Dropsy (1.0) for El Capitan, Sierra & High Sierra
Release Notes
Howard Oakley https://eclecticlight.co

These release notes describe how to use and customise **Dropsy**, a simple example app onto which files and folders can be dragged and dropped. It is written in Swift 4 for Xcode version 9.2, and should be compatible with Swift 4.1 and later versions of Xcode too.

Dropsy is a prototype app to perform an action on the files and folders which are dropped onto it, or opened using its menu commands. For the sake of this example, the code simply makes a duplicate copy of whatever is dropped onto or opened in it. You can customise that to perform any other actions which might previously have been part of an AppleScript or other script app.

Although it accepts drag and drop, and handles multiple files and folders, it does not (yet) offer itself as a functional Service through the Finder's contextual menu. I haven't managed to get code working in Swift to do that yet, but you may be luckier or more persistent than me.

Structure

Dropsy has a simple structure consisting of:
- an **AppDelegate**, which is set *not* to open an Untitled file when starting;
- a custom **DRWindow**, which does nothing other than avoids an annoying Xcode warning;
- a **ViewController** which also customises NSWindowDelegate, which does the work;
- a **Document**, which provides a custom `read()` function which is currently empty;
- a single-window **storyboard**, in which there is only a scrolling text view into which you can readily write messages;
- custom document types and services to ensure it accepts all file and folder types.

Document types & services

Select the project at the far left, then the app target, and the Info view within that, to view and edit document types and services as shown below.
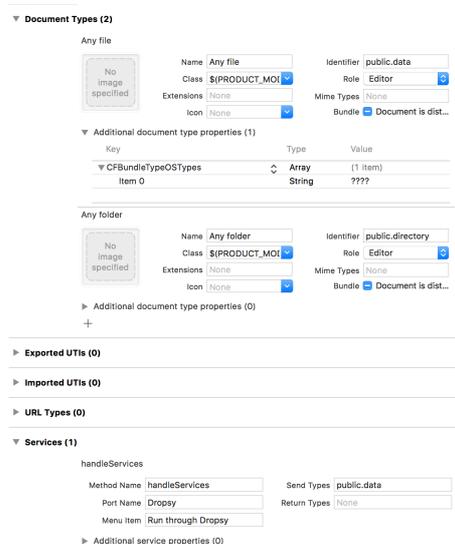
The two document types use UTIs, which you will probably find simpler when you want to accept broad classes of items rather than specific document types. The settings in Services should only be relevant if you are trying to get this to work as a service.

AppDelegate & Document

The only modification to AppDelegate is to add the `applicationShouldOpenUntitledFile()` function, which by returning **false** ensures that the app doesn't keep opening new blank documents.

ViewController
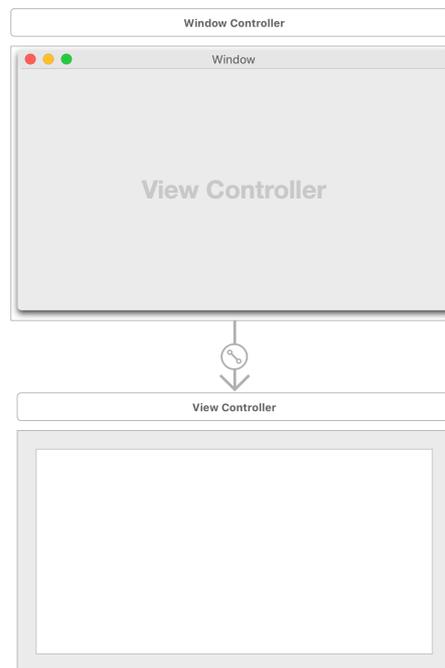
The two **IBOutlets** give access to the scroller (top level) and TextView of the scrolling text view in the window, and two utility functions are provided to make it easy to replace or append text content in that.

The work is performed by the overridden `viewDidAppear()` function. This gets the document, and obtains its URL, as the source URL. It then constructs a suitable name for the output file, and displays the Save File dialog as a sheet in the document window.

When the user clicks on the OK button, the code extracts the destination URL, and copies the source to the destination. It finally copies the attributes of the source to the destination, so that the items appear identical.

This will perform shallow copies through folders, handle complex document types such as RTFD, and work with files opened through the menus and those dragged and dropped onto the app.

Dropsy (1.0) for El Capitan, Sierra & High Sierra
Release Notes
Howard Oakley https://eclecticlight.co

```swift
override func viewDidAppear() {
    self.view.window?.delegate = self
    if let theDoc = self.view.window?.windowController?.document as! Document? {
        if let sourceUrl = theDoc.fileURL {
            let theFName1 = sourceUrl.deletingPathExtension().lastPathComponent
            var theFExt1 = sourceUrl.pathExtension
            if theFExt1 != "" {
                theFExt1 = "." + theFExt1
            }
            let FS = NSSavePanel()
            FS.showsHiddenFiles = true
            FS.treatsFilePackagesAsDirectories = true
            FS.canCreateDirectories = true
            FS.prompt = "Duplicate to file"
            FS.message = "Enter the name of a new file"
            FS.nameFieldStringValue = theFName1 + "-Copy" + theFExt1
            FS.beginSheetModal(for: self.view.window!) { result in
                if result.rawValue == NSFileHandlingPanelOKButton {
                    guard let destUrl = FS.url else { return }
                    let FM = FileManager.default
                    do {
                        try FM.copyItem(at: sourceUrl, to: destUrl)
                        let theAttrs = try FM.attributesOfItem(atPath: sourceUrl.path)
                        try FM.setAttributes(theAttrs, ofItemAtPath: destUrl.path)
                        let theStr = "Saved copy of " + sourceUrl.lastPathComponent + " to " +
                            destUrl.lastPathComponent + "\n"
                        self.appendOutputText(string: theStr)
                    } catch {
                        self.appendOutputText(string: error.localizedDescription)
                    }
                }
            }
        }
    }
}
```

Rights

Dropsy is provided free of charge and without any claims over copyright or other ownership, despite what may be automatically generated into its source files. You are welcome to use it for any project whatsoever without obtaining any other permissions or the payment of any royalties, etc. The only exception to this is publishing or distributing the source or project code for any commercial purpose and claiming it as your own: I don't like plagiarism of any kind, please.

If you use it, I would greatly appreciate your acknowledging that fact in your app's credits, together with a link to The Eclectic Light Company blog at `https://eclecticlight.co` However that is entirely optional. It would also be really nice if you could comment on my blog as to what you have done with it, if you can.

If you make changes or add features such as getting it to function as a Finder Service, I'd really appreciate knowing about it, please. Then others can benefit from those improvements. If you improve or fix my rubbish coding, then please let me know too – I'm always happy to learn.

Change list

Dropsy 1.0:
• initial release, without function as a Service

11 April 2018.