

NAME

platypus — create Mac OS X application wrapper around a script.

SYNOPSIS

```
platypus [-vh] [-O] [-FASDBR] [-ydlHx] [-bgn] [-KYL] [-P profile] [-a appName]
  [-o outputType] [-i icon] [-p interpreter] [-V version] [-u author]
  [-I identifier] [-f bundledFile] [-X suffixes] [-T filetypes]
  [-G interpreterArgs] [-C scriptArgs] [scriptFile]
  [destinationPath]
```

DESCRIPTION

platypus is the command line counterpart to the Platypus Mac OS X application. It is a tool which creates a Mac OS X application bundle which executes a script when run.

platypus defaults to creating application bundles, but can also create Platypus profiles that can subsequently be loaded by either the command line program or the Platypus graphical application.

The following flags are supported (and are parsed in the order in which they are passed):

- O Create profile file as output instead of creating an application bundle. When you select this option, the "destinationPath" parameter (i.e. the final parameter to the program) should have a .platypus suffix. Please note that in order to create reusable profiles, this option requires that all paths to bundled files, icons etc. be absolute. If you provide '-' as the output file, the profile property list will be dumped to STDOUT.
- P [*profile*]
Loads all settings from a Platypus profile file. You will still need to specify a destination path for the application. It is possible to override individual settings in the profile with subsequent parameters.
- a [*appName*]
Specifies the name of the application. This can be different from the name of the .app bundle itself, and is displayed in the application's menus, "About" window and Info.plist property list.
- o [*outputType*]
Specifies the script application's output type, which can be one of the following:
 - 'None' The application does not display any output.
 - 'Progress Bar' The application displays a progress bar while running.
 - 'Text Window' The application displays a window with a text field with all script output.
 - 'Web View' The application displays a window with a web view which renders script output as HTML.
 - 'Status Menu' The application displays a status menu item in the menubar which runs the script and displays its output when selected.
 - 'Droplet' The application displays a droplet window for dropping files on for processing by the script.The default output type is "Progress Bar".
- i [*icon*]
Specifies a file to use as icon for the application bundle. The icon file should be a regular Mac OS X .icns file. If not specified, the default Platypus app icon will be used. If specified, but left empty (""), no application icon will be set for the app.
- Q [*icon*]
Specifies a file to use as icon for the application's documents. Should be a Mac OS X .icns file.
- p [*interpreter*]
Sets the interpreter for your script (e.g. /usr/bin/perl). If the interpreter is not specified, the command line tool will try to guess the correct interpreter. If this fails, the standard shell interpreter /bin/sh is used.

- v** [*version*]
Sets the application bundle's version. This is displayed in the Info.plist property list and About window.
- u** [*author*]
Sets the name of the application author (e.g. "Apple Computer" or "John Smith"). If not specified, it defaults to the current user's full user name.
- f** [*bundledFile*]
Specifies a file to be bundled with the script application. This file will be copied over the Resources folder of the application bundle, which is the same folder in which your script will eventually reside. You can specify any number of files to be bundled, but you must use the -f flag each time, since the flag only accepts a single file path after the -f flag.
- I** [*identifier*]
Sets the application's bundle identifier. An application identifier is a reverse DNS name (e.g. com.apple.iTunes) that uniquely identifies your application. If this option is left empty, it will default to an identifier of the format "org.username.appname" (e.g. org.sveinbjorn.Platypus).
- A** This flag makes the application request administrator privileges via Apple's Security Framework (i.e. prompt for a password) and then executes the script with those privileges. For details on the nature of these privileges, see the Apple documentation for the AuthorizationExecuteWithPrivileges() in Security.framework. This is not quite equivalent to root privileges.
- S** Secure bundled script. This encrypts the bundled script, to fend off prying eyes. Please note that nothing will make the script 'secure' from anyone knowledgeable enough to get a clear text version.
- D** Makes your script application droppable, i.e. capable of receiving drag and dropped files as arguments to the script. The application bundle's property list is modified so that it can receive dropped files in the Dock and Finder. These files are then passed on to the script as arguments via @ARGV.
- F** Makes your script application droppable, so it accepts dragged text snippets, which are passed to script as arguments.
- B** This option causes the application to run in the background. Its icon will not appear in the Dock. This is done by registering the application with the Finder as a user interface element (LSUIElement).
- R** This option makes the application quit after the script has been executed.
- x** [*suffixes*]
Only appropriate if you are using the -D option. This flag allows you to specify the file suffixes (e.g. .txt, .wav) your application can open. This should be a |-separated string (e.g. "txt|wav|jpg").
- T** [*filetypes*]
Only appropriate if you are using the -D option. This flag allows you to specify the file type codes (e.g. '8BIM', 'TEXT') your application can open. This should be a |-separated string of 4-character ASCII strings (e.g. "8BIM|TEXT|R*ch").
- G** [*arguments*]
Arguments for the script interpreter. These should be specified as a |-separated string (e.g. '-w|-s|-l').
- C** [*arguments*]
Arguments to the script. These should be specified as a |-separated string (e.g. '-w|-s|-l').
- b** [*hexColor*]
For Text Window output mode only. Set background color of text output (e.g. #ffffff).

- g** [*hexColor*]
For Text Window output mode only. Set foreground color of text output (e.g. #000000).
- n** [*fontName*]
For Text Window output mode only. Set font and fontsize for text output field (e.g. 'Monaco 10').
- E** [*encodingNumber*]
For Text Window output mode only. Set text encoding for script output. Must be one of the numbers defined in the NSString class reference (e.g. 4=UTF-8, 1=ASCII, etc.). Default is UTF8.
- K** [*kind*]
For Status Menu output mode only. Set display kind for Status Menu output mode. This can be "Text", "Icon and text" or "Icon".
- Y** [*title*]
For Status Menu output mode only. Set the display title for the status item in Status Menu output mode.
- I** [*imagePath*]
For Status Menu output mode only. Set the icon image for the status item in Status Menu output mode. This must be a 16x16 pixel image in one of the image formats supported by the Cocoa APIs.
- x** Create XML format property lists instead of binary ones.
- d** Development mode. A symlink to the original script is created inside the application bundle instead of a copy. Symlinks are also created for any bundled files. This option is incompatible with the -S option.
- l** Don't optimize application. Platypus defaults to compiling the bundled xib file to reduce application size, which makes it uneditable. Optimization only takes place if you have Apple's Developer Tools installed, since it uses the program /Developer/usr/bin/ibtool.
- y** Force mode. With this flag set, the program will happily overwrite any previous files and folders in destination path. Use with caution.
- H** [*xibPath*]
Specify an alternate xib to copy to application bundle. This allows you to integrate a customised Platypus application xib into your build process.
- v** Print the version of this program
- h** Print help and usage string

Exits 0 on success, and >0 if an error occurs.

EXAMPLES

```
platypus -P myProfile.platypus ~/Desktop/MyApplication.app
```

```
platypus -o 'Text Window' script.pl PerlScript.app
```

```
platypus -a 'My App' -p /usr/bin/python myPythonScript
```

```
platypus -D -a MyDroplet -o 'Droplet' ~/droplet.sh
```

FILES

/usr/local/bin/platypus	program binary
/usr/local/share/platypus/ScriptExec	executable binary
/usr/local/share/platypus/MainMenu.nib	Nib file for app
/usr/local/share/platypus/PlatypusDefault.icns	Default icon

AUTHORS

This manual page was written by Sveinbjorn Thordarson <sveinbjornt@gmail.com> To support Platypus development, please visit <http://sveinbjorn.org/donations>.