**Titles:**

* Advanced Patterns for Automated UI Testing
* Advanced UI Automation Testing
* Automated Regression Testing - How To Manage Complexity And Stay Green
* It Takes A Village To Write Effective UI Tests
* Automated UI Tests In The Real World
* Writing Maintainable Browser Tests
* Maintaining Effective UI Tests Is A Whole-Team Job
* 5 Keys For Writing Maintainable Browser Tests

Abstract:

Automated browser tests can be a nightmare to write and maintain. Unlike unit tests they’re tough to set up, slow to run, make permanent changes to data in the system, and they’re incredibly brittle. Even a tiny CSS change can result in cascading failures!

You can’t totally avoid these issues, but you CAN manage them by adopting a few key patterns and techniques. This session explores the lessons learned by one product team as they moved from manual regression testing to an automated process using Canopy and Selenium. You’ll go beyond the basics of single-page testing and see real-world suggestions for managing test data, executing complex and multi-page tests, organizing and writing test code, and coping with “permutation explosion”. You’ll also learn techniques for involving the entire cross-functional team in the process.

This session is designed for developers and architects that know how to automate the browser, but are struggling to leverage that knowledge into a suite of maintainable UI tests that provide acceptable ROI. It focuses on patterns and principles, not a specific automation technology.

Notes:

* We’ve learned a ton about how to effectively do UI testing on a complex product, and it totally takes the whole team. Devs have to write testable code, and write resilient tests. QA has to think through how much data gets created in advance, and how much is created specific to each test. Ops has to help with restoring to known states or managing the CI infrastructure.