

Apache Cordova: Next Major Planning

General

- Commit package-lock.json
- [DONE] Drop node 4 support, add node 10 support
- Drop dependency on Q, use native promises [[CB-14158](#)]
- Remove any leftover support for .cordova/config.json
- Update READMEs to describe the current public interface of the package
- [DONE] Remove any committed node_modules in platforms/plugins

Tools

cordova-common

- Bugfixes and improved test cases for config munging (especially plist files)
- [DONE] Remove usage of cordova-registry-mapper
- Move platform specific code to platform repos.

cordova-lib

- [DONE] Remove all --no-fetch code
- Remove support for dependencies in config.xml (they belong in package.json)
- [DONE] Consider deprecating [Context.prototype.requireCordovaModule](#)
- Add install/setup step that sets up the platforms
- Review/Prune platform subcommands (check, update, nosave)

cordova-fetch

- [DONE] Make fetching code not slow and really buggy
- Look into reducing supported options (link, ...)

cordova-serve

- ...

cordova-create

- Refactoring/cleanup as proposed by raphinesse

- Allow templates to include .gitignore files

cordova-cli

- Add `cordova install` command
- [DONE] Deprecate `cordova plugin search` command
 - <https://github.com/apache/cordova-lib/pull/671>
 - <https://github.com/apache/cordova-cli/pull/306>
- Migrate to options parser that supports subcommands and generates usage text (yargs?)
- Remove callback parameter of main CLI method

plugman

- Migrate plugman-cli to options parser that supports subcommands and generates usage text (yargs?)

cordova-weinre

- DEPRECATE

Platforms

cordova-android

- Add support for Kotlin
- [DONE] Add support for adaptive icons
- [DONE] Any bugfixes to deal with Android P
- [DONE] Keep support for Android 4.4 (Kitkat)
- Bring splashscreen plugin inside
- Bring statusbar plugin inside
- Bring whitelist plugin inside
- Improve requirement checks
- Add HTTP whitelist support, mirroring iOS ATS

cordova-ios

- [DONE] Ensure automatic provisioning works properly
- [DONE] Ensure Entitlements work properly
- [DONE] Ensure Swift plugins work
- [DONE] Drop support for iOS 9

- [DONE] Add support for iOS 12
- Bring splashscreen plugin inside
- Bring statusbar plugin inside
- Bring WKWebView plugin inside (bridge webview plugin, change webview during runtime)
- Add support for background colour preference in WKWebView
- Use Launch Storyboard as default in new projects
- [DONE] Improve CocoaPods Support
 'Pods/Pods' breakage, does not build on Xcode 10 beta
<https://issues.apache.org/jira/browse/CB-14057> and
<https://github.com/apache/cordova-ios/issues/394>

cordova-windows

- Remove all non-Win10 code ([WIP PR](#))
- Encourage building PWAs and publishing to Windows Store that way?

cordova-browser

- ...

Client

cordova-js

- [DONE] ~~Decide what to do about~~ Remove the browserify flag
 - Browserify was intended to package all the js together so it could be loaded quicker. Tests prove that even with all of our plugins loading, the time between the page starting to load and deviceready only changes by 50ms.
 - Browserify is the reason that every platform has to pack it's own cordova-exec into every project it adds, so that the browserify can pack cordova-js+platform-exec+platform specific plugin code.
 - Browserify has sat behind a flag for years, and has been used in less than 0.001% of the builds in that time. This means we can't simply make the flag the default, because given the number above, this code still seems untested.
 - Browserify does not play well with other transpilers, like babel, rollup, or webpack. We should be trying to play well with as many npm/node-js based build tools as possible.
 - Suggestion: drop browserify flag, and remove all conditional code

cordova-app-hello-world

- [DONE] Update to properly handle iOS safe areas in WKWebView
- Update to reflect latest best practices

cordova-template-reference

- DEPRECATE (merge into cordova-app-hello-world)