

DropShell 2.0 — July 1st 1994

by Leonard Rosenthol, Marshall Clow, and Stephan Somogyi

DropShell is a collection of C source files that allow you to easily build your own System 7-style drop-box applications.

Basic Features

DropShell consists of four source files and five header files that are designed to let you quickly add your own application's functionality without having to change the DS sources themselves much.

DropShell implements support for dragging files and folder onto the application's icon via AppleEvents — it doesn't cheat and use the System 6 compatibility provided by System 7. DropShell is also a fully factored application — it performs its work by sending AppleEvents to itself. This also allows an application built on DropShell to be “recorded” by an OSA-compatible scripting system.

DropShell handles a simple user interface including a splash screen, a “Select File...” option and complete balloon help for all menus and dialogs.

Changes for 2.0

- DropShell is no longer provided in a Pascal version. We hope Peter and Quinn will forgive us.
- DropShell now uses the Universal Headers and can be compiled for 68k or PowerPC.
- DropShell is supplied with project files for Metrowerks' integrated 68k and PowerPC compilers. These projects were built with version 1.0.1 of the compilers, which are on the CodeWarrior 3.5 CD. The source code, however, also compiles under the THINK C 7.x, Symantec C++ 7.x for both 68k and PowerPC, MPW C, and Apple PPCC compilers.
- A small number of changes have been made since version 1.0 to ease development. For example, PreFlightDocs receives as a parameter the number of FSSpecs passed from the ODOC AppleEvent.
- Several handy utility routines have been added to DSUtils.c

How to Build a DropShell Application

The source is well-commented as to how the DropShell is organized and contains some very useful programming techniques and reusable source code.

- **DropShell.c** — This file contains `main()` as well as ancillary code such as menu handling that makes DropShell a Macintosh app. You will only have to make minor modifications to this file.
- **DSAppleEvents.c** — This file contains all the AppleEvent handler code. You should never have to modify this file.
- **DSUserProcs.c** — This is the main file you will have to change and the one you should study the closest. It contains the code that gains control after an AppleEvent is received. Control is handed to code in this file as follows:

PreFlightDoc - receives control after an ODOC AppleEvent is received but before each individual items within the ODOC is processed and allows you to make preparations to handle the items. It also receives the number of files/folders in the AppleEvent, in case you need to preparations based on the number.

OpenDoc - is called once per item passed in an ODOC event. It determines whether the item is a file or folder and then calls `ProcessItem` or `ProcessFolder` accordingly.

PostFlightDoc - receives control after every item has been processed by OpenDoc.

- **DSUtils.c** — This file contains a number of very general and reusable utility routines used by DropShell. It contains, among other things, code for walking folders, to force a Finder update, miscellaneous FSSpec handling routines, and other useful stuff.

General Comments

It's worth spending the time to investigate how DropShell works by single-stepping through it with a source-level debugger. DropShell isn't large and you will gain a much better understanding of it this way.

Rules and Regulations

The authors provide this source code as-is, with no warranties about its suitability for any purpose whatsoever. If you format your hard disk, or do anything else inconvenient, it's not our fault.

This source code may be distributed freely, as long as all the files remain together, are unaltered, and are accompanied by this documentation. Under no circumstances are you permitted to distribute modified versions of the DropShell source code, headers or its documentation.

There is no licensing fee associated with using DropShell. We do ask, however, that you mention DropShell and its authors in your application's About box. We are also very interested

to see how people use DropShell and would appreciate receiving copies of applications created with it.

Contacting the Authors

If you have any questions or comments, the authors can be reached via e-mail at the following addresses.

Leonard Rosenthol — leonardr@netcom.com

Marshall Clow — mclow@san_marcos.csusm.edu

Stephan Somogyi — somogyi@ziff.com