Flutter
RFC: Move 32-bit iOS to "Best Effort" Tier

SUMMARY

Flutter's support for 32-bit iOS devices should move from the "Supported" tier to the "Best Effort" tier.

Author: Zach Anderson (zanderso@github)
Go Link: flutter.dev/go/rfc-32-bit-ios-support
Created: 12/21 / **Last updated:** 12/21

OBJECTIVE

The goal of this document is to gather comments considering whether Flutter's support for 32-bit iOS devices should move from the "Supported" tier to the "Best Effort" tier.

BACKGROUND

Flutter's support tiers are defined at flutter.dev. In particular, tier 1 (the "Supported" tier) requires that the platform is tested on Flutter's CI on pre- and post- commit, and undergoes additional validation before release on Flutter's beta channel. In contrast, for tier 2 (the "Best Effort" tier) Flutter relies on the community to test releases while Flutter supports these platforms only through coding practices, build configurations, and ad-hoc testing (e.g. at an engineer's desk).

Flutter defines support for iOS devices in terms of the iOS version. The minimum iOS version in the "Supported" tier is 9.3.6. This is the most recent version that will run on the oldest device in the device lab, an iPhone 4S. The most recent version that can run on the next older device (iPhone 4) is iOS 7, which cannot run Flutter.

Usage of older iOS versions is quite low. Apple reports that 93% of all iOS devices

are on iOS 13 or newer. StatCounter reports that 1.52% of devices run iOS 9 or older. Slack and others are dropping support even for versions as new as iOS 12. A selling point of Flutter is support for older devices, however this should be balanced against the maintenance burden.

Only 32-bit devices can run 9.3.6. All 64-bit devices will automatically upgrade to a more recent version. The 32-bit devices that can run Flutter are the iPhone 4S, 5, and 5C, and the 2nd, 3d and 4th generation iPads. The iPhone 5C and the 4th gen iPad would not run 9.3.6. Rather they would upgrade to iOS 10.3. Apple places the iPhone 5 and 5C, and the 4th generation iPad in its "Vintage" tier, while the iPhone 4S, and the 2nd and 3d generation iPads are in its "Obsolete" tier.

Flutter CI currently runs one integration test (native_ui_tests) and one benchmark (flutter_gallery__transition_perf_e2e) on a 32-bit iPhone 4S device running iOS 9.3.6. None of the other tests or benchmarks are run, and no other 32-bit iOS devices are tested on Flutter's CI.

OVERVIEW

Flutter's current testing of iOS 9.3.6 doesn't meet the requirements for the "Supported" tier, and expanding testing is infeasible. Support for iOS 9.3.6 should therefore be officially reduced to the "Best Effort" tier.

This would allow the iPhone 4S device, which is expensive to maintain, to be removed from CI. This would also have the side-effect that the minimum iOS version in the "Supported" tier would increase to iOS 14, which would exclude all 32-bit iOS devices. This change would not affect the support level of devices already in the "Best Effort" tier such as iOS 10 devices. Because of this side-effect, this doc discusses support for 32-bit devices rather than support for iOS 9.3.6.

Non-goals

This doc does not consider whether support for 32-bit devices should be dropped down to Flutter's "Unsupported" tier. However, I outline a proposed timeline for doing so below.

DETAILED DISCUSSION

The current level of testing on 32-bit devices is very limited. Flutter CI runs one integration test and one benchmark. These two tests are low-coverage and have not caught any issues for over a year. (Instead, issues impacting operation on iOS >8 have been caught while building flutter/engine artifacts.)

Testing on 32-bit devices is also at a reduced level because the single iPhone 4S device in the lab is frequently inoperative. In addition to the tests being marked flaky, and being hosted only in the "staging" environment, in the past month the

integration test has had only ~20 successful runs out of the past >400 attempts, and the benchmark has had only 2 successful runs in the same time period on a similar number of attempts.

Expanding test coverage on this class of devices is infeasible for a few reasons.

- Keeping the lab's only such device working requires frequent manual intervention. Adding further devices would increase the likelihood that there is always at least one device requiring attention. This would not be a good use of the infrastructure team's on-call bandwidth
- The devices would have to be acquired piecemeal from third-party sellers e.g. ebay. Adding them to CI would require an investment from the infrastructure team that could be better spent elsewhere.
- Getting more tests working on 32-bit devices would require an investment of engineering time from the iOS team. Since the full test suite has not run on these devices in a long time (or maybe ever?) it is likely that many of them will require modifications before working correctly.

In addition to more accurately reflecting the de facto support level, moving 32-bit to the "Best Effort" tier would have some follow-on benefits.

- The iPhone 4S devices could be removed from the device lab, lowering maintenance burden on the infrastructure team.
- The Mac tethered to the 4S devices can host newer devices, which will increase non-32-bit capacity.
- The 32-bit tests and benchmarks can be deleted along with their entries in infrastructure configuration files.
- The natural progression of these older devices and iOS versions from "Supported", to "Best Effort", to "Unsupported" could be smooth and well-communicated rather than jumping directly from "Supported" to "Unsupported".

User Impact

Between now and the 32-bit devices moving to the "Unsupported" tier, users are unlikely to experience any regression in support. Since test coverage is already low, and issues affecting 32-bit/iOS 9 devices are already caught when building flutter/engine artifacts, the support level is effectively already at the "Best Effort" level.

Alternatives

Since the iPhone 4S device in the device lab is the only 32-bit device, removing it and doing nothing else means that support for all 32-bit iOS devices becomes no better than "Best effort". With respect to this document, that effect is working as intended. Compensating by adding e.g. iPhone 5C devices to the device lab would

work against the benefits of removing the iPhone 4S devices. In particular, doing so would require ongoing investment from the iOS and infrastructure teams. That option seems undesirable, however it does achieve the goal of reducing the support level for iOS 9.3.6.

DOCUMENTATION PLAN

The change would need to be announced and the website updated to reflect it.

Draft announcement:

> Due to decreased usage, and increased difficulty in maintaining the target devices in our lab, we are moving support for iOS 9.3.6 from the "Supported" tier to the "Best Effort" tier. This means that support for iOS 9.3.6 and support for 32-bit iOS devices will be maintained only through coding practices, and ad-hoc and community testing. See the description of Flutter's support tiers here.
>
> In the second half of this year, we expect to drop support for 32-bit iOS devices from Flutter stable releases. This means that apps built against stable Flutter SDKs after that point will no longer work on 32-bit iOS devices, and the minimum iOS version supported by Flutter will increase to iOS 12.5. Stay tuned for further announcements with a more detailed timeline.

TIMELINE

Here is a proposed timeline for moving 32-bit iOS devices from "Supported" to "Best Effort", and then from "Best Effort" to "Unsupported" with sample dates for the releases after 2.10.

When	What
Now	Begin removing iPhone 4S devices from infrastructure (#97148)
2.10 stable, early February 2022	- Announce that 32-bit iOS devices are only supported as part of the "Best Effort" tier. - Announce that the following stable release will be the last release to support 32-bit iOS devices.
2.13(?) beta, April(?) 2022	Following this branch cut, support for

	32-bit iOS can be removed/broken on the master channel.
2.13(?) stable, May(?) 2022	The last stable release to support 32-bit iOS.
2.14(?) beta, May(?) 2022	The first beta release in which 32-bit iOS devices are in the “Unsupported” tier.
2.17(?) stable, August(?) 2022	The first stable release in which 32-bit iOS devices are in the “Unsupported” tier.