

A dramatic landscape featuring dark, silhouetted mountains in the foreground and middle ground. The sky is filled with a vibrant sunset or sunrise, with a gradient of orange and red hues. Wispy, white and orange-tinted clouds are scattered across the sky, catching the light from the sun. The overall mood is serene and powerful.

Qubit.

- Integrating Qubit with your continuous delivery process

January 2020

## Overview

Running a personalization program alongside the day-to-day development of an eCommerce platform can be a tricky process to manage. How do you ensure releases don't clash? How do you run integration tests against personalization code interacting with your platform logic?

Having the right toolchain is key here and provides the pathway to developing, testing, and releasing with confidence.

## Developing locally

A key focus over the last couple of years at Qubit has been empowering developers and helping our customers integrate our technology into their existing workflows and best practices.

As a further milestone, in 2018 we released [Qubit CLI](#) and [packages](#) to enable developers to build Qubit Experiences using their existing toolchain. One of the key advantages to this enhancement is that code can now be pulled down to a local machine and checked into a main codebase. Developers can then work in their editor of choice and run code locally via our hot-reloading build system and Chrome extension.

We have seen great uptake in these features since launch, with over 3500 experiences developed via the CLI across our clients and partners.

Since migrating Qubit experiences to modules, they can be managed like any other NPM package, for example, requiring them as dependencies. You can also develop unit tests against them using your framework of choice—we use [jest](#).

## Automating

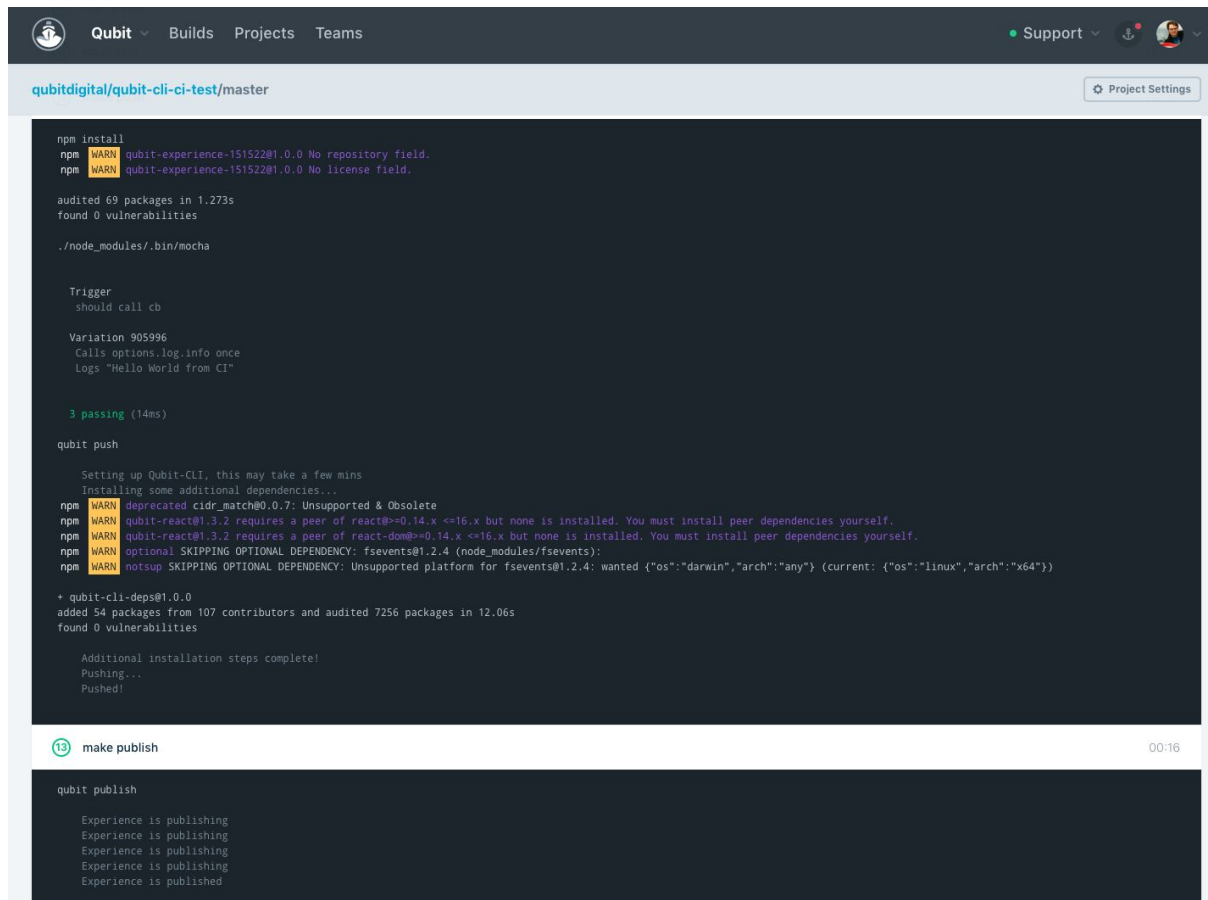
Automating your test suite is the natural next step. To meet this need, we are pleased to announce support for Qubit CLI inside your CI/CD environment.

The feature is small, but a powerful one - you can now get a long-lived token from the Qubit CLI (qubit token). When set as the QUBIT\_TOKEN environment variable, it will be used to authenticate when managing experiences and the hosted private NPM registry we provide for packages. You can also revoke a token via `qubit revoke`.

See [Generating a Token For CI/CD Integration](#) for details.

With this capability, it is now possible to hold your experiences in a repo along with any tests written against them and have your build system ([Codeship](#), [Jenkins](#), [Google Cloud](#)

[Build](#), etc) install any dependencies, run your test suite, then, if they all pass, push the code up (qubit push). If you wanted, you could even publish (qubit publish).



```
qbitdigital/qbit-cli-ci-test/master Project Settings

npm install
npm WARN qbit-experience-151522@1.0.0 No repository field.
npm WARN qbit-experience-151522@1.0.0 No license field.

audited 69 packages in 1.273s
found 0 vulnerabilities

./node_modules/.bin/mocha

Trigger
should call cb

Variation 905996
Calls: options.log.info once
Logs: "Hello World from CI"

3 passing (14ms)

qubit push

Setting up Qubit-CLI, this may take a few mins
Installing some additional dependencies...
npm WARN deprecated cidr_match@0.0.7: Unsupported & Obsolete
npm WARN qbit-react@1.3.2 requires a peer of react@=>0.14.x <=&16.x but none is installed. You must install peer dependencies yourself.
npm WARN qbit-react@1.3.2 requires a peer of react-dom@=>0.14.x <=&16.x but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.4 (node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.4: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

+ qbit-cli-deps@1.0.0
added 54 packages from 107 contributors and audited 7256 packages in 12.06s
found 0 vulnerabilities

Additional installation steps complete!
Pushing...
Pushed!

13 make publish 00:16

qubit publish

Experience is publishing
Experience is publishing
Experience is publishing
Experience is publishing
Experience is published
```

*Example of running unit tests, pushing the code, and publishing from within CodeShip.*

This feature isn't just for unit testing though. You can now control experiences programmatically via a CLI, unlocking use cases such as automatically publishing alongside your automated deployment of major site changes, or scripting an automatic traffic split switch for an experience.

We are excited to see what interesting use cases our customers and partners find for this capability and keen to hear your feedback for what other capabilities you would like to see in our developer toolset.

Don't hesitate to contact Qubit if you have any questions about the content of this article or wish to provide feedback.