Extended Attributes and Zip Archive

This thread has been locked by a moderator.

677

This has come up a few times so I thought I’d write it down so that Future Quinn™ could point folks at it.

If you have any questions or feedback, please start a new thread and tag it with *Files and Storage* so that I see it.

Share and Enjoy
—
Quinn “The Eskimo!” @ Developer Technical Support @ Apple
`let myEmail = "eskimo" + "1" + "@" + "apple.com"`

Extended Attributes and Zip Archive

In a traditional Unix file system, a file consists of a flat sequence of bytes and a very limited set of metadata. In contrast, a traditional Mac OS file had a second sequence of bytes, the resource fork, and various other bits of Mac metadata. This caused an interoperability problem between Mac and Unix systems because the latter has no place to store the Mac metadata.

To resolve this problem Apple introduced the AppleDouble file format. This stores the Mac’s data fork in the Unix file and all the Mac metadata in a companion AppleDouble file. If the file name was `foo`, this companion file was called `._foo`. The leading dot meant that this companion file was hidden by many tools.

macOS does not typically need AppleDouble files because Apple file systems, like APFS and HFS Plus, have built-in support for Mac metadata. However, macOS will create and use AppleDouble files when working with a volume format that doesn’t support Mac metadata, like FAT32.

Similarly macOS will create and use AppleDouble files when working with archive file formats, like zip, that don’t have native support for Mac metadata.

macOS 10.4 added support for extended attributes at the Unix layer. For more on this, see the `getxattr` man page. As with traditional Mac metadata, this is supported natively by Apple file systems but must be stored in an AppleDouble file elsewhere.

Note At the API level the original Mac metadata is modelled as two well-known extended attributes, `XATTR_RESOURCEFORK_NAME` and `XATTR_FINDERINFO_NAME`.

When creating a zip archive macOS supports two approaches for storing AppleDouble files:

- By default it stores the AppleDouble file next to the original file.

```
% ditto -c -k --keepParent root root-default.zip
% unzip -t root-default.zip
Archive:  root-default.zip
    testing: root/                                      OK
    testing: root/nested/                               OK
    testing: root/nested/FileWithMetadata               OK
    testing: root/nested/._FileWithMetadata             OK
No errors detected in compressed data of root-default.zip.
```

- Alternatively, it can create a parallel hierarchy, rooted in `__MACOSX`, that holds all AppleDouble files.

```
% ditto -c -k --keepParent --sequesterRsrc root root-sequestered.zip
% unzip -t root-sequestered.zip
Archive:  root-sequestered.zip
    testing: root/                                      OK
    testing: root/nested/                               OK
    testing: root/nested/FileWithMetadata               OK
    testing: __MACOSX/                                  OK
    testing: __MACOSX/root/                             OK
    testing: __MACOSX/root/nested/                      OK
    testing: __MACOSX/root/nested/._FileWithMetadata    OK
No errors detected in compressed data of root-sequestered.zip.
```

The latter is commonly used for mail attachments because it makes it easy for the recipient to discard all the Mac metadata.

Files and Storage

Reply

Posted 1 year ago by eskimo

Add a Comment

Developer › Forums

Platforms
iOS
iPadOS
macOS
tvOS
watchOS

Tools
Swift
SwiftUI
SF Symbols
Swift Playgrounds
TestFlight
Xcode
Xcode Cloud

Topics & Technologies
Accessibility
Accessories
App Extensions
App Store
Audio & Video
Augmented Reality
Business
Design
Distribution
Education
Fonts
Games
Health & Fitness
In-App Purchase
Localization
Maps & Location
Machine Learning
Security
Safari & Web

Resources
Documentation
Curriculum
Downloads
Forums
Videos

Support
Support Articles
Contact Us
Bug Reporting
System Status

Account
Apple Developer
App Store Connect
Certificates, IDs, & Profiles
Feedback Assistant

Programs
Apple Developer Program
Apple Developer Enterprise Program
App Store Small Business Program
MFi Program
News Partner Program
Video Partner Program
Security Bounty Program
Security Research Device Program

Events
App Accelerators
App Store Awards
Apple Design Awards
Apple Developer Academies
Entrepreneur Camp
Tech Talks
WWDC

To view the latest developer news, visit News and Updates .