

Modern Approaches To Testing

“How modern are your approaches?”

Konstantin Schädler

“I don’t tend to create a lot of code on projects;
I just make things work.”

Stuart Mottram

“Have some”

Charles Franklin

“Regression? What is the nature of regression?”

Peter Lieder

“Regression tests. This is only important if you have regressions.”

Peter Lieder

“Actually I don’t give a damn about testing”

David Ireland

“Do it. Just f***ing do it. You muppet.”

Tom Parker

“I like tests”

Matthew Sackman

“Is this your talk?”

Irene Papakonstantinou

“No 😭”

Ceri Storey

Alex Kahl

- Testing is not an end in itself. Always ask yourself why and what you're testing for.
- Testing is approached in an ideological manner. It depends on the circumstances. Like anything else.

Generative Tests

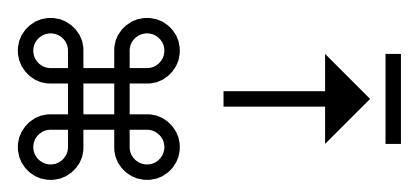
Unit Tests

- Easy (or should be easy)
- A good unit test suite tests the **edge cases**
- What are the edge cases?

Definining Generative Tests

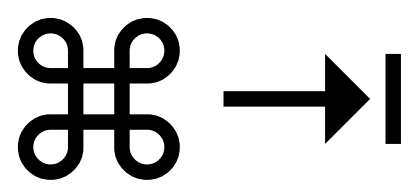
- Randomly generated input
- Define the relationship between input and output

I'm not Sorry



Generative Integration Testing

- Systems consist of many components
- Loose coupling is a myth
- Emergent behaviour is interesting behaviour



This is hard work...

Generalising the Problem

- Generators
- Shrinking
- Properties
- Reproducibility

Tools to Look For

- Anything 'QuickCheck'
 - QuickCheck
 - clojure/test.check and clojure.spec
 - JSVerify
- Gatling / clj-gatling

Post Modern Testing

Soak Testing

goshawkdb

Ask Matthew.

Left as an exercise for
the reader

Post-post Modern Testing

Monitoring

Mutation Testing

Ask Olof.

