



Products Services Developer Resources Contact STS About STS



Trapping the Quit <application> Menu Item



There are two ways to trap the Quit <application> menu item in OS X, depending on what you want to do.

Approach #1: Apple Events

The first method is kind of "brute force", and just traps the Quit apple event in an `appleEvent` handler. If you don't pass the `appleEvent`, your application will not quit. This is useful if you have an application that you want to keep the user from quitting. Simply trap the `aevtquit` Apple Event, and don't pass it. If you **do** want to quit, do the things you want to do and then pass it. For example, this provides a confirmation dialog:

```
on appleEvent pClass,pID,pSender
  switch (pClass & pID)
  case "aevtquit"
    answer "Are you sure you want to quit?" with "Yes" or "No"
    if it = "Yes" then
      pass appleEvent
    end if
    break
  default
    pass appleEvent
    break
  end switch
end appleEvent
```

This approach is also useful if your standalone does not have any menus; it prevents you from having to add a dummy menubar and the corresponding stack resizing that goes with it.

Approach #2: Using a Menu Item

In this case, the menu item in the File menu that you create needs to exactly match what is presented by OS X in the Application menu, which is "Quit <appName>", where <appName> is the name of the \ application defined in the `Info.plist` file in the `CFBundleExecutable` key. For a standalone built with MetaCard and not changed, this would read "Quit MetaCard".

So how do you deal with a standalone that is to play in both OS 9 and OS X? In OS 9, you want the File menu's last item to read **Quit**, but in OS X to read **Quit MyApp**. MetaCard/Revolution will automatically take the last two lines of the first menu (a separator and the Quit menu item) and remove them under OS X to eliminate duplication with the **Quit MyApp** menu item which is automatically created by the OS and placed in the Application Menu.

So the best way to do this is to define your File menu so that the last two items are a separator followed by **Quit**, and then in the `menuPick` handler of the File menu button do something like this:

```
on menuPick pItem
  switch pItem
  -- "case"s for other menu items go here
  case "Quit"
  case "Quit MyApp"
    doQuit -- your quit handler or just "quit"
    break
  end switch
```

This will show **Quit** in OS 9, and **Quit MyApp** in OS X, and the `menuPick` handler will trap for both and allow you to process the event.

Posted 1/18/2003 by Ken Ray - Thanks go to Richard Gaskin of [FourthWorld Media Corporation](#) for help on this

[Print this tip](#)

[News and Rumors](#) [Products](#) [Services](#) [Developer Resources](#) [Contact STS](#) [About STS](#)

Copyright ©1997-2013 Sons of Thunder Software, Inc. All rights reserved.
Send all comments to webmaster@sonsofthunder.com.