

On Free Memory



This thread has been locked by a moderator.



👁 1.3k

Apple platforms have APIs, inherited from Mach, that return low-level memory statistics [1]. Some third-party tools report those statistics as if they were relevant to app developers. In most cases they are not.

Specifically, any report on the amount of *free memory* is unlikely to be useful. This number is the size of the kernel's free memory pool. Most VM-based OSes attempt to keep a certain amount of free memory available to avoid running out of memory in critical circumstances (for example, when trying to allocate memory on the page out path). But the size of that pool has very little to do with the amount of memory pressure on the system. Remember that the kernel's goal is to keep the pool at a certain size, so it lets memory go to ‘waste’ until critical thresholds are crossed, as which point it starts cleaning up.

As a concrete example of this, consider an iPhone running Safari. When you press the Home button, you return to the home screen but the system leaves Safari loaded in memory. At this point the free memory is X, where X is slightly higher than the threshold where the system would remove Safari from memory. Now apply some memory pressure to the system. It responds by removing Safari from memory, whereupon the free memory jumps up to Y, where Y is much larger than X. So putting memory pressure on the system has resulted in a *larger* free memory value. If you were using this free memory value as a proxy for memory pressure, you'd end up drawing exactly the wrong conclusion.

Apple platforms have various standard techniques for handling memory pressure — memory warnings, `NSCache`, purgeable memory, `DISPATCH_SOURCE_TYPE_MEMORYPRESSURE`, and so on — and you should use those techniques rather than any sort of free memory value.

Finally:

- If you want to get a summary of how much memory your app is using, you should look at its *footprint*. If you're building an automated test suite, you can get this value programmatically. See [this post](#) for details.
- If you want to find out how much memory headroom you have, check out [the `os_proc_available_memory` function](#).

Share and Enjoy

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

[1] Specifically, the `host_statistics` API and the various structures defined in `<mach/vm_statistics.h>`.

Revision History

- **2023-03-03** Added a reference to `os_proc_available_memory`.
- **2021-02-27** Fixed a broken link. Made other minor editorial changes.
- **2019-06-27** First posted.

Kernel

Reply

Posted 3 years ago by eskimo

Add a Comment

This site contains user submitted content, comments and opinions and is for informational purposes only. Apple disclaims any and all liability for the acts, omissions and conduct of any third parties in connection with or related to your use of the site. All postings and use of the content on this site are subject to the [Apple Developer Forums Participation Agreement](#).

🍏 > 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