

Evergreen Sprouts

Make it green from the start!

Daniel Mrose

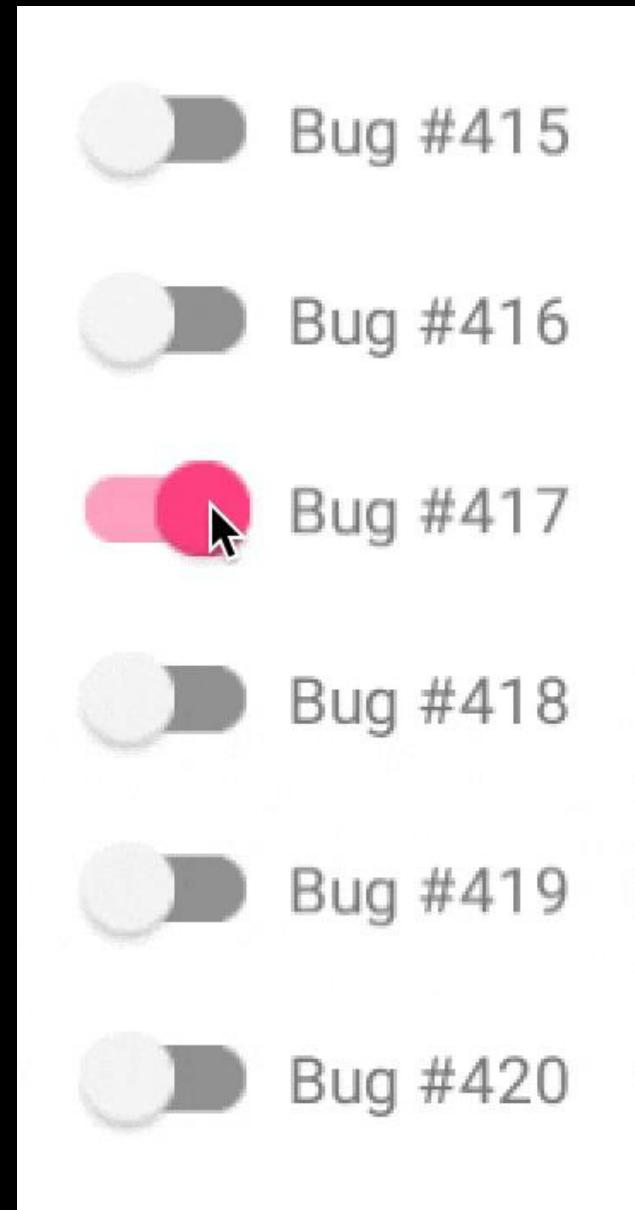
Mark J. Becker

Andrei Dulvac

Andrei Tuicu

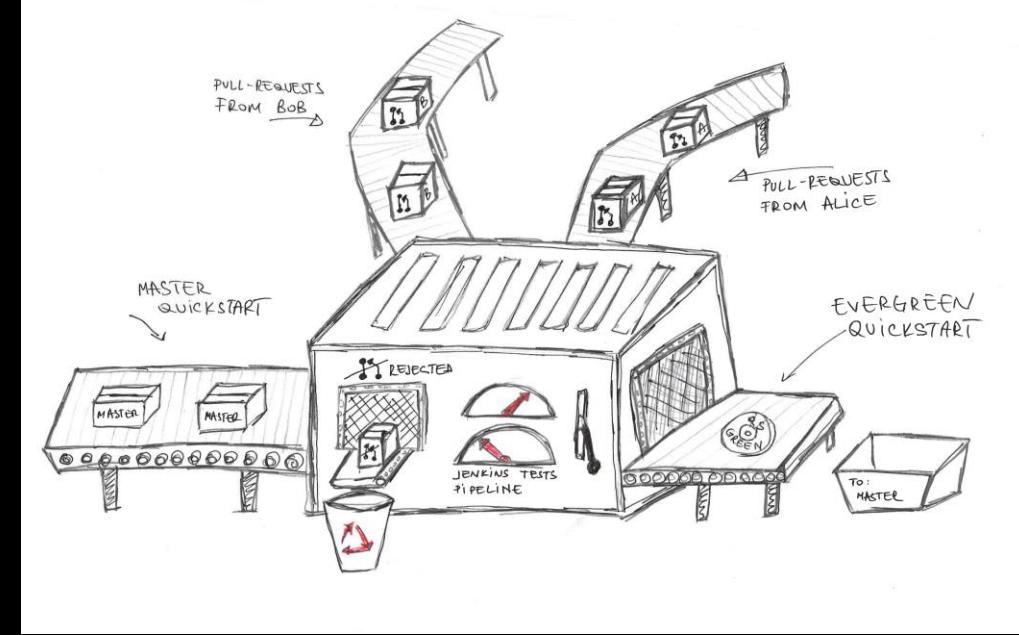
Motivation

- Overall:
 - Rules/KPIs/Process steps need to be enforced by people
 - Long UI/Feature Freeze
 - Complex and Decoupled Quality Process
- Between teams:
 - Preventable bugs are introduced (especially UI – Evergreen QS can't catch every regression)
 - Bugs found by other teams tests need to be filled and planned, instead of being part of the original ticket effort (preventable effort)
- Inside teams:
 - Mostly waterfall development

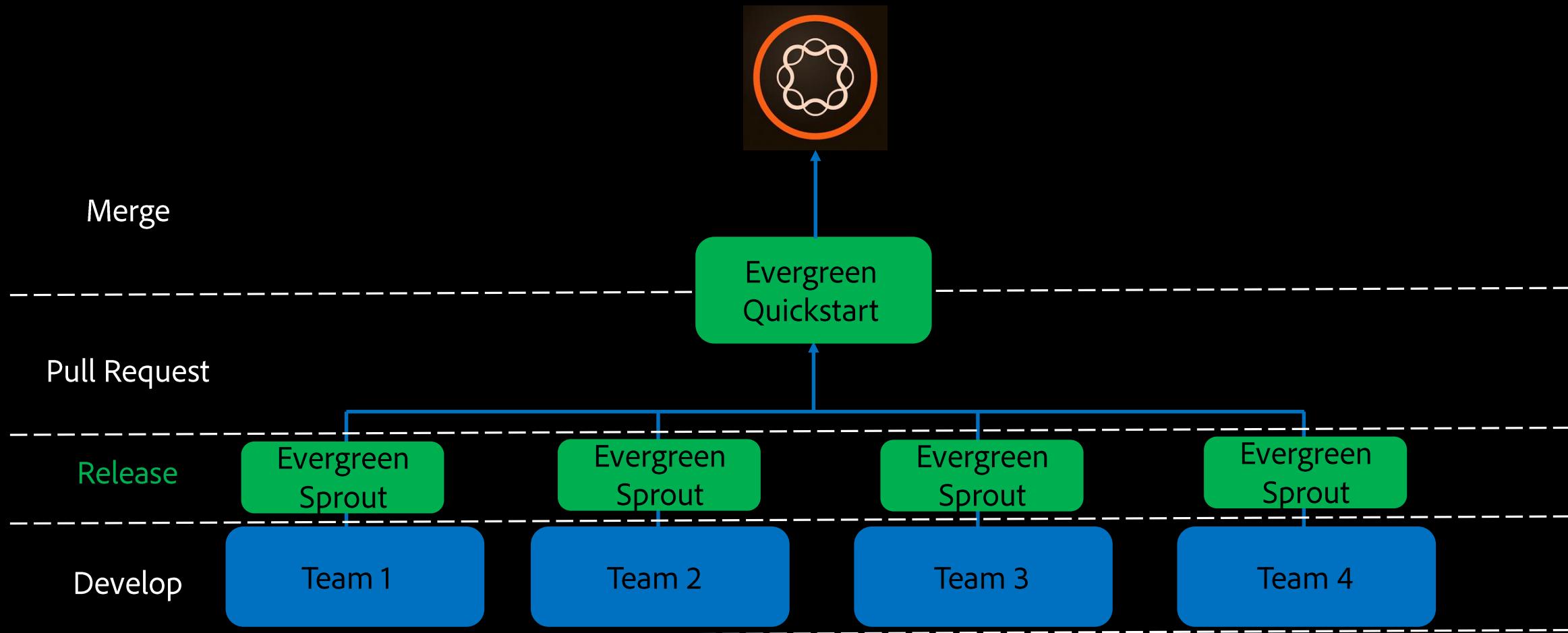


Evergreen Quickstart - Statistics (1 Dec 2016 - 7 April 2017)

- Total pull-requests since Dec 1, 2016 (4 months): 2252
- Caught Regressions: 258 (11.5% of all the PRs)
 - Quickstart build fails 54



Where to act next?

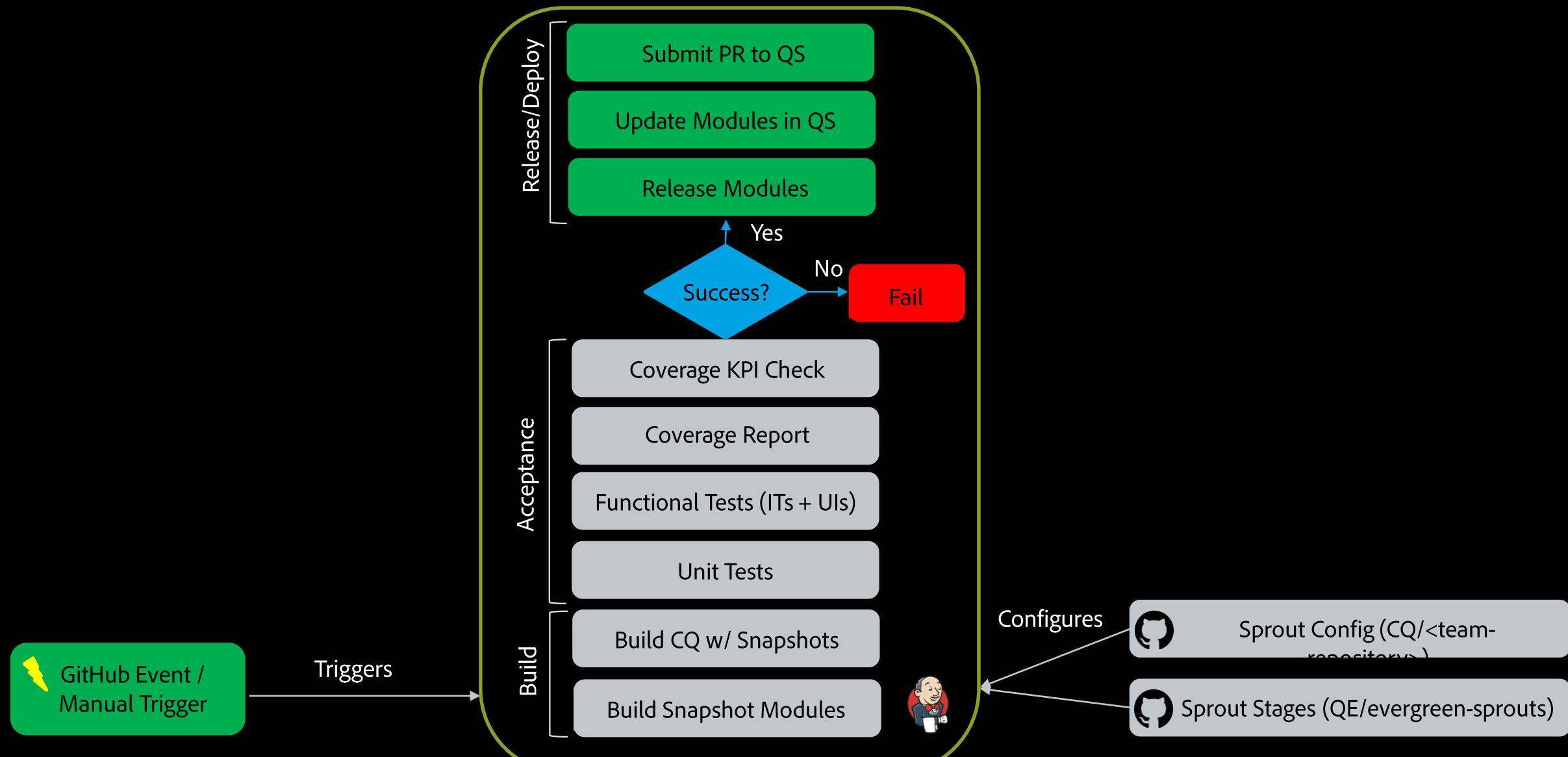


What are Evergreen Sprouts?

- The **Evergreen Sprouts** project aims to complement the **Evergreen Quickstart** project by testing modules that go in the Quickstart automatically, streamlining module releases and pull-requests to Quickstart
- Fully customizable CI pipeline
- Based on the Jenkins pipeline framework



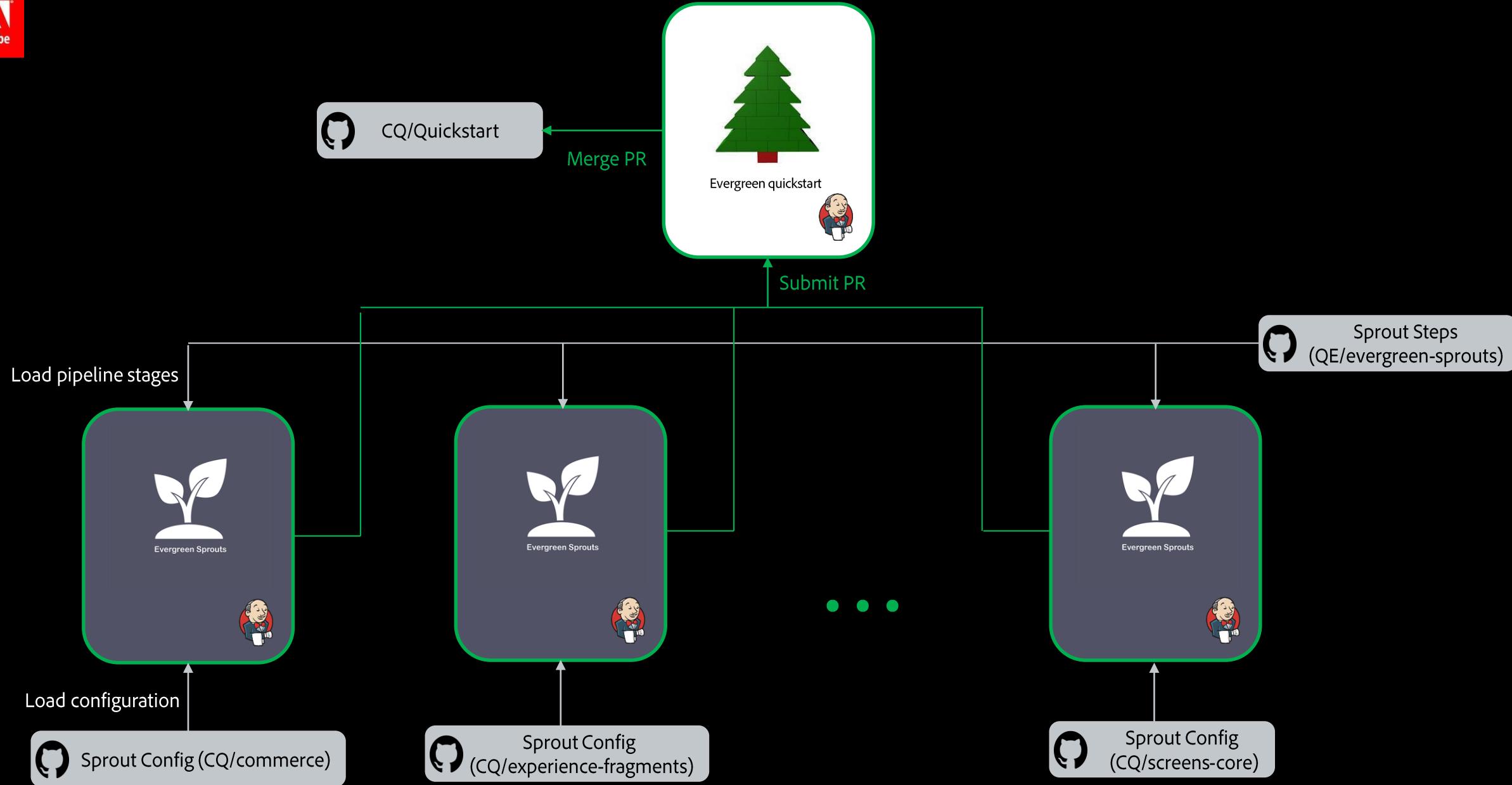
What are Evergreen Sprouts?



Goals

- Make **Evergreen Sprouts** the de facto standard for validating and releasing artifacts
- Ensure that for every module release the **quality KPIs are met** (all tests pass, no preventable bugs are introduced)
- Ultimate goal: Ensure that **every load** is a **potential release candidate**, to allow more frequent AEM releases.





DEMO

What can it do?

- Configuration and stages in Git
- Build Snapshots
- Build QS with Snapshots (6.3 and 6.4)
- Run integration tests (IT & UI) on built (6.3 & 6.4) and prebuilt (6.x) quickstart
- Save artifacts
- Parallel Test Execution
- Build for every push
- Email notifications
- Slack notifications
- Releases
- Run Evergreen Quickstart ITs
- Create Quickstart PR
- Code coverage analysis
- Code coverage KPI check
- Stage and Test Run filtering
- Fully configurable and easily extensible
- Pull Request validation

+ more to come ☺

Evergreen Sprout MVP

- Implement an Evergreen Sprout
- Define a set of stable tests (UT and/or IT and/or UI)
- Release modules through the pipeline after the stable tests passed

Overall results

- One step closer to more frequent QS releases
- Automation becomes the key to scaling the growing AEM development team
- Switch between a reactive to proactive approach to development
- More and stable tests written earlier
- Better quality for every team release
- Clear ownership of code and tests



Thank you!

- Feature Epic: <https://jira.corp.adobe.com/browse/NPR-13439>
- Adoption Epic: <https://jira.corp.adobe.com/browse/CQ-4204391>
- Technical Documentation: <https://git.corp.adobe.com/QE/evergreen-sprout/wiki>
- Wiki Page: <https://wiki.corp.adobe.com/display/cq5/Evergreen+Sprouts>
- Slack Channel: #evergreen-sprouts
- Mailing List: Grp-Evergreen-Sprouts@adobe.com
- Git Repository: QE/evergreen-sprouts