Wcode Manual

By Bryan Goldstein

widgetfactor@gmail.com

Table of Contents

Creating a new widget

Using the template system

When first opening the template system you are prompted with a panel that asks you what type of widget you would like to make. You can find and install more templates online, google "Wcode Templates" or something similar. The templates that are packaged with Wcode are Countdown Widget, Default Widget, Flash Game or Movie Widget, RSS Widget, and Search Widget. Now I will describe how to put each of these templates to use and which one to choose.

Creating an RSS widget

The most popular template to choose is the RSS template. (note: the RSS template is not what is being sold in Wcode and is available for separate, free download, and thus not violating the creative commons license on code used to create it.) The RSS template creates a widget that gets news headlines from the news sites of your choice. The RSS widget also allows you to search a website too just like the search widget template mentioned later.

To start creating an RSS widget, select "RSS Widget" and click ok. Now fill in a name, display name, your email address, your company name, your website address, and find the images you may want to use. You can leave any field you feel is unimportant blank. In order to have your widget check a news site you need to find an RSS feed of that news site. Then copy the RSS feed's URL into the feed URL field. Now you need to make a name for this feed, type the name in the field below the URL field. Once you have done both of those things you can click add button and it will be added. You can add as many feeds as you like. This part throws some people off so pay attention. In order to allow your widget to search a website, find a site you that you want your widget to search. Now search that site for something such as *test*, once you have searched the word *test* take a look at the URL of the site it took you to. Now copy the section of the URL before the word you searched (test) and paste it in the Search URL Prefix field then copy everything after the word and put it in the Search URL Suffix field. You can also add a placeholder to the Search Placeholder field,

which is the text in the search box when there is nothing else in it. Finally, press ok again. Note: anything that is left blank will be given its default value and your widget will still be created. Now you have the code for your RSS widget.

Creating a search widget

To start creating a search widget, select "Search Widget" and click ok. Now fill in the widgets name, your email address, and your website name. Now find any images you would like to use. This part throws some people off so pay attention. In order to allow your widget to search a website, find a site you that you want your widget to search. Now search that site for something such as *test*, once you have searched the word *test* take a look at the URL of the site it took you to. Now copy the section of the URL before the word you searched (test) and paste it in the Search URL Prefix field then copy everything after the word and put it in the Search URL Suffix field. You can also add a placeholder to the Search Placeholder field, which is the text in the search box when there is nothing else in it. Finally, press ok again. Note: anything that is left blank will be given its default value and your widget will still be created. Now you have the code for your search widget.

Creating a countdown widget

To start creating a countdown widget, select "Countdown Widget" and click ok. Now fill in the widgets name, a display name, and pick the images you would like to use. Use the calendar to pick a date. Now click ok and you have the code for your countdown widget. Remember that if you leave fields blank they will be replaced with a default value.

Creating a flash widget

To start creating a flash widget, select "Flash Game or Movie Widget" and click ok. Now fill in the name and pick the Images you would like to use. Then click the browse button under Flash Game/Movie, select the flash file you want your widget to play. Press ok again and you have your flash widget. Remember that if you leave fields blank they will be replaced with a default value.

Creating an empty widget

To start creating an empty widget, select "Default Widget" and click ok. Now fill in the name and pick the Images you would like to use. Press ok again and you have your widget. Remember that if you leave fields blank they will be replaced with a default value.

Installing and creating a widget from a downloaded template

Find a template from the Internet. A good way to do this is either to see if there are extra templates available at www.widgetfactor.com, or search google for Wcode templates. Once you have the template downloaded double click it and Wcode will open and ask if you would like to edit or install this template, click install. The template should now appear in the list with all the other templates. Now fill out all of the forms to the best of your ability, and if the developer of the template did it right it should allow you to leave anything you want blank.

Editing a widget

Opening a widget

To open a widget you need to press open, command-o, File > Open, or the open toolbar item. Now find the widget you want to open and press open.

Editing files

First click on a text file in the file list to open it. Now it should open in Wcode's editor and you should be able to edit it. Now try clicking on an image file. With the image file you should be able to drag an image of your own in to replace the current one you can also paste one in. You can also drag the image out or copy it. The other thing you can do is open it in

another application for editing which is discussed in another section. Now click on a folder the folder will open in Wcode's file browser. Finally you will notice that the Info.plist file opens in a different editor. The Info.plist's editor will help you by giving the Info.plist's settings a more user-friendly approach.

Searching

In the toolbar there is a handy search box for searching the contents of files so you can get to the file and line your looking for quickly.   All you need to do is click into the editor you want to search and start typing.   The first result will be scrolled to and selected in all of your files.

Renaming Files

Renaming a file is a synch. Select the file you want to rename, right click on it, and select rename in the menu. Rename the file and press ok when you are done, or press cancel if you change your mind.

Opening in an external editor

To open a file in a different program select a file and right click it. Click Open With… and select the application you wish to open the file with. That's all, remember not to open the file in Wcode again until you are done editing it in the separate program, otherwise Wcode will write over what you have written in the other program.

Saving

To save just press command-s, or you can click the save button in the toolbar, or you can even go to File > Save. That's all you need to do. If you want to save as, go to File > Save as and choose where you want to save your widget.

Built in Widget Preview

When you want to quickly test your widget right in Wcode, you can just click the preview button in the toolbar and it will appear. It doesn't

work as well as running in dashboard but it does the job. Also, be careful when dragging it around.

Building and Running

Building and running is simple. Just click the Build and Run button in the toolbar, and after you are prompted about installing the widget, dashboard will run your widget. If you don't want that prompt every time you run your widget move the widget to your widgets folder and open it from there.

Creating your own template

How does the template system work?

The template system is the greatest part of the new Wcode. If you look in ~/Library/Application Support/Wcode/Templates, you will find all the templates that are installed on your computer. A template is just a widget with an xml file called template.plist. In order to make your ordinary widget into a template you need to create this file. You must also change the extension from wdgt to wdtp in order to make your widget a template.

Creating a template.plist file

In your widget create a template.plist file, or copy a template.plist file from an existing template. You can make this file with property list editor or if you know how to make property list you can do it with a text editor. The root of the template.plist file should be an array. Each item in the array is a dictionary. Each one of these dictionaries defines the parameters for an option presented to the end user. All of these dictionaries should contain the three keys, TPVName, TPVTooltip, TPVFiles, and TPVType. The TPV stands for template variable, don't ask me why I made them like that. TPVName should contain a string with the name of the option given to the user. TPVTooltip should contain a string with the text that will be shown

when the user hovers their mouse over the input view. TPVFiles should contain an array of the files that have tags in them that the templates system should input the user-entered value script.js, or Scripts/script.js if it is in a directory called Scripts. Next, TPVType is a string with the type of input to present the user with. The options for TPVType are Text, Image, Array, Array2, Date, File, PopupButton, and Checkbox. Each one of these input types have their own parameters which I will go over in the next sections. After the template.plist file is complete, you need to define where the values go in each of the files you have pointed to. To do this in place of the value that needs to be entered by the user, in the file where it will go, use the tag Template_*TPVName* where *TPVName* is the TPVName of the variable to be associated with it.   The date type adds a little extra parameter to this tag that I will explain.

TPVType - Text

This is the simplest of the input types. The text input type gives users a nice big text field for them to enter whatever you ask them to enter. The only extra tag that needs to be added, which is required with all of the types except PopupButton and Date is TPVDefaultValue. TPVDefaultValue is the value that will be assumed if the user leaves the input blank. In this case the TPVDefaultValue is a string, but it is not a string with all types. You can find an example of this in any of the templates that are packaged with Wcode.

TPVType - Image

This input allows the user to choose an image by either dragging one in or selecting browser and choosing one. Again with this input the only extra tag that is needed is TPVDefaultValue. In this case TPVDefaultValue is a string pointing to an image that will be used if the user does not pick an image. Do not precede this value with a slash, for instance use default.png, or Images/default.png. The TPVDefaultValue can be the same as a value in the TPVFiles array. You can find an example of this in any of the templates that are packaged with Wcode.

TPVType – Array

Although there is no example of this type in the templates that are packaged with Wcode, it is very similar to the Array2 type seen in the RSS template. The array input will give the user a popup button, a text field, and add and remove buttons. This allows the user to create an array of values. Array2 is similar except the user is given two text fields instead of just one. This type also has a TPVDefaultValue, but instead of being a string like the others this TPVDefaultValue is just array. The default value is never shown to the user in all cases it is just used if the user leaves it blank. Another unique value you must give the array type is TPVFormat. When the template system injects the array into the widget rarely do you want it to just list the strings that the user inputs. Most of the time you will want code to be wrapped around what the user inputs. The way to use TPVFormat is type the code you want to be wrapped around each of the values in the array and simply put Template_Val where you want the value to go. That's all you need to do for the array type.

TPVType – Array2

You can find an example of this in the RSS Widget, it is very similar to the Array type which has no exampled packaged with Wcode. The array2 input will give the user a popup button, two text fields, and add and remove buttons. This allows the user to create an array of values. Array is similar to this except the user is given one text fields instead of two. Array two has the most complicated TPVDefaultValue, it is an array, containing two arrays. The first of these two arrays is the first field's default values and the second array is the second text field's default values. The default value is never shown to the user in all cases it is just used if the user leaves it blank. Another unique value you must give the array2 type is TPVFormat, it is similar to the array type's TPVFormat. When the template system injects the array into the widget rarely do you want it to just list the strings that the user inputs. Most of the time you will want code to be wrapped around what the user inputs. The way to use TPVFormat is type the code you want to be wrapped around each of the values in the array and simply put Template_Val1 and Template_Val2 where you want the first and second value to go. Now you have your very complicated array2 variable working.

TPVType – Date

The date input is one of my personal favorites. It isn't that complicated to use but pay attention or you might miss something. Date has no extra tags. It doesn't have a TPVDefaultValue. The only thing you need to know for the date type is the you need to mark it differently in the files that the user selected value is going into. I explained how you are supposed to tag where the values should go with Template_*TPVName*, well for the date type you need to add something a little extra. Template_*TPVName_x* where *x* is one of the following letters depending on what part of the date you want:

x	Description
a	Abbreviated weekday name
A	Full weekday name
b	Abbreviated month name
B	Full month name
c	Shorthand for "%X %x", the locale format for date and time
d	Day of the month as a decimal number (01–31)
e	Same as %d but does not print the leading 0 for days 1 through 9
F	Milliseconds as a decimal number (000–999)
H	Hour based on a 24–hour clock as a decimal number (00–23)
I	Hour based on a 12–hour clock as a decimal number (01–12)
j	Day of the year as a decimal number (001–366)
m	Month as a decimal number (01–12)
M	Minute as a decimal number (00–59)
p	AM/PM designation for the locale
S	Second as a decimal number (00–59)
w	Weekday as a decimal number (0–6), where Sunday is 0
x	Date using the date representation for the locale, including the time zone
X	Time using the time representation for the locale
y	Year without century (00–99)
Y	Year with century (such as 1990)
Z	Time zone name (such as Pacific Daylight Time)
z	Time zone offset in hours and minutes from GMT (HHMM)

Remember, you can put as many tags as you want in any of the files you specify. If you understood how to do this it is quite simple. And now you have a nice calendar input instead of asking the user to type the date.

TPVType – File

For the file input you can put two types of files into TPVFiles. You can put a directory, leave one of the files objects blank to specify to put the file at the base of the widget, or you can put a file in which case it will look for a tag in the file and replace the tag with a the filename of the chosen file. The TPVDefault value can be a file name or it can just be text saying place file name here or something like that. You don't need a default value for the file input type if you do not wish to put one. Another unique value that needs to go into the file input type is TPVFileTypes. TPVFileTypes is an array of file types that are allowed. For instance the flash template packaged with Wcode has a TPVFileTypes containing one the one string, swf, because that is the only file type it accepts. That is all you need to know for the file input type.

TPVType – PopupButton

There is no example of this in any of the templates packaged with Wcode. The popup button type allows the user to select from a list of different options. There is no TPVDefaultValue for the popup button type. There are two additional values that are needed though. You will need TPVOptions and TPVValues. These are both arrays, the TPVOptions are the options that are shown to the user and TPVValues are the values that correspond with the options. That's really it, now you can have a popup button in your template.

TPVType – Checkbox

There is no example of the checkbox input in any of the templates packaged with Wcode either. It is pretty simple though. The TPVDefaultValue is a string of either 1 or 0 where zero is unchecked and one is checked. Then just add a TPVCheckedValue and TPVUncheckedValue, which are strings containing the value when checked and when unchecked. That is all you need to know.

Still need help?

If you still need help with anything don't hesitate to email me. If you want help, guidance, or just support with creating a template please email me. I will be overjoyed to help you. I am counting on people to make templates, and hoping this will be what sets Wcode apart from the other widget making software. So make a template and it will be worth it every time you see a new widget that was made with your template.