



# Using Design Time Images in PeopleTools

PT 8.58

Randall Groncki

## Introduction

PeopleTools has the power to deliver an image rich, modern UI for your enterprise. Whether it's branding, employee photos or the millions of images for the catalog's widgets, we can deliver the modern UI you need within the PeopleSoft architecture.

There are two primary image categories in PeopleSoft: Design Time images and those images uploaded by users. This document will specifically cover the Design Time images.

Design time images are those uploaded into the App Designer when creating and modifying pages. These images are managed objects and migratable. Though extensively used by PeopleSoft, the image handling objects and documentation really haven't been updated since the original version 8 tools debuted in 2000.

Using Design Time images in fluid pages will be addressed in its own document and not in the scope of this document.

## Viewing design images currently on your system

PeopleSoft delivers thousands of images with the different applications. These are stored in the system as Image Objects in the Content Tools tables

- PSCONTDEFN
  - Content Definition Table
  - Look for ContType = 1 for images
- PSCONTENT
  - Table that holds the actual images
  - Also: scripts, XML and CSS (ContTypes 4 & 9)

Unfortunately, there is no one delivered place to view the available images in either PeopleTools or the online application. Some of the tools do provide a preview of the image AFTER you have chosen the image to use, but nothing to aid in selecting the correct image other than the descriptive name.

We have created a custom component to view all design time images. Download the image viewing component at our GitHub site: <https://github.com/PeopleToolsTechTips/PT3-E011-Design-Time-Images>

## Design Time Images










\*Image Type  

Image Name  

### Images

	Image Name	Format	
1881	PS_SUBMITTED_APPLICATION_ICN	JPG	
1882	PS_SUBMITTED_APPLICATION_S_FL	svg	
1883	PS_SUCCESSION_OPTIONS_32_ICN	gif	
1884	PS_SUCESSOR_ICN	gif	
1885	PS_SUCCESS_L_FL	svg	
1886	PS_SUCCESS_RES_S_FL	svg	
1887	PS_SUCCESS_S_FL	svg	

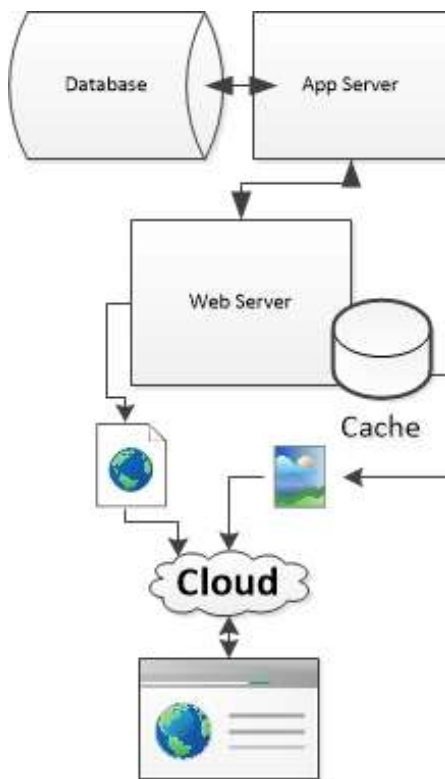
## How PeopleTools delivers images to the browser

A primary challenge with images is that the application does not provide the image to the user's browser in the HTML as it can with most other elements such as text, JavaScript and CSS. Rather the application places the image in a known location of the web server and provides the browser a link to that location.

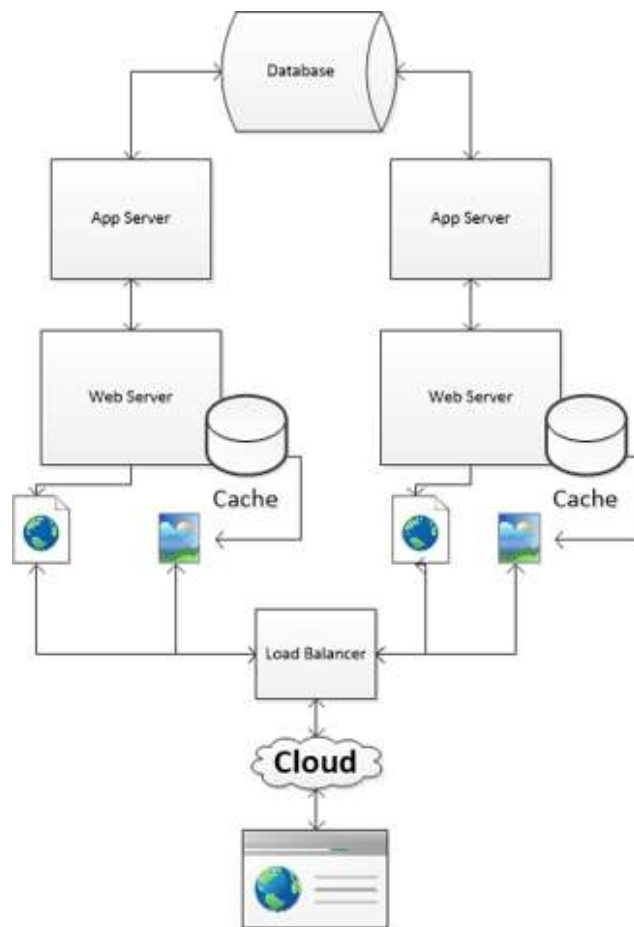
If the architecture contains multiple Web and App servers behind a load balancer, we can't know ahead of time where that image is going to be for the user... and the next user after that.

## Browser gets Images and objects from the Web Server's Cache

Simple Network – Single Web Server



Complex Network - Multiple Web Servers



When an image is called for at runtime, the App Server places the designated image in the web server's cache and then provides a link to that image in the HTML delivered to the user's browser.

As developers, we never need to pre-stage images on web or file servers. This is bad practice since it would create a versioning and ongoing maintenance tasks. Instead, the architecture handles image management for us.

## Primary Image Types

In PeopleSoft HCM Application on PeopleTools 8.58, there are 8 distinct types of images in the content table:

Type	Format
BMP	Windows Bitmap
DIB	Device-Independent Bitmap
GIF	Graphics Interchange Format
JPG	Joint Photographic Experts Group
PNG	Portable Network Graphics
SVG	Scalable Vector Graphics

Many of the images and formats are artifacts of PeopleSoft applications from many years (decades) ago. Today, we commonly use three image formats:

Format	Common Use
PNG	Best for images with large, uniformly colored areas. Generally, computer generated images such as logos and page graphics. *Lossless compression.
JPG	Common for photographs * Lossy Compression
SVG	Scripted (XML) for web graphics Preferred image type for Navigation Tile Icons *No Compression

Fun image search: find the different PeopleSoft and Oracle logos throughout the years still in the image catalog. The original PeopleSoft logo is named "PTNUI\_TEMP"



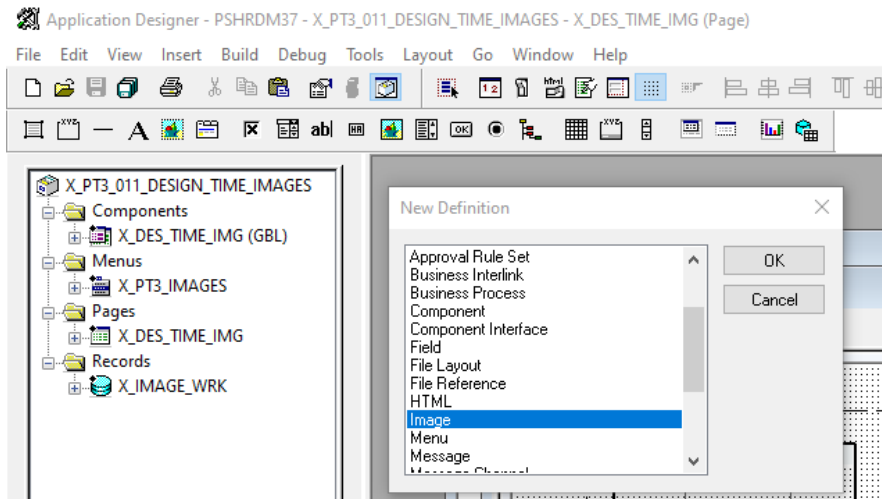
## Uploading new images

Design Time images are PeopleTools managed Objects.

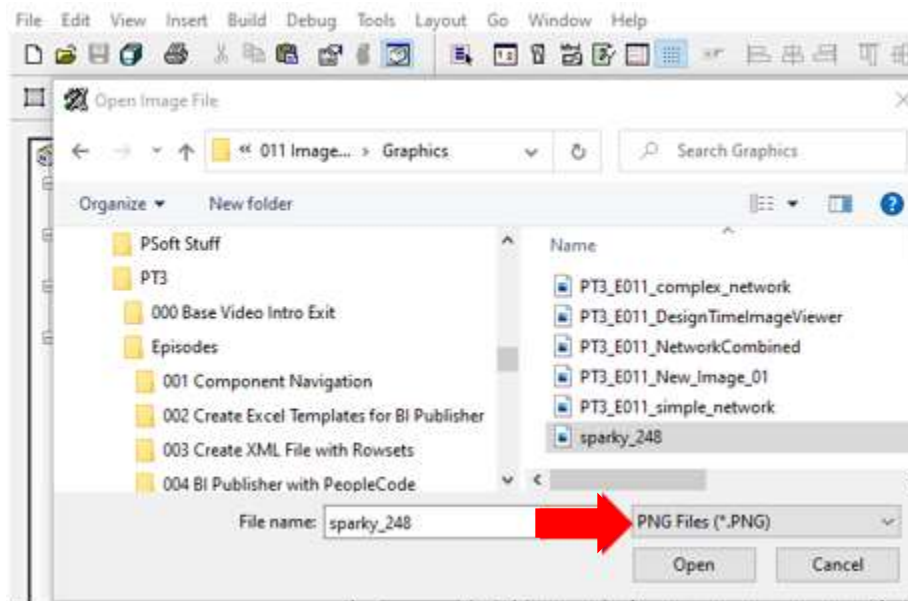
We add them in the App Designer. For this example, we will add an image for “Sparky”, which we found on the web.

First, acquire the image and place it somewhere convenient on your machine.

In App Designer, click on File > New, and then choose image as the definition type:



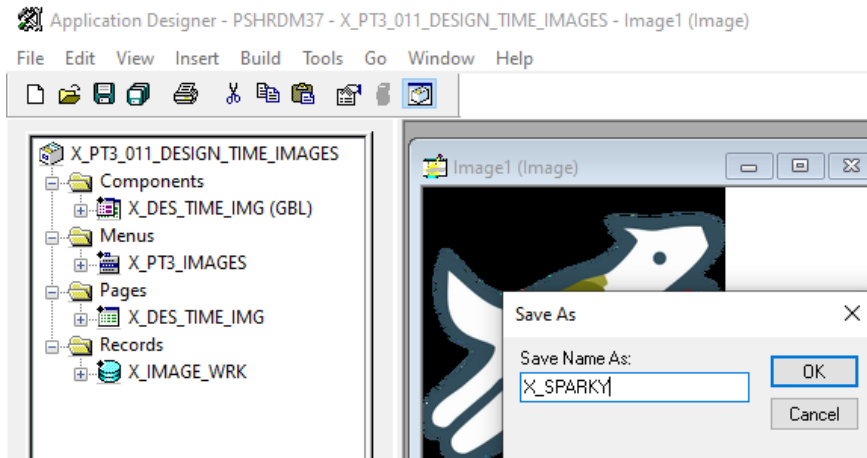
Find your image. Ensure the file type is correctly set to your image type. It defaults to JPG.



Your object will open in the workspace with a default name. Some of the image formats (PNG, JPG) will display in the object. Sometimes the image will not display should it be too large or a format unsupported for display in the App Designer (SVG).

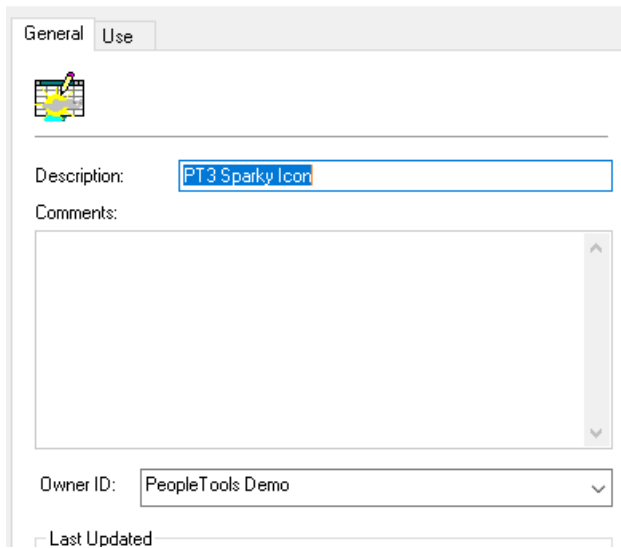
If the image is larger than 32K, the system will generate a warning. Some DB systems will not support images larger than 32K.

Save your new image object: File > Save

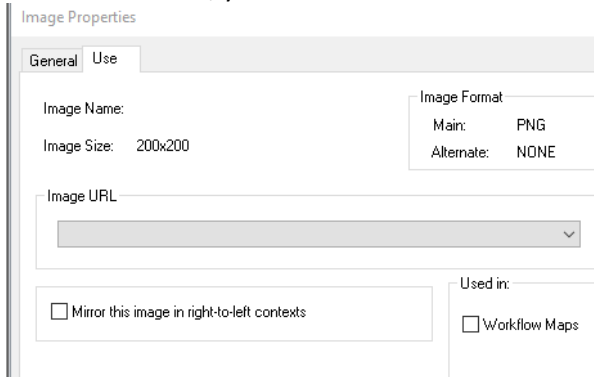


The system will prompt you for the image properties

Image Properties



On the "Use" tab, you can see more details about your image



## Image Use Controls

### Image URL

Store the image on a file server rather than in the database. This URL points to the image location for the system to retrieve at run time.

### Mirror this image in right-to-left contexts

Flip the image horizontally in languages that read right to left.

## Placing images on a page

*Fluid page images are outside the scope of this document. Fluid images will be addressed in its own paper and how to video. This document specifically speaks to Classic and Classic Plus pages.*

There are four primary ways to place an image on a page

1. Static Image Page Control
2. HTML Area with image declared as a constant value in the properties dialog box
3. HTML Area with image declared dynamically through PeopleCode
4. Button

## Static Image Page Control

Add a static image to the page where you want your image

Open the Properties box for the Static Image

### Label Tab

The screenshot shows the 'Static Image Properties' dialog box with the 'Label' tab selected. The dialog is divided into three main sections: 'Alt tag for image', 'Image', and 'Image Format / Size'.  
1. 'Alt tag for image': Contains a 'Label Type' dropdown menu set to 'Text', 'Message Set' and 'Msg Number' input fields, and a 'Label' text box containing 'Informative Label'.  
2. 'Image': Contains an 'Image ID' dropdown menu set to 'X\_SPARKY', a 'Type' dropdown menu set to 'PNG', and 'Width' and 'Height' input fields both set to '200'. Below these fields is a small thumbnail image of a white dog with yellow and black markings.  
3. 'Image Format / Size': Contains radio buttons for 'Scale' (selected) and 'Size', and 'Width' and 'Height' input fields both set to '75'. A 'Reset to actual image size' button is located below these fields.  
An 'OK' button is located at the bottom right of the dialog box.

The "Alt Tag" group box controls the hover text for the image.

The “Image” Group Box chooses the image object using the “Image Id” field. The Type and size fields will default from the selected image object properties. These are not changeable. This image will display if it is a type the App Designer is able to show.

The “Image Format / Size” group box allows the designer to change the size of the image on the page.

- Scale allows to change the size of the image on a scale.
- Size presents the image in at its native dimensions (width & height do not apply)

## HTML Areas

The “%Image ()” HTML metatag is used with HTML areas to pull the image into the control at runtime.

The App Servers will use this metatag to place the correct image in the Web Server cache and replace the metatag with the correct html pointer to that image.

In the example of our Sparky image object named “X\_SPARKY”, you would reference like this:

```
<img src=%Image (X_SPARKY) />
```

Since it is an HTML area, you can use additional styling parameters.

```
<img src=%Image (X_SPARKY) width=75px; height=75px/>
```

## HTML Area with image declared as a constant value in the properties dialog box

Add an HTML Area to the page where you want your image.

Open the control Properties and select the “HTML” Tab

In the “Value” group box, choose “Constant”.

Put the HTML reference using the “%Image ()” metatag in the dialog box.

HTML | Label | General | Fluid

Value

Field  Constant  HTML Tags Only

Record Name:

Field Name:

```
<img src=%Image(X_PT3_BADGE) width=75px; height=75px/>
```

Respond Only Once



## HTML Area with image declared dynamically through PeopleCode

Add an HTML Area to the page where you want your image.

Open the control Properties and select the "HTML" Tab

In the "Value" group box, choose "Field"

Place a Record and Field in the corresponding boxes. This is most often a Derived Work Record field.

HTML Area Properties

HTML Area Properties dialog box showing the 'HTML' tab selected. The 'Value' group box contains the following options:

- Field
- Constant
- HTML Tags Only

Record Name: X\_IMAGE\_WRK

Field Name: HTMLAREA

Respond Only Once

Using PeopleCode, place the HTML in that Record.Field value for the image.

```
&Rec_X_IMAGE_WRK.HTMLAREA.Value = "<img src=%Image(X_PT3_BADGE) width=50px;  
height=50px/>";
```

The image can be dynamically changed at runtime to reflect information or changes in the system.

## Using Images in a button

Button controls can use images instead of labels to convey more meaning or visual shortcut in the UI.

Add a button to your page. Fill out the “Type” tab in the Properties dialog box

The screenshot shows the 'Push Button/Hyperlink Properties' dialog box with the 'Type' tab selected. The 'Type' section has three radio buttons: 'Push Button' (selected), 'Hyperlink', and 'Action Widget'. Below this are four dropdown menus: 'Destination' (PeopleCode Command), 'Record Name' (X\_IMAGE\_WRK), 'Field Name' (VIEW\_PB), and 'Style' (Use Default Style).

On the Label Tab Choose “Image” for Label Type

The screenshot shows the 'Push Button/Hyperlink Properties' dialog box with the 'Label' tab selected. The 'Label Type' section has three radio buttons: 'Text', 'Image' (selected), and 'Both'. The 'Label Size' section has four radio buttons: 'Small Image', 'Large Image', 'Standard Text', and 'Custom' (selected). The 'Alt tag for image' section has a 'Type' dropdown (RFT Long), a 'Text' field (View Status and Results), a 'Label ID' dropdown (Use Default Label), and a 'Context' field. The 'Label Image Format / Size' section has two radio buttons: 'Scale' (selected) and 'Size', and two input fields for 'Width' (34) and 'Height' (34), along with a 'Reset to actual image size' button. The 'Label Image' section has an 'Image ID' dropdown (PT3\_BADGE), 'Width' (400) and 'Height' (400) input fields, and an 'On Mouse Over' dropdown.

In the “Label Image” group box on the right side of the properties box, choose the image in the “Image ID” field. The width and height fields will default from the chosen image object’s properties.

In the “Label Size” group box, choosing “Small Image” or “Large Image” will have no effect on size in Classic and Classic Plus rendering. The image will render on the page at its native size. For general reference, the small button is supposed to be 16x16 pixels while the large button is 32x32 pixels.

Choosing “Custom” label size and then using the “Scale” control on the right side of the Properties box will allow you to render the image at a different size than native.

The “Small Image” and “Large Image” controls along with the “Both” option for label type to work with fluid pages.

The “Alt tag for image” group box allows you to define the hover text for the button.

## Tile Images

Both the Tile Wizard and the CRef Fluid Page require SVG images for Tile icons. Each has a prompt view that limits the user to only images uploaded as type SVG. Attempting to place another image object that is not type SVG will result in a page error.

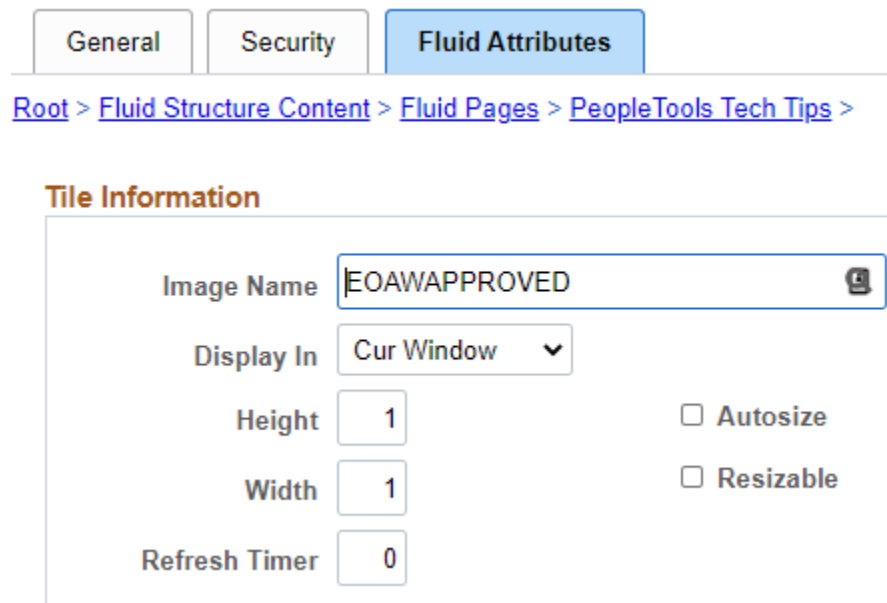
The Tile packages will resize an image that is too large for a tile to fit the given tile space. It will not increase the size of a small image to fit the tile size.

Since an SVG image is just an XML File, you will need a Graphics Software that can create or convert images to type SVG such as Adobe Illustrator. MicroSoft Paint does not have the ability to create or edit SVG images. There are online web-based utilities to convert PNG, GIF and other image formats into SVG files with varying degrees of success.

AConvert.com ([PNG to SVG - Convert image online \(aconvert.com\)](http://aconvert.com)) seemed to work fairly well.

Once you have your SVG Image, upload it into App Designer just as you would any other image. It is then available to the Content Refs as a tile image.

Use the Structure and Content component or the Tile wizard to add an image to your tile.



The screenshot shows the 'Fluid Attributes' tab selected in a software interface. Below the tabs is a breadcrumb trail: [Root](#) > [Fluid Structure Content](#) > [Fluid Pages](#) > [PeopleTools Tech Tips](#) >. The main section is titled 'Tile Information' and contains the following settings:

- Image Name:
- Display In:
- Height:   Autosize
- Width:   Resizable
- Refresh Timer: