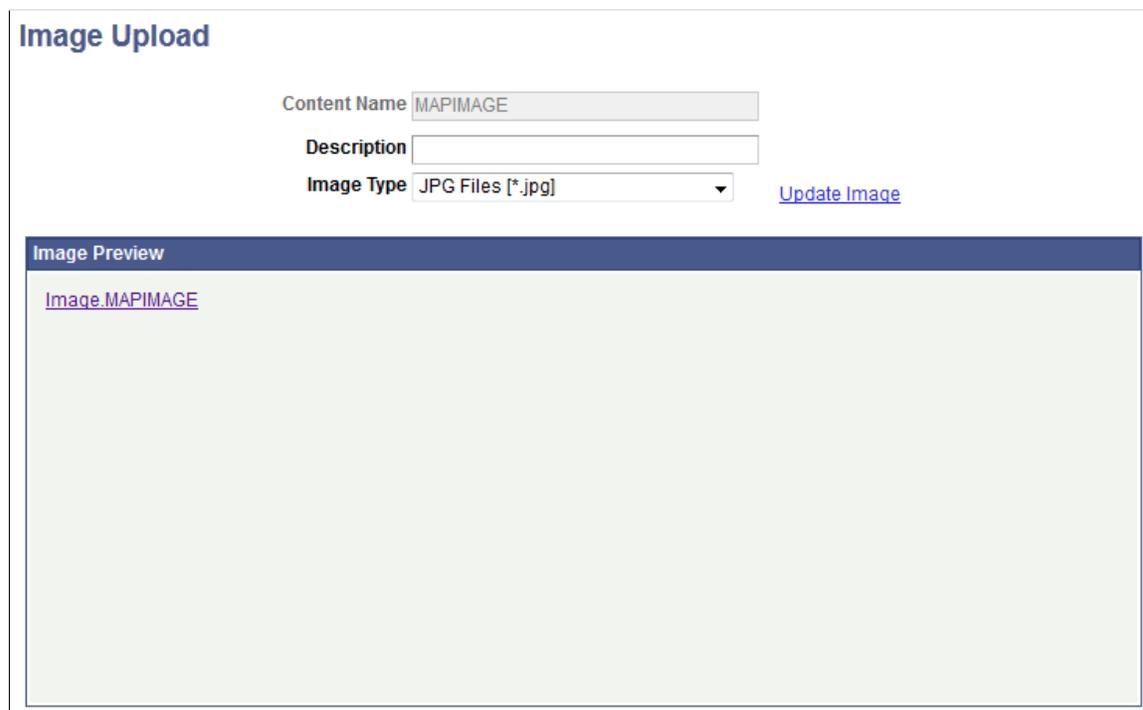
After you add an image to the system it appears in the Image Preview section of the page.

Image: Image Upload page

This example illustrates the fields and controls on the Image Upload page after you have uploaded an image to the system or are accessing the page to update an existing image. You can find definitions for the fields and controls later on this page.

Content Name	This read-only field displays the name of the image defined when the image was added to the system.
Description	Enter a description for the image.
Image Type	Select the image format to add. The supported file formats are:

- *BMP Files [*.bmp]*. Bitmap format.

- *DIB Files [*.dib]*. Device Independent Bitmap format.

- *GIF Files [*.gif]*. Graphics Interchange Format format.

- *JPG Files [*jpg]*. (Default.) Joint Photographic Experts Group format.

Update Image	Click the link to browse for and upload an image.
Image Test	Click the link to view the true size of the image in the Image Preview area.

Uploading Images for Use in the Mobile Application Platform

To upload an image for use in the Mobile Application Platform:

1. Access the Layout Image Upload page.

To access the page select PeopleTools >Mobile Application Platform >Utilities >Image Upload and click the Add a New Value tab.

2. In the Content Name field enter a name for the image.

3. Click the Add button.

The Image Upload page appears.

4. In the Description field, enter a description of the image.

5. From the Image Type drop-down list select the format of the image you are uploading.

6. Click the Update Image link.

The File Attachment dialog box appears.

7. Browse for the image to upload and click the Upload button.

The Image Upload page appears and the uploaded image appears in the Image Preview pane.

To view the true size of the image, click the image in the Image Preview section.

Updating Images in Use in the Mobile Application Platform

Use the information in this section to replace an image in use in the Mobile Application Platform with another image.

To update an image in use in the Mobile Application Platform:

1. Access the Image Upload page.

To access the page select PeopleTools >Mobile Application Platform >Utilities >Image Upload.

2. Click the Update Image link.

The File Attachment dialog box appears.

3. Browse for the image to upload and click the Upload button.

The Image Upload page appears and the uploaded image appears in the Image Preview pane.

To view the true size of the image, click the image in the Image Preview section.

Dynamically Registering Images

This topic provides an overview of registering images for dynamic display functionality in the Mobile Application Platform .

Understanding Dynamically Registering Images

You can register dynamic images defined as primitive elements to ensure that they are pushed to the integration gateway upon file synchronization.

Use the following PeopleCode to get the proper image URL to add to the designated dynamic image primitive element:

```
&string = %IntBroker. GetImageURL(string <IMAGE NAME>)
```

Prerequisites for Dynamically Registering Images

To dynamically register images for use in the Mobile Application Platform, they must already exist in the database.

Use the Image Upload page or PeopleSoft Application Designer to add images to the database.

Related Links

Uploading and Updating Images for the Mobile Application Platform
"Creating Image Definitions" (PeopleTools 8.57: Application Designer Developer's Guide)

Using the MAP Dynamic Image Registration Page

Use the MAP Dynamic Image Registration page (IB_DOCLOIMGREG) to add images for dynamic registration.

To access the page select PeopleTools >Mobile Application Platform >MAP Utilities >Dynamic Image Registration.

Image: MAP Dynamic Image Registration page

This example illustrates the fields and controls on the MAP Dynamic Image Registration page. You can find definitions for the fields and controls later on this page.
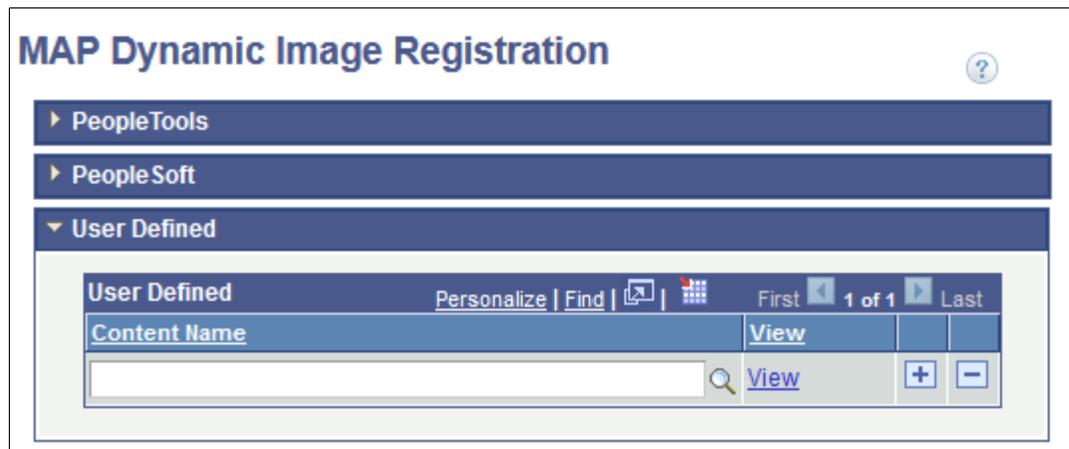
The MAP Dynamic Image Registration page feature three sections:

PeopleTools	The PeopleTools section contains read-only images defined as part of the Mobile Application Platform framework and delivered with PeopleTools.
	Note: The images in this section are read-only.

As images are add as part of the framework, these can be added without the possibility of getting removed as part of a Project Copy by applications or by the user.

PeopleSoft

The PeopleSoft section is for use by application pillars to add images. As with the PeopleTools section images, these images are not removed during Project Copy.

User Defined

The User Defined section is for customers to add their images.

The following controls appear on the page:

Click the icon to expand a section on the page.

Click the icon to collapse a section on the page.

Content Name

Enter the name of the image or use the Lookup button to search for an image in the database.

View

Click the link to access the View Image page to preview/view the image.

Using the View Image Page

Use the View Image page to preview an image before you add it to the MAP Dynamic Image Registration page or to view and existing image.

To access the page click the View link in any of the following grids on the MAP Dynamic Image Registration page:

- PeopleTools grid.

- PeopleSoft grid.

- User Defined grid.

Image: View Image page

This example illustrates the fields and controls on the View Image page. You can find definitions for the fields and controls later on this page.
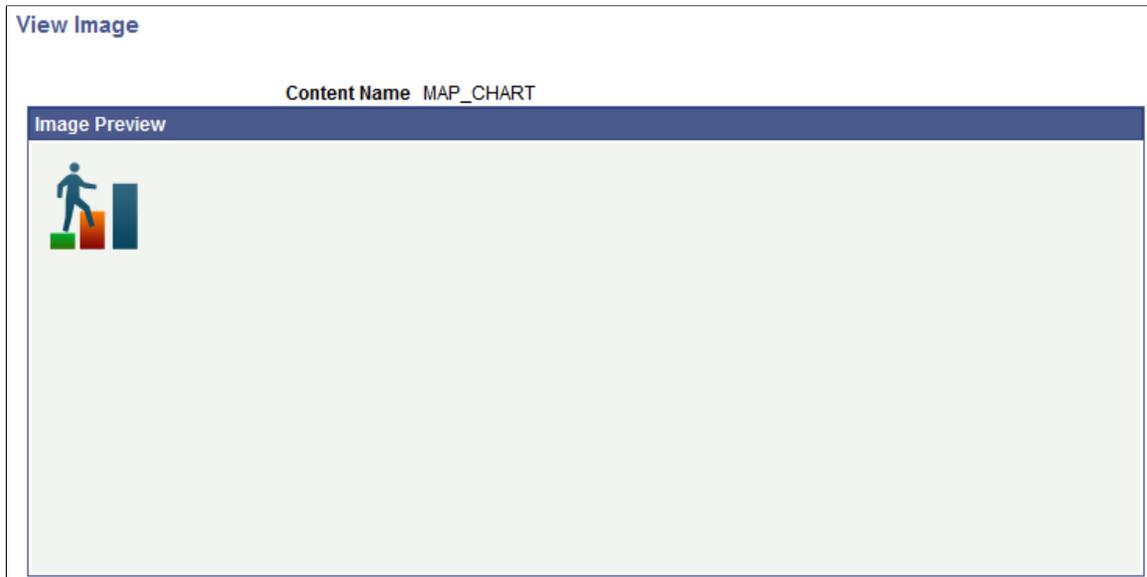
Content Name   Displays the name of the image.

Return   Click the link to return to the MAP Dynamic Image Registration
                                              page.

Chapter 57

Using MAP Store

Understanding the MAP Store

The MAP Store provides a location to store all MAP applications that are available, as well as a repository for MAP content that can be used to build new MAP applications using the Layout Rapid Application Designer. The MAP Store includes a detailed description of what the application will provide, as well as the actual URL to invoke the application.

Developers will register layouts as a complete application or only register certain aspects of a layout based on a container within the layout. Developers should include a very detailed description of what the layout or layout container performs. This description will be the primary source of information the end user will use when they create applications using the Layout Rapid Application Designer.

Adding MAP Layout to the MAP Store

To add a layout to the MAP Store:

1. Select PeopleTools, Mobile Application Platform, MAP Administration, MAP Store.

2. Click the Register New Layout Link.

3. Enter the required informations and click the Details link.

4. Click the Add Sequence button on the MAP Store Layout Details page to add the URI sequences that are applicable for this application.

5. Click OK.

6. Click Save to register the application.

Important! In order to register Layouts that were created prior to PeopleTools 8.55, the Layout needs to be saved in PeopleTools 8.55 before it can be added. In the Layout Designer open the layout and click Save.

Using the MAP Store Page

Use the MAP Store page (IB_MAPSTORE) to add a layout to the MAP Store.

Navigation

PeopleTools > Mobile Application Platform > MAP Administration >MAP Store.

Click the Register New Layout Link.

Image: MAP Store page

This example illustrates the fields and controls on the MAP Store page. You can find definitions for the fields and controls later on this page.

Layout Name	Select the layout name for the layout that contains the application or specific containers you wish to register.
Location	Location refers to the where the layout will be used in an application. Valid Values are: • Content • Header • Footer **Note:** This field is not used for application registration.
Page	Page number on the layout where the container is defined. **Note:** This field is not used for application registration.
Container	Select the container to use. **Note:** This field is not used for application registration.
Group	Groups are used to identify layouts that can logically be grouped together. The application can be assigned to an existing group or a new group can be created by simply typing in the new Group Name. **Note:** Groups are used to search for applications on both on the Map Store page and Layout Rapid App Designer page.
Description	Add description that identifies the application or container.

> **Note:** The description is used on the Layout Rapid App Designer page as a way to search for an application.

Long Description	Add a very detailed description explaining the purpose of the application and exactly what it will do.

> **Note:** The long description is used on the Layout Rapid App Designer page as a way to search for an application.

Details	This link is used to open the MAP Store Details page.
Application	Select this checkbox to indicate that this layout is a complete application.

> **Note:** If this checkbox is selected, the fields Location, Page and Container are hidden and not applicable.

Using the MAP Store Details Page

Use the Map Store Layout Details page to add URI sequences for either a complete application or a container application.

Navigation

Click the Details link on the Map Store page.

Image: MAP Store Layout Details page

This example illustrates the fields and controls on the MAP Store Layout Details page. You can find definitions for the fields and controls later on this page.

Owner ID	(Optional) Select the object owner for this MAP application.
Class	(Optional) Enter a class to use as a wrapper when this layout is used within the Layout Rapid App Designer.

Style	(Optional) Enter a style to use as a wrapper when this layout is used within the Layout Rapid App Designer.
Add Sequence	Determine which URI sequences are applicable for this application. At least one must be registered. Use the Add Sequence button to get a list of the URI sequence numbers and a description of each. **Note:** If a description of the URI is not displayed, a description should be added on the appropriate layout as this description is used in the Layout Rapid App Designer to determine which URI to select.
Delete	Use this button to delete a URI sequence that has been selected, simply enable the checkbox on the appropriate row and then select the Delete button.

Using the Select URI Sequence Page

The URI sequences available in the selected application are displayed.

Navigation

Click the Add Sequence button on the MAP Store Layout Details page.

Image: Select URI Sequence page

This is an example of the Select URI Sequence page.

Searching Map Store

The Map Store will contain all the registered applications. Use the MAP Store search page to:

- Select and edit a registered MAP Layout or Layout Container.

- Retrieve the URL for a registered application.

- Register a new layout using the Register a New Layout link.

Using the MAP Store Search Page

Use the MAP Store Search page (IB_MAPSTORESEARCH) to search for registered applications and layouts.

Navigation

PeopleTools > Mobile Application Platform > MAP Administration >MAP Store

Image: MAP Store Search page

This example illustrates the fields and controls on the MAP Store Search page. You can find definitions for the fields and controls later on this page.

In this example, there is one registered application and 3 registered containers. For registered applications, use the URL to invoke the application.

Layout Name	Search based on layout name.
Description	Search based on description.
Group	Search based on group name.
Application	Select this checkbox to search for complete applications only.

Note: PeopleSoft delivers containers and applications that begin with PT.

Related Links

Adding MAP Layout to the MAP Store

Chapter 58

Creating MAP Applications Using Layout Rapid App Designer

Describing Layout Rapid App Designer

The Layout Rapid App Designer provides the ability to connect existing MAP applications together to build a customized MAP application without having to write any People Code, Java Script or CSS. This designer can be used not only by development but primarily by an actual end user. Development uses the Layout Designer to build small specific applications and register these applications or containers in the MAP store. These applications can then be selected by the end use to build a complete application.

In order to take full advantage of this designer MAP applications registered in the MAP store should be very descriptive. The end user should be able to properly determine what the application will actually do and what parameters are needed in order to get the desired result from the description provided when the application or container was registered. The end user can then use the concept of *connected applications* to create a customized application for their needs. Connected applications refer to one selected MAP application from the MAP store that can be connected via its defined document (business logic) to another MAP application. For example one MAP application when executed generates a purchase order. This purchase order can then be passed into another MAP applications which can use that defined purchase order to execute additional business logic.

Building an Application Using Layout Rapid App Designer

To build a new application using Layout Rapid App Designer:

1. Select PeopleTools, Mobile Application Designer, Layout Rapid App Designer.

2. Click the Add New Layout link.

3. Enter a layout name and select the template for the application, then click OK.

4. Use the Layout Rapid App Designer to create the application.

See Adding New Layout in Layout Rapid App Designer and Using Layout Rapid App Designer.

Adding New Layout in Layout Rapid App Designer

Use the Enter a New Layout Name page (IB_MAPRASEARCH_SEC) to add a new rapid application layout to the system.

Navigation

To access the page, select PeopleTools > Mobile Application Platform > Layout Rapid App Designer .
Click the Add New Layout link.

Image: Enter a New Layout Name page

This example illustrates the fields and controls on the Enter a New Layout Name page. You can find
definitions for the fields and controls later on this page.

| Layout Name	Enter a layout name.
Template Name	Enter or select the name of the layout template to apply to the layout. Two templates specifically designed for rapid app are delivered: • PT_TEMPLATE_ONE: Single pane, mobile template • PT_TEMPLATE_TWO: Two pane, tablet template
View Template Layout	Click the link to view the template specified Template Name.
Include PeopleSoft Header	Check the box to include the header from the PeopleSoft Fluid interface as the header for the application. When the PeopleSoft Header is selected, it will automatically include the PT_FLUID_TEMPLATE in your MAP application. The PT_FLUID_TEMPLATE is used to get FLUID styles for MAP applications.

Using Layout Rapid App Designer

Use the Layout Rapid App Designer page (IB_MAPRASEARCH_SEC) to build a new layout.

Navigation

To access the page, select PeopleTools > Mobile Application Platform > Layout Rapid App Designer .

Image: Layout Rapid App Designer page

This example illustrates the fields and controls on the Layout Rapid App Designer page. You can find definitions for the fields and controls later on this page
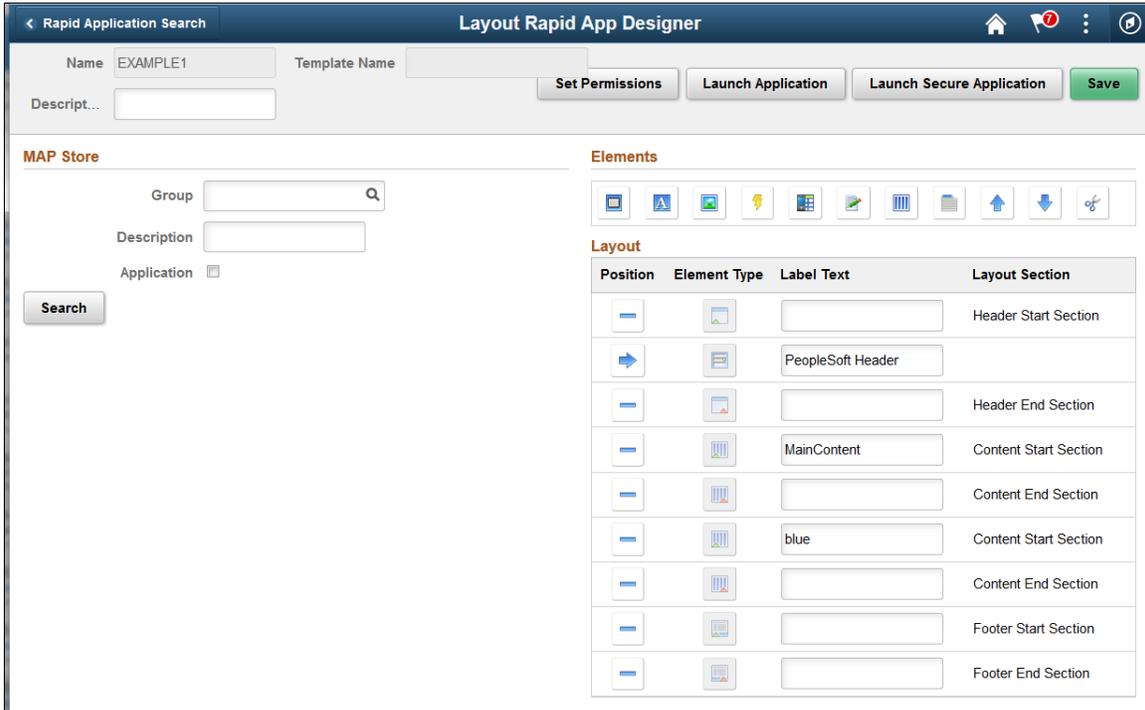
Description	Enter a description for your application.
Elements	This section provides a list of elements to build the layout.
Map Store	This section provides a list of applications available to select from when building the application.
Layout	This is the actual layout grid to define the layout.

Elements

The following elements are available in Layout Rapid App Designer.

Button	Label
	Add Button
	Add Static Text
	Add Image

Button	Label
	Add URL
	Add Mobile Grid
	Add Input
	Add Container
	Select Menu

Use the following buttons to move and delete page elements in the Layout grid:

Button	Label	Description
	Move Item Up	Click a row in the Layout grid and click the button to move the selected page element up in the grid.
	Move Item Down	Click a row in the Layout grid and click the button to move the selected page element down in the grid.
	Delete Current Row	Click a row in the grid and click the button to delete the selected page element from the page.

Map Store

This section provides the ability to search for a list of containers registered in the MAP Store that can be added to your layout. You can narrow down the search criteria by selecting a specific group, matching a description or requesting only applications.

Image: Map Store Search in Layout Rapid App Designer

This example illustrates the fields and controls on the Map Store Search in Layout Rapid App Designer. You can find definitions for the fields and controls later on this page.

The search results are displayed.

Icon	An icon is displayed to identify the layout as either an application or container.
Description	The description for the layout is displayed.
Long Description	Click this link to view the long description for the layout.

Managing Headers and Footers in Layout Rapid App Designer

In the Layout Rapid App Designer, the header and footer sections are defined in the Layout Grid by default. This topic discusses how to manage headers and footers in Layout Rapid App Designer.

Including Elements in Header and Footer Sections

To insert elements in the header or footer section, place the current row (arrow icon) on the Header Start Section or the Footer Start Section.

You can include the following page elements into header and footer sections:

• Button

• URL

• Mobile Grid

• Container

• Static text

Note: Mobile Grid and Input are not allowed in a Container in a Header or Footer. If a Container is deleted from the Header or Footer all elements contained within will also be deleted.

Setting Header or Footer Properties

The same properties are available for both headers and footers. The property will determine how the header or footer is displayed on the page.

To set the header properties click on the Header icon in the layout grid. To set the footer properties, click on the Footer icon on the layout grid.

Image: Header properties page

This example illustrates the fields and controls on the Header properties page. You can find definitions for the fields and controls later on this page.
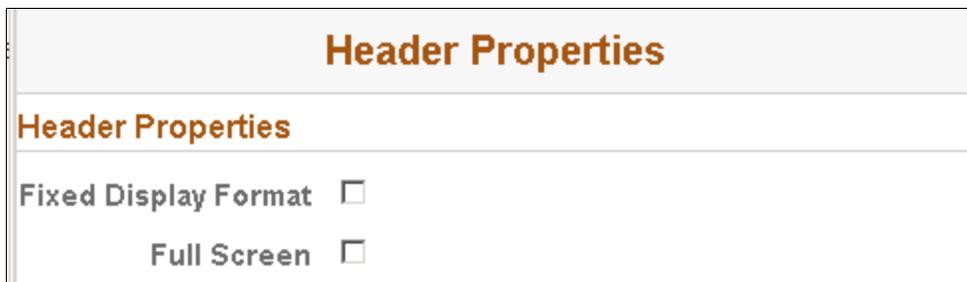
Fixed Display Format	For headers, the header will appear at the top of the viewport and remain there as the user scrolls, regardless of where the user is in the content.
	For footers, the footer will appear at the bottom of the viewport and remain there as the user scrolls, regardless of where the user is in the content.
Full Screen	For headers, the header will appear within the viewport, however as the user scrolls the header will also scroll and may no longer be displayed.
	For footers, the footer will appear within the viewport, however as the user scrolls the footer will also scroll and may no longer be displayed.

# Adding Input Elements in Layout Rapid App Designer

When you create a layout in Layout Rapid App Designer, it is not associated with a document layout, therefore you may need to add input elements before you can add your containers from the MAP Store.

To add an input page element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row for Main Content start section.

4. Click the input icon.

5. The Create Primitive page is displayed. Select the type for your new primitive and enter the appropriate length and description.

6. Click OK.

7. A message is displayed to Save the layout before you can use the field. Click Save.

8. If this is the first Input saved an additional element of type button will be defined on the layout. The Update button cannot be deleted as this is the only way to take any input values entered by the user.

Image: Example layout with input element added

This example illustrates the layout with input element added. The Update button was added when the layout was saved.
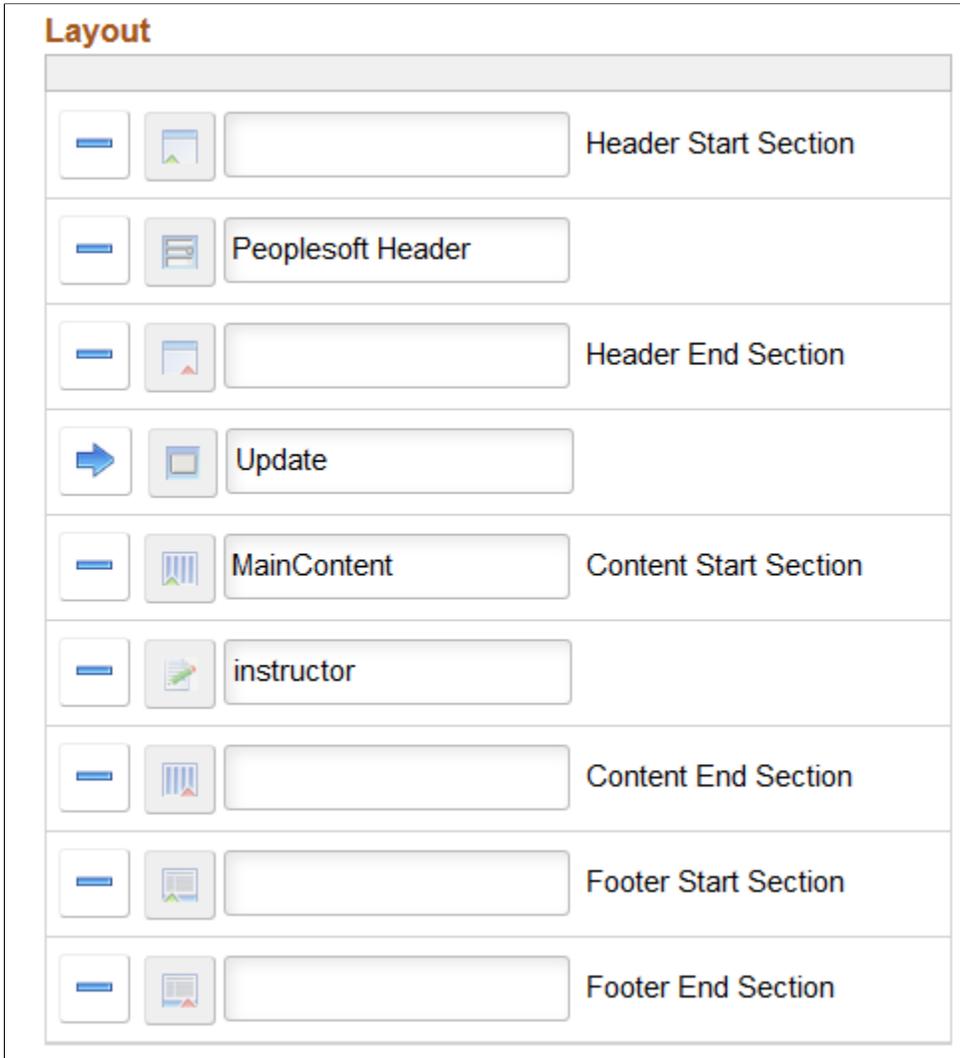
Creating the Primitive

When you click the Input icon, the Create Primitive page (IB_MAPRAPRMSEL_SEC) is displayed.

Image: Create Primitive page

This example illustrates the fields and controls on the Create Primitive page. You can find definitions for the fields and controls later on this page.

Type	Select the input type. Based on the type selected
	• String
	• Integer
	• Date-Time
	• Date
	• Boolean
Length	Enter the length if applicable.
Description	Enter a description for the primitive.

Setting Input Properties

The input primitive will be defaulted based on the type selected. The framework supports basic HTML 5 text input controls, and renders them according. Input types such as Email, Telephone, URL, Search and Number are new input types for HTML 5. Most touch devices don't have physical keyboards. When one of these Input types is defined, the appropriate optimized virtual keyboard will be displayed.

The developer can select another input type that would correspond with the primitive type. For example a primitive of type String, will default in as text. The developer can select another type such as number, password, input range, month, URL, telephone, week, email, or hide text area.

Navigation

Click the Input icon on the layout grid to display the input properties.

Image: Input Properties page

This example illustrates the fields and controls on the Input Properties page. You can find definitions for the fields and controls later on this page.
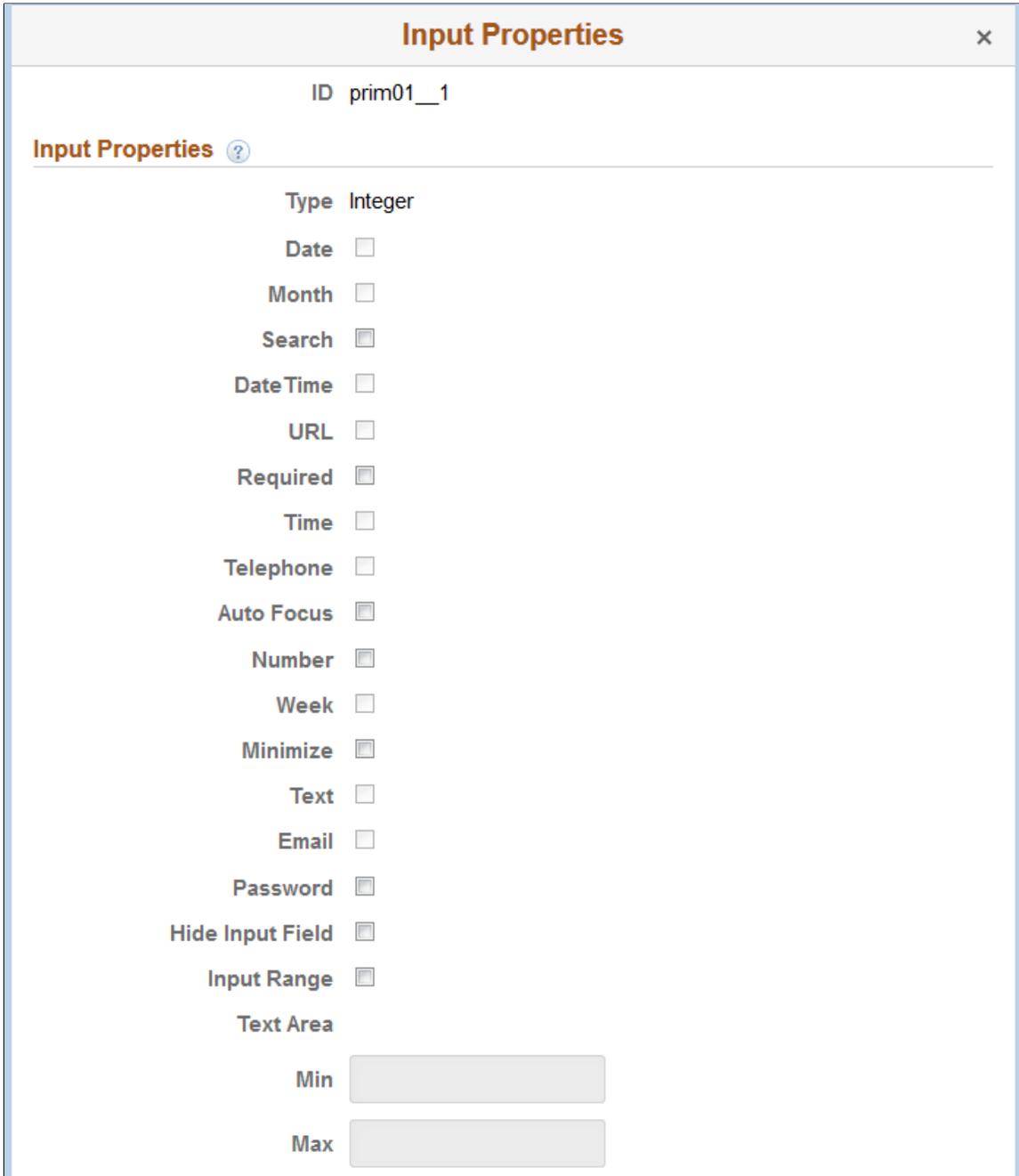
Auto Focus	Select the property for the input element to automatically get focus when the page loads.
Input Range	When you select this input type the Max and Min fields become enabled for you to define the minimum and maximum range.
Minimize	Select the option to create a more compact version of the element.

Required	Specify that an input field must be filled out before submitting the form.
Search	Select this property to define the input with a search field.

Managing Referenced Containers in Layout Rapid App Designer

To add containers to layout grid:

1. Use the Map Store Search to display the available layouts in the MAP Store.

2. Position your cursor on the row above where you want to add the container.

3. Click the Container icon in the MAP Store search list.

4. The URI mapping page is displayed.

5. After the URI mapping is complete, the container is added to the grid layout.

Using the URI Mapping Page

Use the URI Mapping page to define the primitives to pass to the application.

Image: URI Mapping page

This example illustrates the fields and controls on the URI Mapping page. You can find definitions for the fields and controls later on this page.

Mapping Layouts	Click the drop down to select from the available mapping layouts for this application.
	Once selected, the available primitives will be populated in the Available Primitive section.
Available Layout URIs	Select the URI applicable to your application.
URI Elements Requiring Mapping	You can define constant or primitive values for URI parameter values. When you select the Constant control, a Const Value

field appears where you enter the constant value. When you select the Primitive control, a Select Primitive link appears to access the Select Primitive page to define a primitive for the URI parameter.

Select Primitive   This link will appear when you select Primitive. Click the link to open the Primitive Mapping page.

Managing Containers in Layout Rapid App Designer

When you add a new layout using Layout Rapid Application Designer, containers as defined on the selected template will be defaulted on the grid. Containers can be nested, meaning that a container can hold other containers. Select of the Add Container icon will create a start and end container. The developer can then add additional element types to the container.

Defining Container Properties for Class

Any style changes in the Rapid Application are applied using the Container Properties Class section. When you create a new Rapid Application Layout, you need to specify the template. In Template Designer you can add additional classes, styles and sub templates to the template. See Using the Containers Page and Using the Add Sub Templates Page.

To apply class properties to the container:

1. If this is a new layout, save the layout.

    Note: You must save the new layout for the Container Class Properties to display.

2. Add a new container if necessary.

3. In the layout grid, click on the icon for Container Start Section.

Image: Container Properties – Class Section

This example illustrates the fields and controls on the Container Properties – Class Section. You can find definitions for the fields and controls later on this page.


Template Name	The template name for this rapid application is displayed.
Sub Template Classes	Select this link to select a sub template class.
Containers	The containers for the template are displayed.
Classes	Expand this section to view the available classes.
Add Class	To add a class, select the class and click the Add Class button.
Clear	Click this button to clear all classes.

Adding Sub Template Classes

To add sub template elements to the container:

1. Click the Sub Template Classes link.

2. Select the Sub Template from the drop down.

3. Select the class.

Image: Select Sub Template Class Information page

This example illustrates the fields and controls on the Select Sub Template Class Information page. You can find definitions for the fields and controls later on this page.
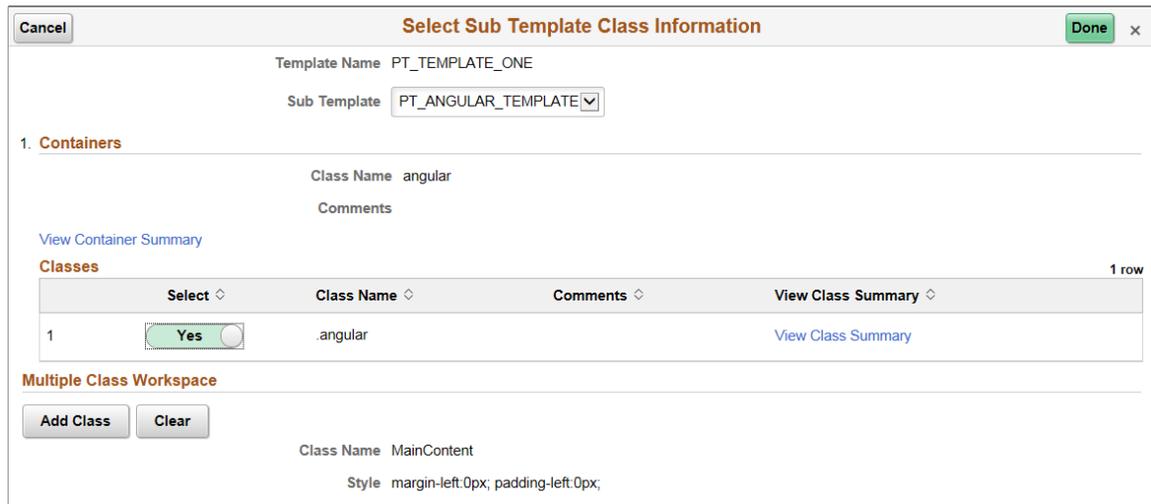
4. Click the Add Class button.

    The class will be added to the existing class.

    Image: Sub Template Class added

    This example illustrates the Multiple Class Workspace with the Sub Template Class added.
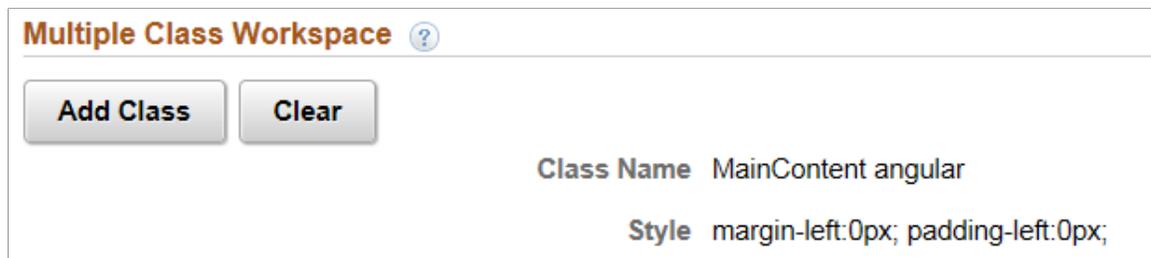
5. Click Done.

Adding Classes

To add additional classes, the classes must be available on the template for the Rapid Application.

1. Expand the Classes for the Container you want to add.

2. Select the Class.

Image: Class Selection with Classes – Classes

This example illustrates the Class Selection section with the Classes expanded for the Container blue.

3. Click Add Class.

4. The class is added.

Image: Class added

This example illustrates Container Properties page where a second class has been added.

Defining Container Properties

Image: Container Properties page

This example illustrates the fields and controls on the Container Properties page. You can find definitions for the fields and controls later on this page.

Collapsible Content

In order to create content that can expand and contract, the Collapsible check box should be selected. This will create a header based on the label Text and Content area. The header acts as a button that can be touched to expand and contract the content area. By default, the content area will be collapsed, unless the collapsed check box is disabled. Collapsible content areas can also be grouped to create an accordion. An accordion is essentially a group of collapsible content areas that are aware of each other. To define an accordion, select the collapsible Set check box for the parent container. Add other containers within this

defined collapsible Set container. Ensure to select the collapsible check box for the containers added to this parent (containers defined as collapsible set).

Nested Collapsible Content – You can nest collapsible content. An automatic margin will be created for every new level of collapsible panel. It's recommended not to add more than two levels to avoid UI and DOM complexity.

Accordion – Allows you to group components of collapsible content so that only one panel can be visible at a time. So when you view one panel (content area), all others will be closed

Collapsible	Select to create collapsible content.
Initially Collapsed	By default, collapsible content will be opened when the page loads. Hide the content by selecting this checkbox.
Collapse Set	To create an accordion. By default the framework opens the last collapsible panel in a collapsible set.

Popup Options

To pop up a Container (content area) use a Button or Link.

Select the Popup option to create an overlay (with the contents of the container) that floats above the current page. Select the optional Close button if you wish to have a close button as part of the overlay. By default, the popup logic will show as a Button using the Label Text of the Container for the Button label. Select the optional, Show as Link checkbox to show a link with the Container Text label.

Note: Chaining of popup is not currently supported. If the parent container has popup enabled, any child containers that include popup will not work. The jQuery Mobile Popups framework does not currently support chaining of popups so it's not possible to embed a link from one popup to another popup.

Enable Popup	Select to enable the container to act as a popup.
Popup Position	By default, pop-up content opens centered vertically and horizontally over the page element/component/object selected (also known as the origin).
	The Mobile Application Platform framework also applies some basic collision detection rules to ensure that pop-up content appears on-screen, so the ultimate position may not always be centered over the origin.
	To change the default position of pop-up content, the options are:
	• *Window*. (Default.) Position centered vertically and horizontally over origin.
	• *Origin*: Position over the button or link used to invoke the pop-up content.
Remove Corners	By default buttons appear with rounded corners.
Close Popup Button	Select the box to insert a close button for the pop-up content.

	After you select this control, the Close Button Position drop-down list box appears for selecting the position of the close button in the pop-up.
Remove Icon Shadow	By default icons appear with a shadow effect. Select this property to remove the shadow.
Show as Link	Select to To show the popup as a link instead of a button. The following options will then be available:

- Popup Position

- Close Popup Button

- Overlay Theme

Overlay Theme	Theme of the popup's background, which covers the entire window behind the popup. From the drop-down list select a jQuery theme for the overlay layer of the select menu.
	For more information about jQuery themes in the Mobile Application Platform see Defining jQuery Mobile Theme Properties
Remove Shadow	By default buttons and icons appear with a shadow effect. This option removes the shadow.
Inline Button	Select this box to use an inline button to invoke the popup content.. Inline buttons do not use the whole width of the view port
Button Theme	Theme to use on the button
Mini	Create a more compact version of the button element

Managing Image Elements in Layout Rapid Apps Designer

To add static text page element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row above where you want to add the image.

4. Click the image icon.

5. The Image Properties page is displayed.

6. Select the Add Image checkbox.

7. Use the Image Selector link to select the image from the grid of images displayed.

8. After the image is selected, click OK to add the image to the layout grid.
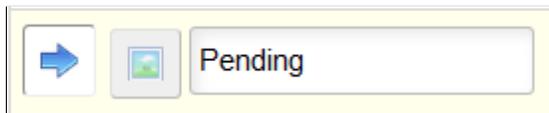
Image: Image Properties page

This example illustrates the fields and controls on the Image Properties page.
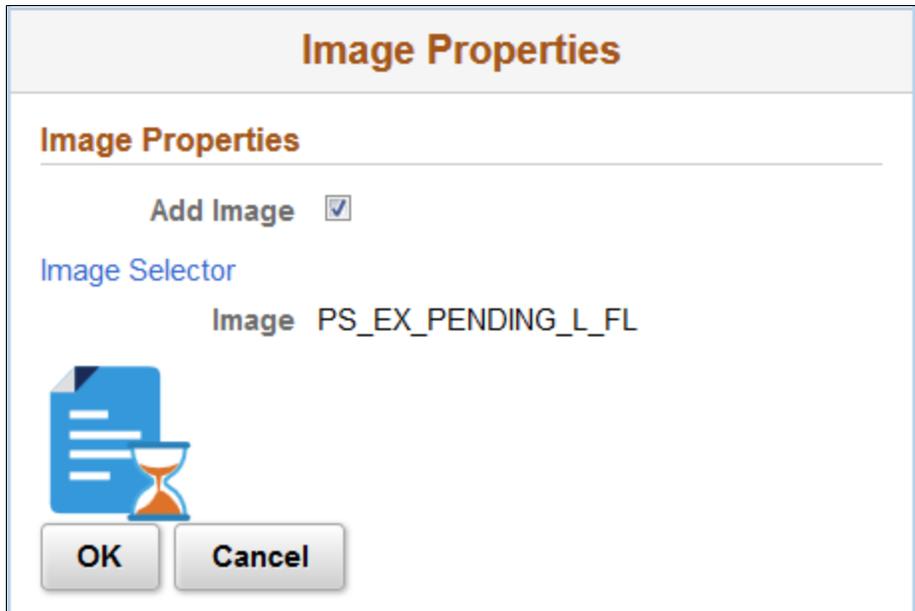
Any text for the image can be added on the layout grid directly.

Image: Example of image placed on layout grid with text

This example illustrates an image placed on layout grid with text.

Managing Mobile Grid Elements in Layout Rapid App Designer

Mobile grids let you position field types next to one another in rows and columns.

Mobile grids are useful for smaller element types like buttons and icons. They are not recommended for large items or chunks of copy, since mobile device resolution is much lower than a desktop and usability is extremely diminished.

You can insert the following Mobile Application Platform page elements into a mobile grid element in Layout Rapid App Designer:

• Button.

• Image/static text.

• Input.

• Static text.

- URL.

Note: If the mobile grid is defined on the header or footer, only URL, Button, and Container are allowed. No nested containers are allowed in a Mobile Grid

---

Defining Grids Page Elements

To define a mobile grid element you use the Layout grid toolbar on the Layout Rapid Application Designer to insert it into the layout page. At the time you define/add a mobile grid page element to your application, there are no other tasks to perform or parameters to specify. After you add the element to a layout grid, use the Mobile Grid Properties page to define styles and options for the page element.

To add a mobile grid to a page in Layout Rapid Application Designer:

1. Access the Layout grid on the Layout Rapid App Designer page.

   Select PeopleTools >Mobile Application Platform >Layout Rapid App Designer. to access the page.

2. Determine the position in the grid to add the page element.

3. Click the row directly above where you want to add the page element to make it the current row.

   The Current Row icon (blue arrow) displays in the first column of the row to indicate that it's the current row.

                                        The Current Row button.

4. Click the Add Mobile Grid button on the Layout grid toolbar.

                                        The Add Mobile Grid button.

5. Selection of the Add Mobile Grid will create a start and end Mobile Grid. The developer can then add any other Element Types within the start and stop icons

   Image: Mobile grid element inserted into the Layout grid

   This example illustrates the fields and controls on the Mobile grid element inserted into the Layout grid. You can find definitions for the fields and controls later on this page.

Defining Mobile Grid Page Element Properties

The Mobile Grid Properties section is where you define the number of columns and rows in the mobile grid .

To define properties for a mobile grid element, click the Mobile Grid starticon in the layout grid.

Image: Mobile Grid Properties page

This example illustrates the fields and controls on the Mobile Grid Properties page. You can find definitions for the fields and controls later on this page.

By default, two columns and one row are defined in a mobile grid. A minimum of two columns must defined and a maximum of five columns is allowed. A minimum of one row must be defined and there is no maximum for rows.

After you define the columns and rows click OK.

Note: The number of columns * number of rows must equal the total number of element types added to the Mobile Grid.

Insert your elements into the mobile grid in the layout page.

Image: Example of mobile grid with 2 columns and 1 row added to the layout in Layout Rapid Application Designer.

This example illustrates a mobile grid with 2 columns and 1 row added to the layout in layout Rapid Application Designer. Each column contains a static text field.

Managing Static Text Elements in Layout Rapid App Designer

To add static text page element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row above where you want to add the static text.

4. Click the static text icon.

5. The Static Text Properties page is displayed.

Image: Static Text Properties page

This example illustrates the fields and controls on the Static Text Properties page. You can find definitions for the fields and controls later on this page.

| Use Label Text | Select this checkbox to use the label text as defined in the layout grid. When you select this checkbox, the other fields on the page are hidden. Click OK to accept this change and enter the label in the layout grid. |

Message Catalog	Click the radio button to select Message Catalog for the static text type.
Message Set Number	Enter a message set number or click the Lookup button to search for one.
Message Number	Enter a message number or click the Lookup button to search for one.
Message Catalog Type	Choose the part of the message catalog enter to use: • Use Message Text. Select the radio button to use the text that appears in the Message Text field. • Use Message Description. Select the radio button to use the text that appears in the Description box.

Modifying Static Text Properties

You can edit the static text properties, as well as change the size of the text after the Static Text element has been added to the grid.

To modify static text properties, click on the Static Text icon in the layout grid.

Image: Example of changing the heading text

This example illustrates an example where the heading text for a static text field using label text is set to h2.
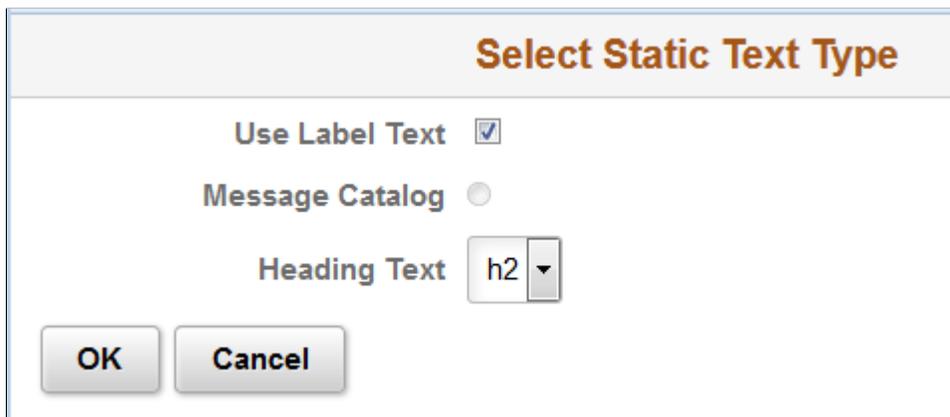
You can modify the static text type, as well as add heading text. Heading text h1–h6 are available from the drop down list.

Managing URL Elements in Layout Rapid App Designer

To add URL page element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row above where you want to add the URL.

4. Click the URL icon.

⚡	Add URL

5. The URL Properties page is displayed.

6. Enter the URL.

7. Optionally you can select to add an image.

Image: URL Properties page

This example illustrates the fields and controls on the URL Properties page.

Managing Button Elements in Layout Rapid App Designer

Either links or buttons can be used to navigate between mobile pages or link to external content. However buttons provide a better interface for the user on a mobile device than a link to tap. The button provides a larger clickable area with text and optionally an icon.

To add URL page element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row above where you want to add the URL.

4. Click the Button icon.

Add Button

5. The Button Properties page is displayed.

6. Enter the URL.

7. Optionally you can select to add an image.

Modifying Button Properties

To modify button properties, click on the Button icon in the layout grid.

Image: Button Properties page

This example illustrates the fields and controls on the Button Properties page. You can find definitions for the fields and controls later on this page.

Inline Button	Select the property so that the length of the button label determines the button width.
Remove Corners	Select the property to remove the border radius around the button.
Remove Icon Shadow	Select the property to remove the shadow around the icon.
Remove Shadow	Select the property to remove the shadow around the button.
Mini	Select the property to create a more compact version of the button.

Managing Select Menu Elements in Layout Rapid App Designer

Select Menu is used to create a drop down list in Layout Rapid App Designer. Values for the drop down list can be defined as input values or created based on a query.

To add a Select Menu element:

1. Select PeopleTools, Mobile Application Platform, Layout Rapid App Designer.

2. Open an existing layout or add a new layout.

3. Position your cursor on the row above where you want to add the menu.

4. Click the Select Menu icon.

                                      Select Menu

5. The Select Menu Properties page is displayed. You will need to define the primitive. The primitive data types supported are:

    •   Date

    •   DateTime

    •   Decimal

    •   Integer

    •   String

    •   Time

Image: Select Menu Properties – Add primitive

This example illustrates the fields and controls on the Select Menu properties for a string primitive. You can find definitions for the fields and controls later on this page.

Type	Select the primitive type.
Length	For a string, decimal, or integer, enter a length. A decimal will also require decimal length.
Description	Enter a description.
Element Name	The element name is read only.

6. Click OK, you will be prompted to save. Save the layout.

7. The Select Menu Properties for the primitive is displayed.

Image: Select Menu Properties – Input Type

This example illustrates the fields and controls on the Select Menu Properties – Input Type. You can find definitions for the fields and controls later on this page.
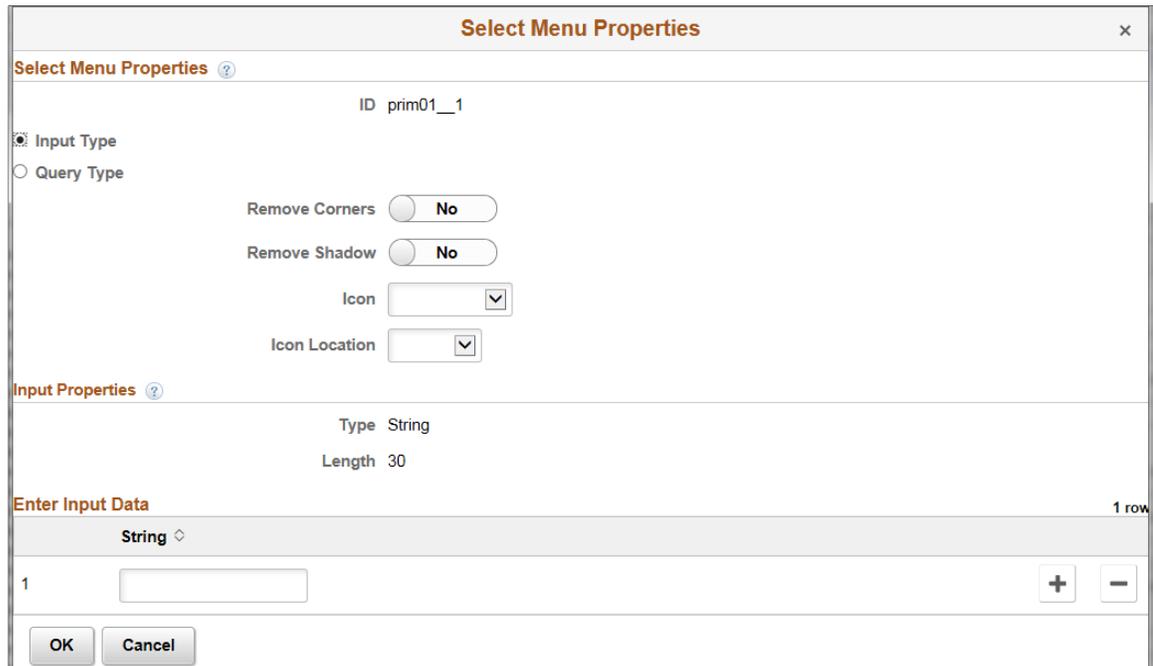
When you select Input Type, you will need to enter the input data manually.

Image: Select Menu Properties – Query Type

This example illustrates the fields and controls on the Select Menu Properties page for Query Type. You can find definitions for the fields and controls later on this page.
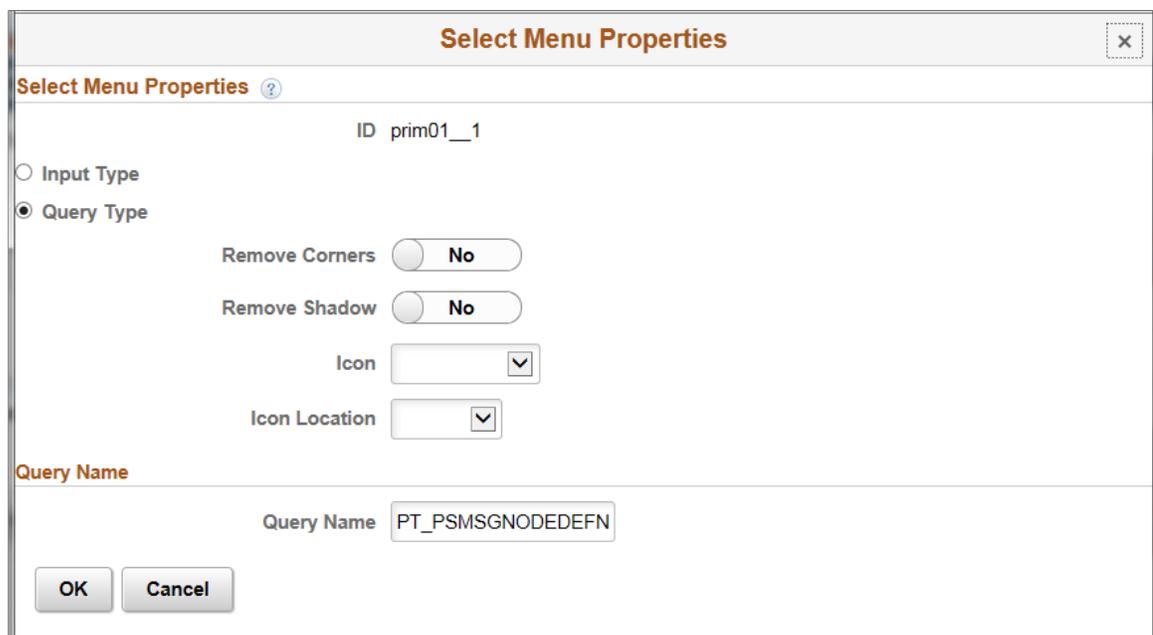
When you select Query Type, you will need to enter the query name. The query must return values that match the defined primitive.

Input Type	Select Input Type to manually enter the input data for your drop down list.
Query Type	Select Query Type if you are using a query to select the values. You will need to enter the query name.
Remove Corners	Select to remove corners
Remove Shadow	Select to remove shadow
Icon	Select the icon to display
Icon Location	Select the icon location.

- Bottom

- Left

- No Text

- Right

- Top

8. Click OK.

9. Save the layout.

Setting Permissions in Layout Rapid App Designer

After you save your layout in Layout Rapid App Designer, the Set Permissions and Launch Applications links are displayed. You must set permission before you can test the application.

Use the Set permissions (IB_MAPRAACCESS) page to set the permission list for the application.

Navigation

Click the Set Permissions link from the Layout Rapid App Designer.

Image: Set Permissions page

This example illustrates the fields and controls on the Set Permissions page. You can find definitions for the fields and controls later on this page.

Cancel	Set Permissions	Apply	Done
Assign Access			1 rows
Permission List		**Access**	
1 ALLPAGES 🔍		Full Access ▾	➕ ➖

Permission List	Add the appropriate permission lists for the application.

Access	Set the access level.
Apply	Click Apply to set the permission.
Done	Click Done to return to the layout Rapid App Designer page.

Launching the Application

After you save the application, the Launch Application link is available. When you click the Launch Application link, it will bring up a new page in the browser with the generated URL. You can bookmark the generated link or copy it.

By default, the MAP Application created will be defined as being in Production Mode. Therefore any runtime errors will not show the details of the error. As with any MAP layout created one can change the mode via the MAP administration page.

Using PeopleSoft Delivered MAP Layouts and Applications

PeopleSoft delivers MAP applications and layouts for PeopleTools. These applications can be used as delivered, however they can also be used with Rapid App Designer to create your own customized MAP application.

There are 3 groups:

1. PT_PEOPLETOOLS_ADMIN

2. PT_INTEGRATION_GATEWAY

3. PT_INTEGRATION_NETWORK

To view the delivered applications and layouts in the MAP Store, select PeopleToolsMobile Application PlatformMAP AdministrationMAP Store

Select the group and click Search.

Image: MAP Store Search by Group

This example illustrates the applications and layouts available in the MAP Store for the PT_PEOPLETOOLS_ADMIN group.

Using MAP Store Application

To launch a MAP Store Application, you can either:

- Launch from URL.

    1. Select PeopleToolsMobile Application PlatformMAP AdministrationMAP Store

    2. Search for the application.

   3. Copy the URL from the Map Store Search page and paste it into a browser.

The MAP signon page is displayed. After you sign on, the application will open.

• Launch the application from Layout Rapid App Designer

1. Select PeopleToolsMobile Application PlatformMAP AdministrationLayout Rapid App Designer

2. Search for and select the application.

3. Click either Launch or Secure Launch button.

The application is launched from PIA, so there is no MAP signon page.

Image: Tools Administration MAP Application

This example illustrates a section of the Tools Administration MAP application.
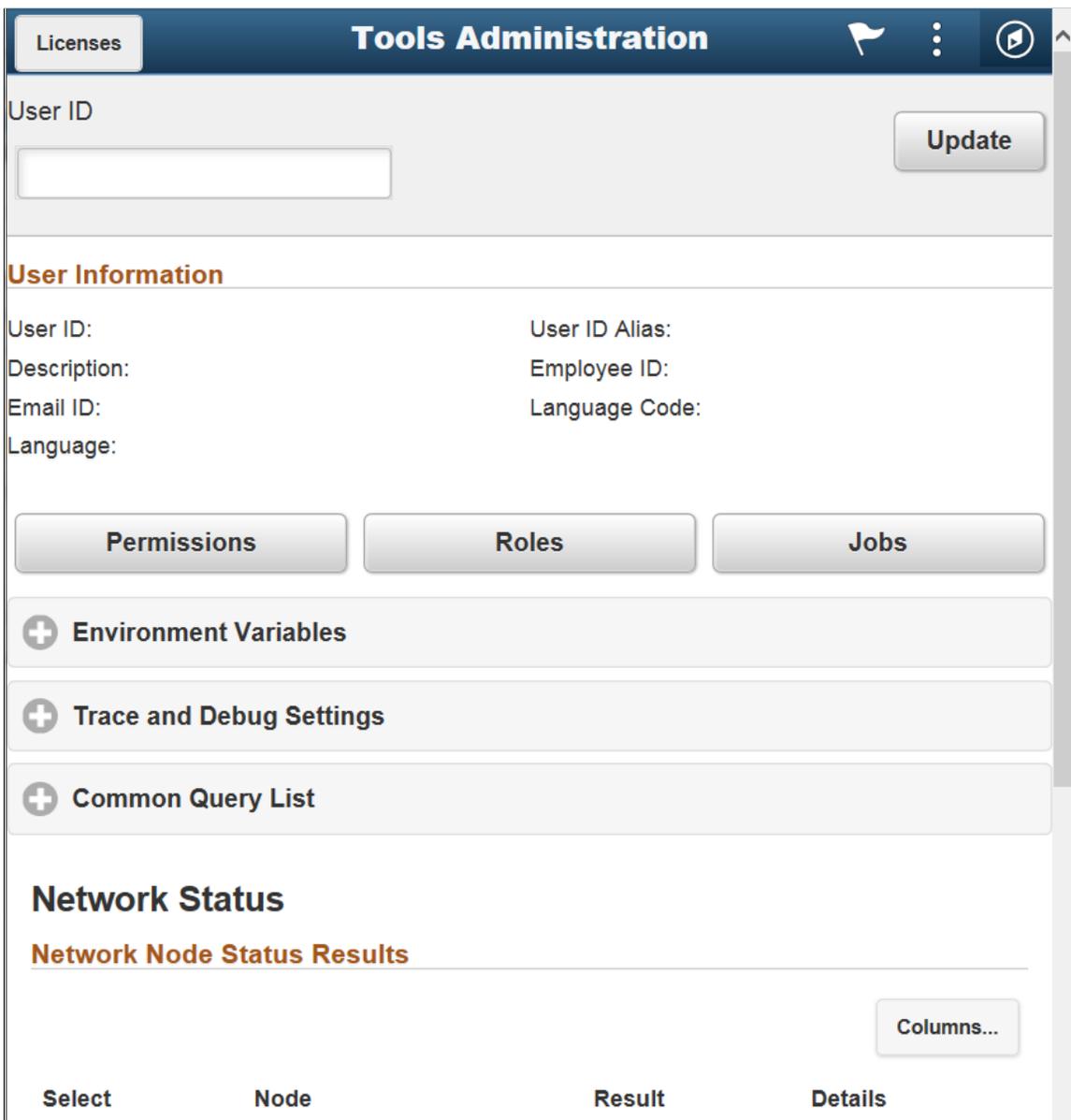
As delivered, this MAP application uses all of the MAP layouts in the PT_PEOPLETOOLS_ADMIN group. You can use the application as is or to review all of the layouts that are available and create your own application using Layout Rapid App Designer

Creating Your Own MAP Application Using Delivered Layouts

For example, you have decided that you would like to create a MAP Application that you can use to set debugging for MAP applications. Delivered in the PT_PEOPLETOOLS_ADMIN group are the layouts for tracing PeopleCode and SQL in MAP.

To create the new application for setting Debugging flags:

1. Select PeopleToolsMobile Application PlatformMAP AdministrationLayout Rapid App Designer

2. Click Add a New layout link.

3. Enter the Layout Name, and indicate whether or not you want to include a PeopleSoft header.

   Image: Enter New Layout Name

   This is an example of the Enter New Layout Name page for this application.

4. Click OK.

5. For this example, the layouts to use are in the PT_PEOPLETOOLS_ADMIN group and the layouts start with Trace. Enter the search criteria and click Search.

Image: Example Search for Layouts

This example illustrates the search used for this application.
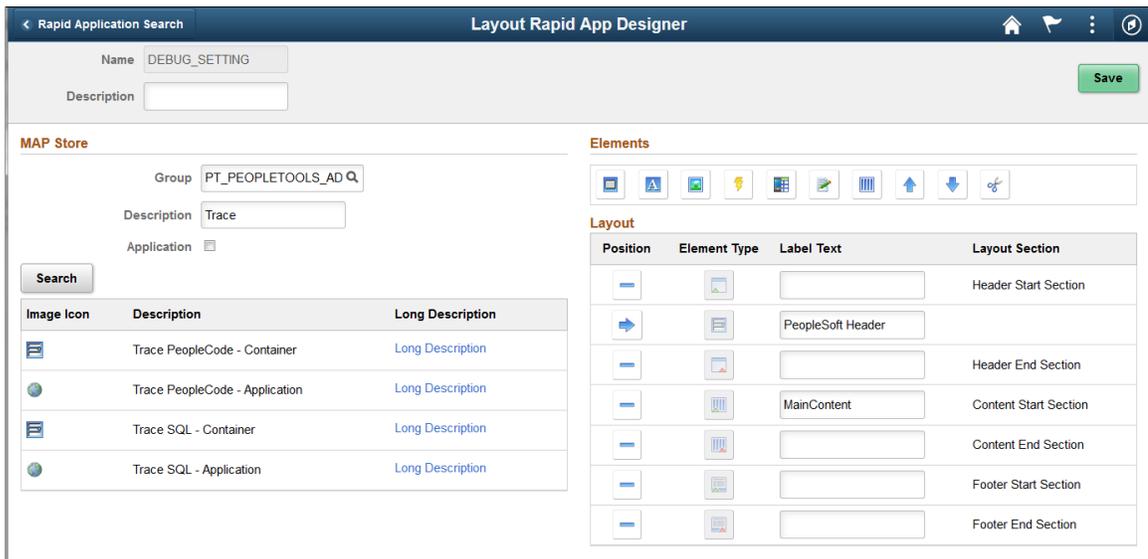
6. To add the content to the new application, position the cursor on the container for MainContent. Click the Container button to add a Container.

7. Change the container label to *MAP Debug and Trace*.

8. Click the icon for Trace PeopleCode – Container.

9. The URI mapping page opens, select *Default Initialization — no parameters required* and click OK.

Image: URI Mapping page

This is an example of the URI Mapping page with Default Initialization — no parameters required selected.
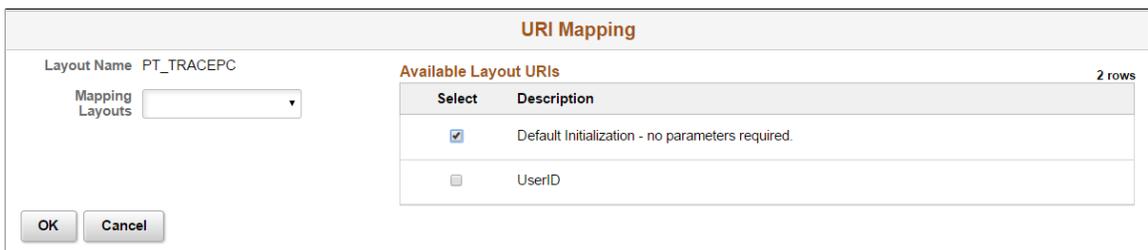
10. Click the icon for Trace - SQL Container.

11. On the URI mapping page select *Default Initialization — no parameters required* and click OK.

Image: Application layout

This is an example of the layout.

12. Click Save.

13. Click the Set Permissions button.

14. Select a permission list that your signed on user ID has access to and click Done.

15. Click Launch Application

16. This is an example of your new MAP application.

Image: New MAP Application

This is an example of the MAP Debug and Trace application that you just created.