BlackDog Foundry

- RSS

Enter search term   →

BlackDog Foundry

- LogDiver
- About

Bookmark This page

Creating an Xcode4 Plugin »

craig
19 Oct
Tutorials
objective-c, plugin, xcode4
7 Comments

Creating an Xcode4 Plugin

This tutorial describes the steps to create a very simple Xcode4 plugin project.

Note that there is no formal support for Xcode plugins, so this information may change and/or become obsolete as Apple makes changes to Xcode. Also, as it is unsupported, you will not be able to submit this plugin to the Mac App Store.

Overview

Although Xcode doesn't *officially* support plugins, it does have some neat behaviour that allows us to inject our custom plugins. On startup, Xcode looks for bundles with an extension of `.xcplugin` that are located in the `~/Library/Application Support/Developer/Shared/Xcode/Plug-ins` directory (if present).

In order to create our plugin, we will be creating an Bundle project and then installing it into that location.

Creating the Project

This post was created using Xcode 4.5.1, however, I believe it should be appropriate for any Xcode 4.x installation.

Create Bundle Project

Click on the *File/New/Project…* menu item, and choose a Bundle in the OS X Framework and Library section:
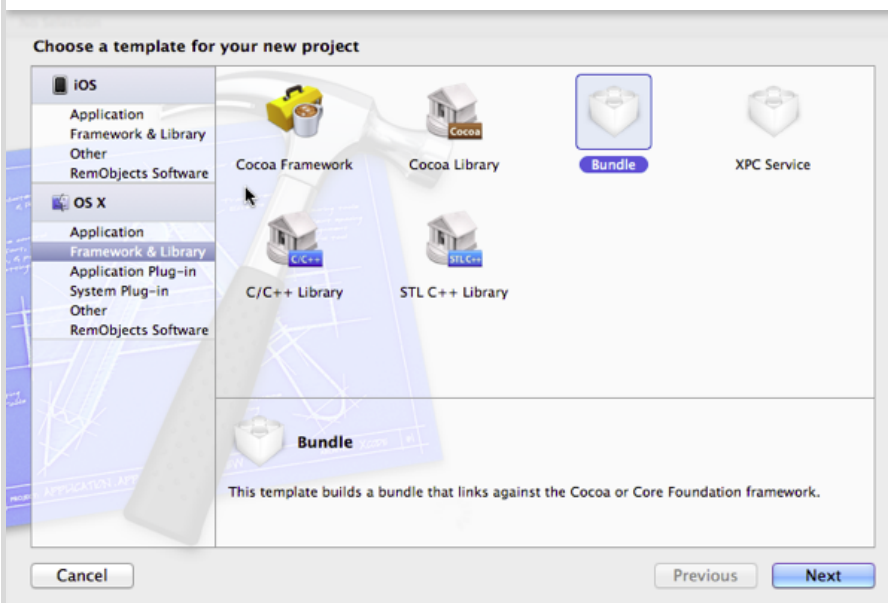
and set your project name and other information. Choose *Cocoa* as the framework that you want to link to, and then save your project.
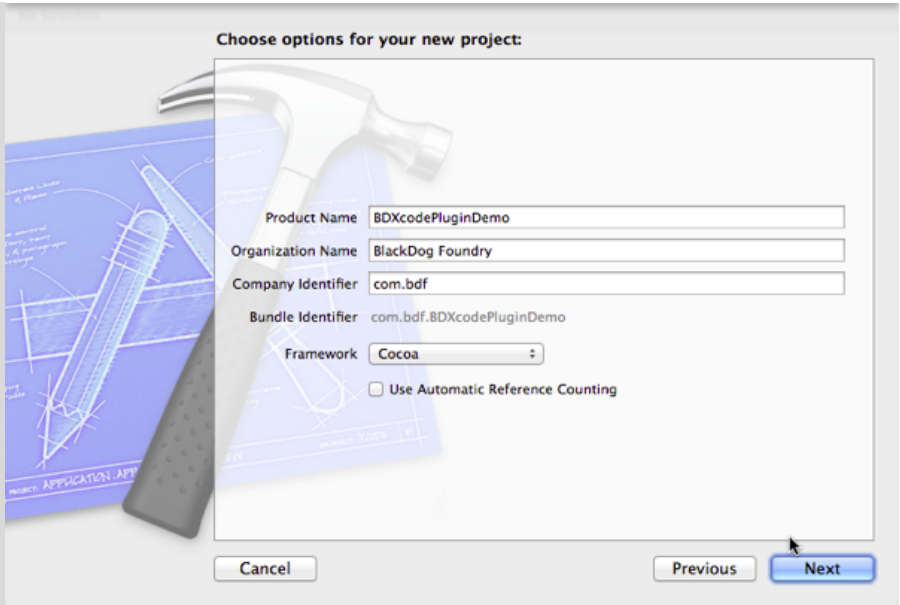
Modify Default Settings

Now that you have your basic project created, we need to clean up some things that we won't use, and change a few settings so that it will generate a valid Xcode plugin.

Remove the following frameworks as they are not needed:

- `Cocoa.framework`
- `CoreData.framework`

You now need to add three new Boolean entries to your `.plist` file:

```
XCGCReady     = YES
XCPluginHasUI = NO
XC4Compatible = YES
```

Next up, we want to change the installation directory. The rationale for this is that every time you re-build the plugin, it will automatically get updated in-place in its eventual home, so you can re-test by simply restarting Xcode. Technically, I don't think you need to do this, but it makes testing a lot easier.

Open up the Build settings for the `BDXcodePluginDemo` target and change the following settings:

Setting	Declaration	Value	Notes
Installation Build Products Location	DSTROOT	$(HOME)	
Installation Directory	INSTALL_PATH	/Library/Application Support/Developer/Shared/Xcode/Plug-ins	Note that it looks like the Installation Directory value is an *absolute* path but, in fact, Xcode appends that to the DSTROOT directory
Deployment Location	DEPLOYMENT_LOCATION	YES	Tells Xcode not to use the Built Products location, and instead to use the installation directory
Wrapper Extension	WRAPPER_EXTENSION	xcplugin	This is the plugin extension that Xcode is looking for
Custom setting	GCC_ENABLE_OBJC_GC	supported	Xcode is a GC-enabled application, so our plugin also needs to be

Adding Behaviour

So, now the project's settings should be correct, but we don't yet have any code. Add a new class called `BDXcodePluginDemo` that inherits from `NSObject`. Open the `BDXcodePluginDemo.m` file, and add the following method:

```objc
+(void)pluginDidLoad:(NSBundle *)plugin {
        NSLog(@"This is our first Xcode plugin!");
}
```

Confirming Build

You should now be able to build the project, and check by running the following command:

```
ls -las $HOME/Library/Application\ Support/Developer/Shared/Xcode/Plug-ins
```

If all goes well, you should see something like:

```
0 drwxr-xr-x  3 edwardsc  staff  102 20 Oct 10:16 .
0 drwxr-xr-x  7 edwardsc  staff  238 19 Oct 16:52 ..
0 drwxr-xr-x  3 edwardsc  staff  102 20 Oct 10:16 BDXcodePluginDemo.xcplugin
```

Restarting Xcode

Run the following command and restart Xcode:

```
tail -f /var/log/system.log
```

You should see `This is our first Xcode plugin!` appear in the log if all is successful.

Source Code

The source code and project for this article is available in the BDXcodePluginDemo project on GitHub.

That project contains the latest code in my series of plugin articles, but you can get the version that relates to this article using the `creating-an-xcode4-plugin` tag.

What Next?

My next article will discuss a few common things that you might want to do. Specifically:

- Listening for notifications. This will give you an idea of the types of things that you can hook into.
- Adding menus. Depending on your plugin, you will probably want to be able to add a menu item.

All articles in this series

- Creating an Xcode4 plugin
- Common Plugin Techniques
- Introducing XcodeExplorer
- Checking out the Xcode controls

7 Comments »

1. *Fellipe* says:
   December 29, 2012 at 5:42 pm

   Good afternoon!

   I'm having some difficulties running my plugin! Namely, when I restart Xcode, the logs say that the plugin could not be loaded due to a GC capability mismatch. Below ate the mentioned log entries:

   Dec 29 15:32:08 Fellipe-Santiago-Scarpa-Caetanos-MacBook.local Xcode[2201]: Error loading /Users/fellipe/Library/Application Support/Developer/Shared/Xcode/Plug-ins/FCImportCleaner.xcplugin/Contents/MacOS/FCImportCleaner: dlopen(/Users/fellipe/Library/Application Support/Developer/Shared/Xcode/Plug-ins/FCImportCleaner.xcplugin/Contents/MacOS/FCImportCleaner, 265): no suitable image found. Did find: /Users/fellipe/Library/Application Support/Developer/Shared/Xcode/Plug-ins/FCImportCleaner.xcplugin/Contents/MacOS/FCImportCleaner: GC capability mismatch

   Dec 29 15:32:08 Fellipe-Santiago-Scarpa-Caetanos-MacBook.local Xcode[2201]: WARNING: Failed to load plugin at: /Users/fellipe/Library/Application Support/Developer/Shared/Xcode/Plug-ins/FCImportCleaner.xcplugin, skipping. Could not load bundle.

2. *craig* says:
   December 30, 2012 at 7:29 am

   Two things to check:

   1) Have you added XCGCReady as a Boolean property to your .plist file?

   2) Have you added the custom setting for GCC_ENABLE_OBJC_GC=supported to the compile-time options for your target?

   If you can email me your project (craig@blackdogfoundry.com), I would be happy to have a look to see if I can find the problem.

3. *nick* says:
   June 4, 2013 at 2:55 pm

   I copied your settings, but when I run the tail I get this error Details: Error writing UI state:Error Domain=DVTCustomDataStoringErrors Code=0 "Cannot save data because the write destination is not writable." UserInfo=0x405967fc0 {NSLocalizedDescription=Cannot save data because the write destination is not writable., NSUnderlyingError=0x406152bc0 "The operation couldn't be completed. (IDEFoundationErrorDomain error 6.)"} Object: Method: -writeStateData Thread: {name = (null), num = 1} Please file a bug at http://bugreport.apple.com with this warning message and any useful information you can provide. Jun 4 09:55:11 dev05A Xcode[50993]: [MT] DVTAssertions: Warning in /SourceCache/IDEKit/IDEKit-2109/Framework/Classes/Workspace/IDEWorkspaceDocument.m:1982 Details: Error writing UI state:Error Domain=DVTCustomDataStoringErrors Code=0 "Cannot save data because the write destination is not writable." UserInfo=0x405777e80 {NSLocalizedDescription=Cannot save data because the write destination is not writable., NSUnderlyingError=0x4056b06a0 "The operation couldn't be completed. (IDEFoundationErrorDomain error 6.)"} Object: Method: -writeStateData Thread: {name = (null), num = 1} Please file a bug at http://bugreport.apple.com with this warning message and any useful information you can provide.

   any ideas where I change permissions?

4. *craig* says:
   June 4, 2013 at 11:00 pm

   I don't think that error is related to your plugin. It just looks like an (unrelated) XCode problem.

5. _Tarek_ says:

    July 10, 2013 at 2:54 am

    Thank you very much for the tutorial!

    All seems to be working perfectly until Xcode5 developer preview 3. DP3 gives the error message: "[MT] PluginLoading: Required plug-in compatibility UUID 63FC1C47-140D-42B0-BB4D-A10B2D225574 for plug-in at path '~/Library/Application Support/Developer/Shared/Xcode/Plug-ins/XcodeExplorer.xcplugin' not present in DVTPlugInCompatibilityUUIDs".

    It seems that all Xcode plugins won't be loaded in Xcode5 DP3. Back to square one?

6. _craig_ says:

    July 11, 2013 at 4:58 am

    I can't discuss the fix as we are both bound by the NDA. However, I will say that getting the plugins working again in Xcode5 is as simple as adding an appropriate entry in the plugin's .plist file.

    Once Xcode5 has been generally released, I will update this article.

7. _Clay Bridges_ says:

    July 29, 2013 at 1:00 pm

    Cf. this already public information:

    https://github.com/gfontenot/XVim/commit/35bbf24bc074e330a9bea852c190d8fdcbfff1ea

Leave a Comment »

```
[                    ]
```
Name (required)
```
[                    ]
```
Mail (will not be published)(required)
```
[                    ]
```
Website
```
[                                        ]
[                                        ]
[                                        ]
[                                        ]
[                                        ]
```

[Submit]

Categories

- Tips (7)
- Tutorials (9)