

Evolution of Mobile Test Automation at Badoo

From 0 to 45 hours of tests

Sathish Gogineni · 17th of May, 2018



- This is not a pure technical presentation!
- This is not about best practices
- This is not about 7 key steps to successful test automation
- This is a story how we evolved and scaled mobile test automation at Badoo



Sathish Gogineni

Mobile Test Automation Lead to Badoo. Working at Badoo since 2012.
Originally from India. Father of 2 kids.



[@satgog](https://twitter.com/satgog)



<https://www.linkedin.com/in/sathishgogineni/>

Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience

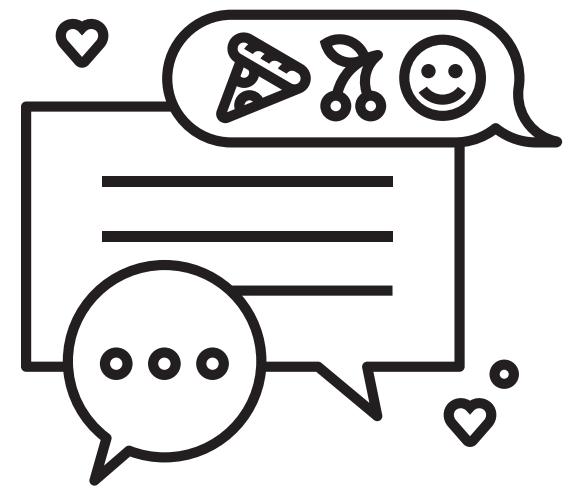


Agenda

- **Tests Reliability**
- **Developers & Manual testers Involvement**
- **100% test coverage**
- **Summary**



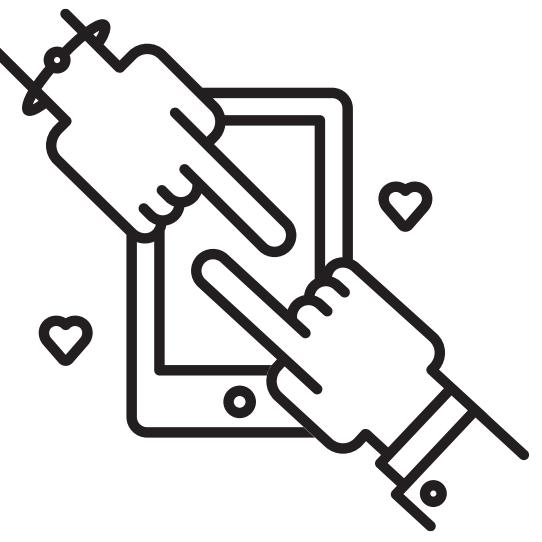
About ❤ badoo



350m
Messages per day



386m
Registered users
worldwide



60m
Monthly active
users



300k
New daily sign ups

Mobile Apps

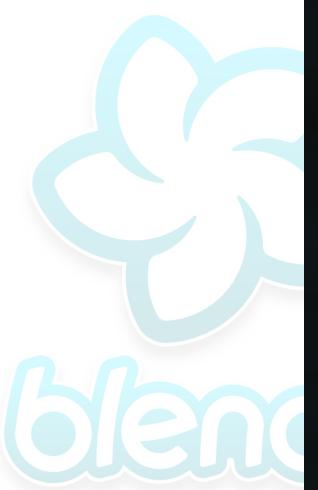
Mobile Apps



Mobile Apps



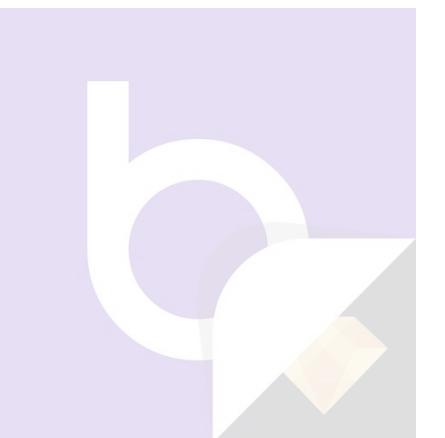
huggle



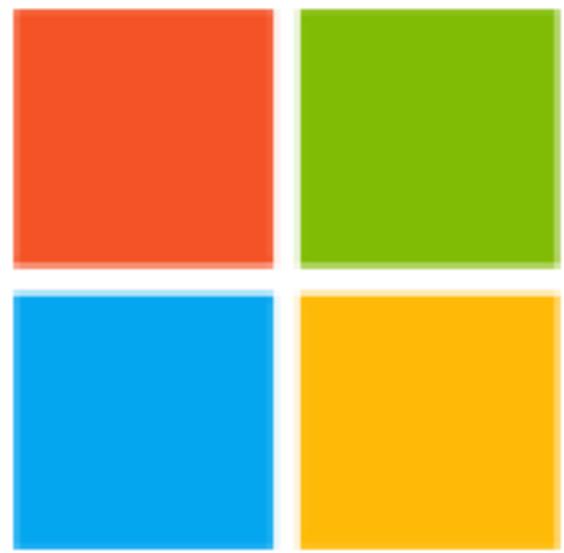
Mobile Apps



HTML



Mobile Apps



Microsoft

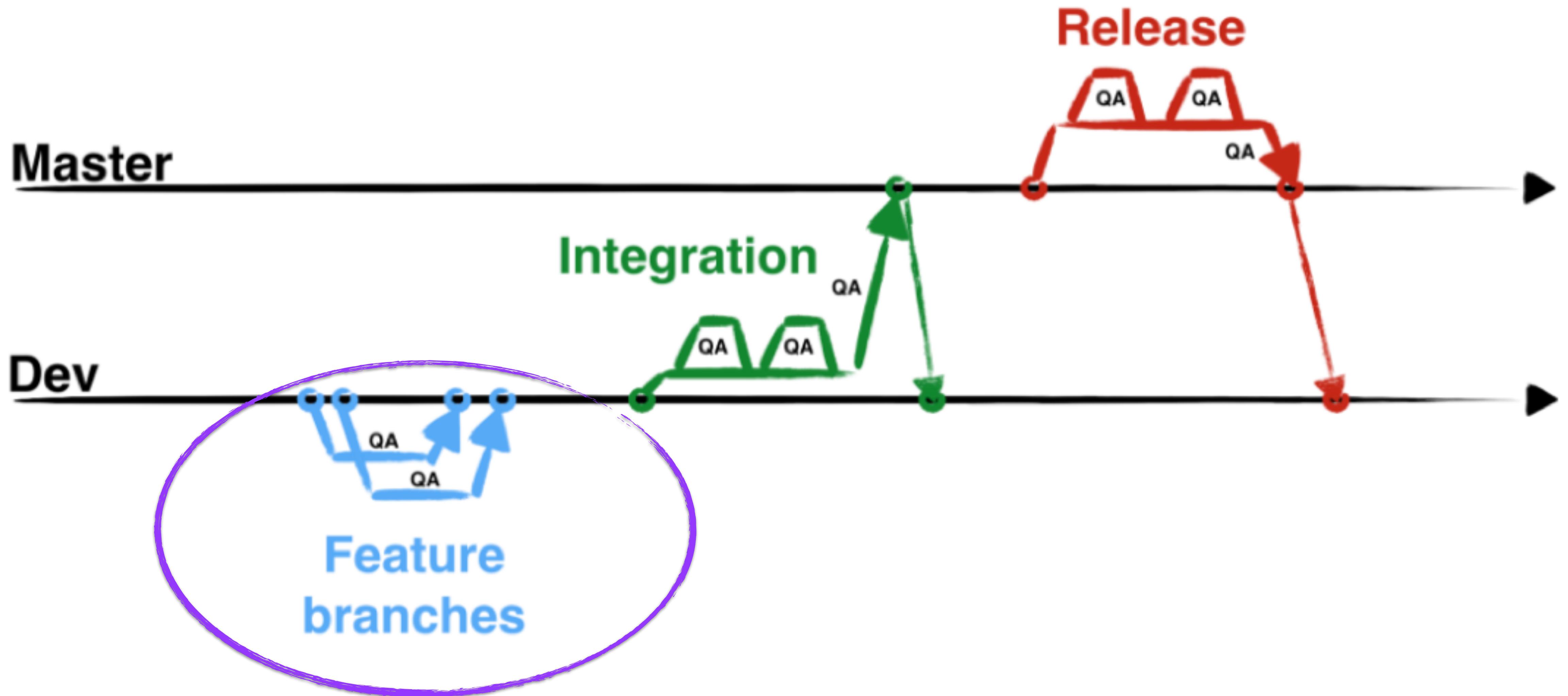
Windows Phone

Is

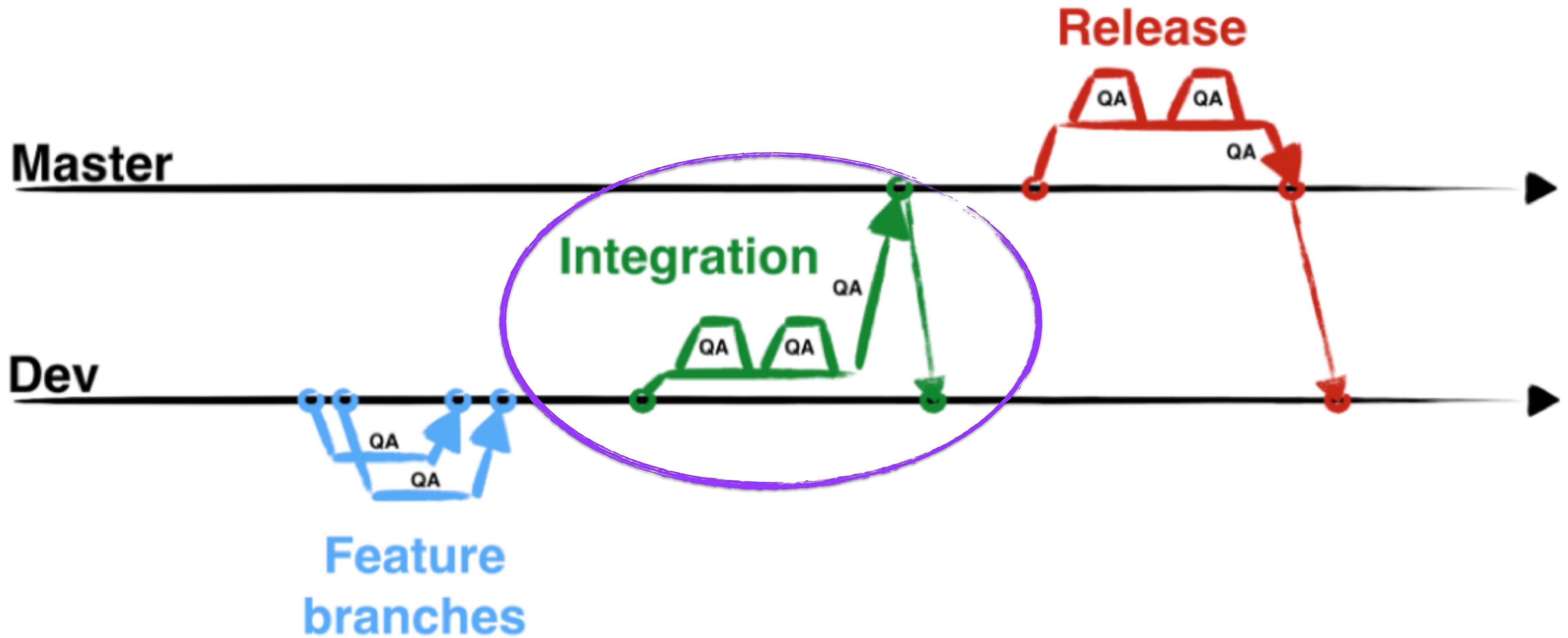
DEAD

Development flow

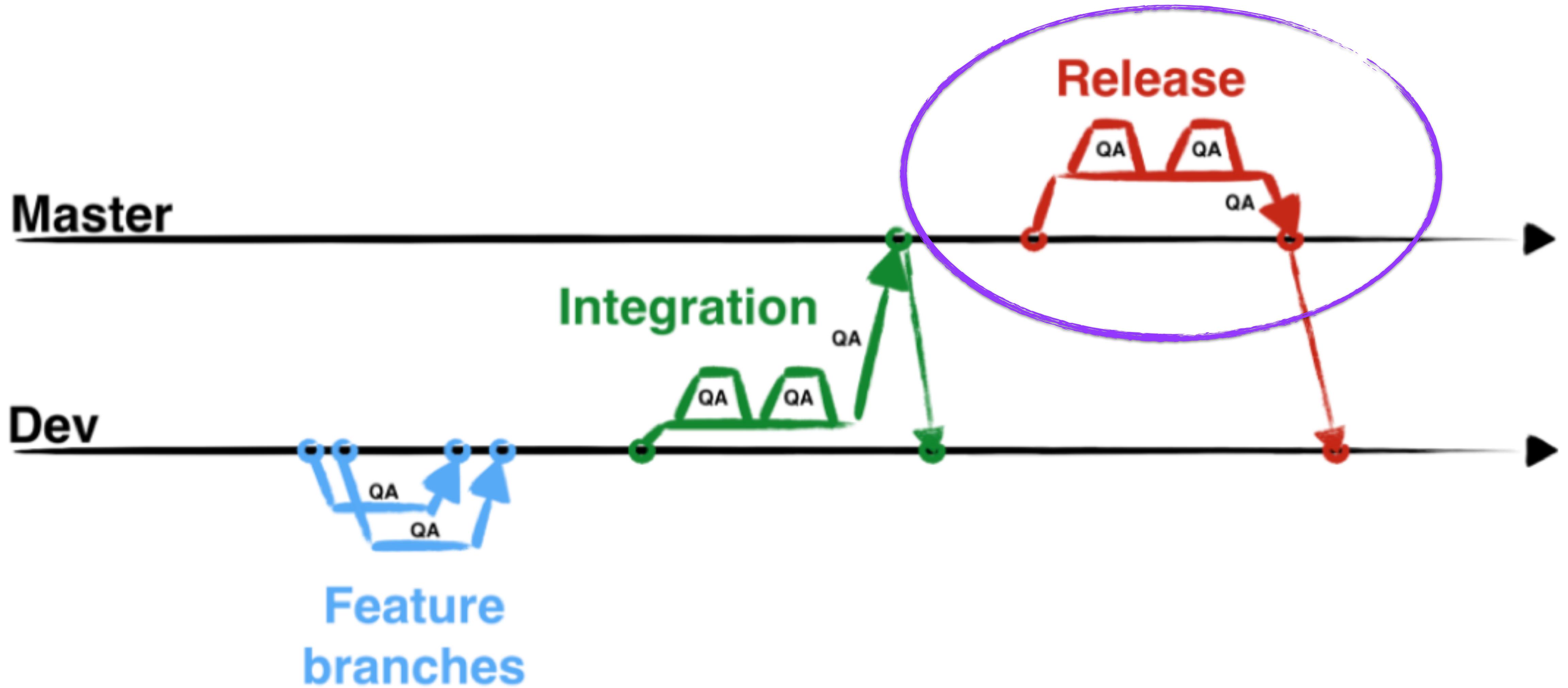
Development flow



Development flow

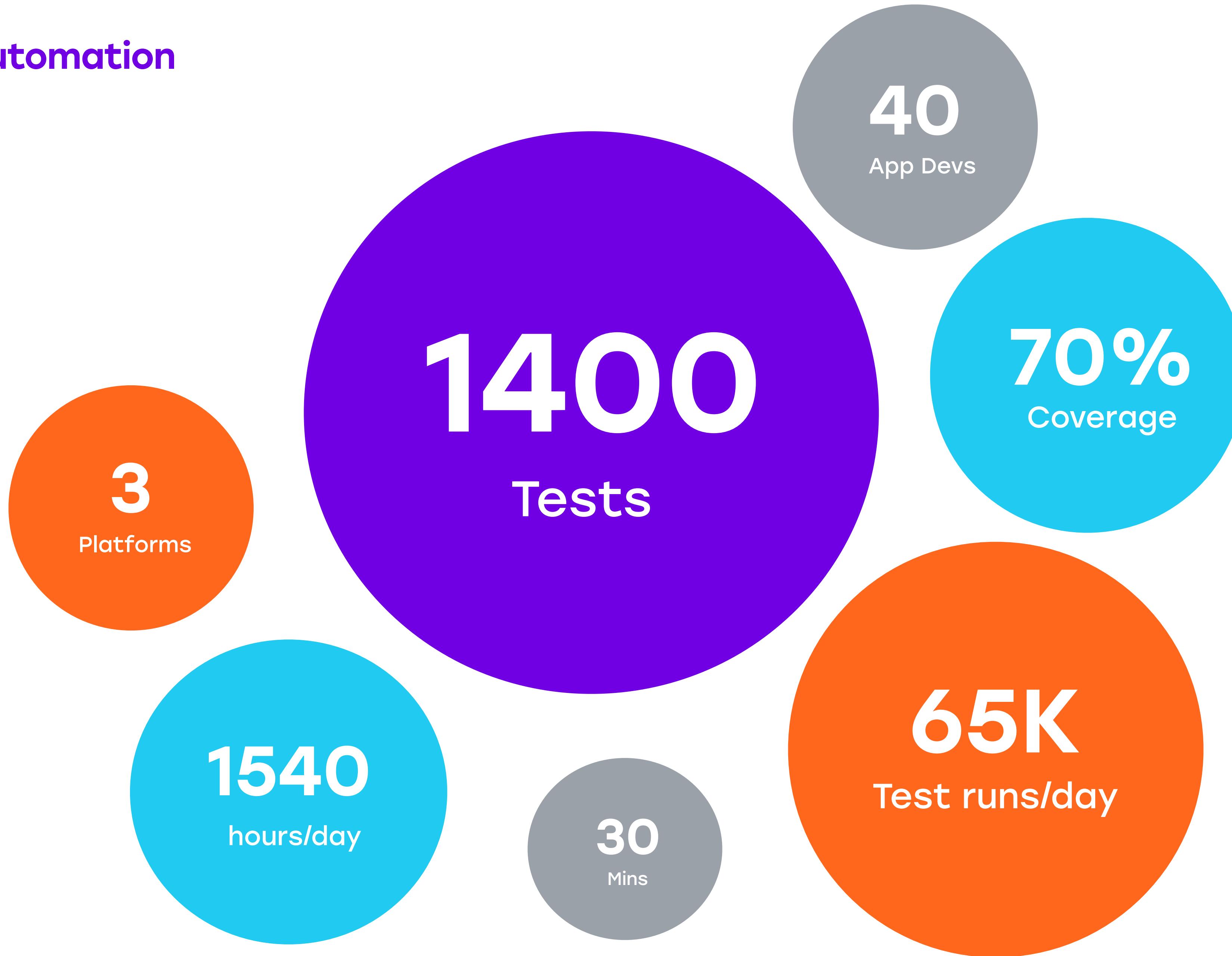


Development flow



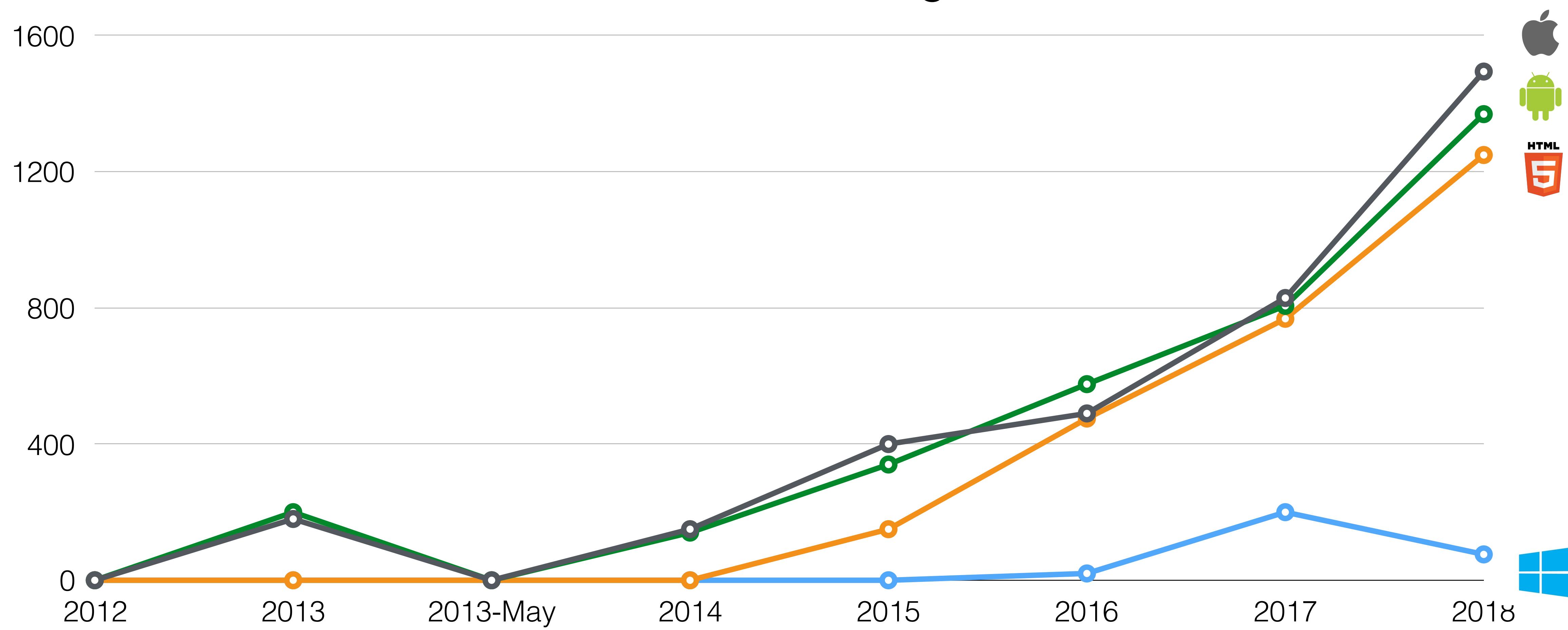
Mobile Test Automation

Mobile Test Automation



Mobile Test Automation

Automated tests growth



Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & Manual testers Involvement
- 100% test coverage
- Summary



Year 2012 Initial Steps

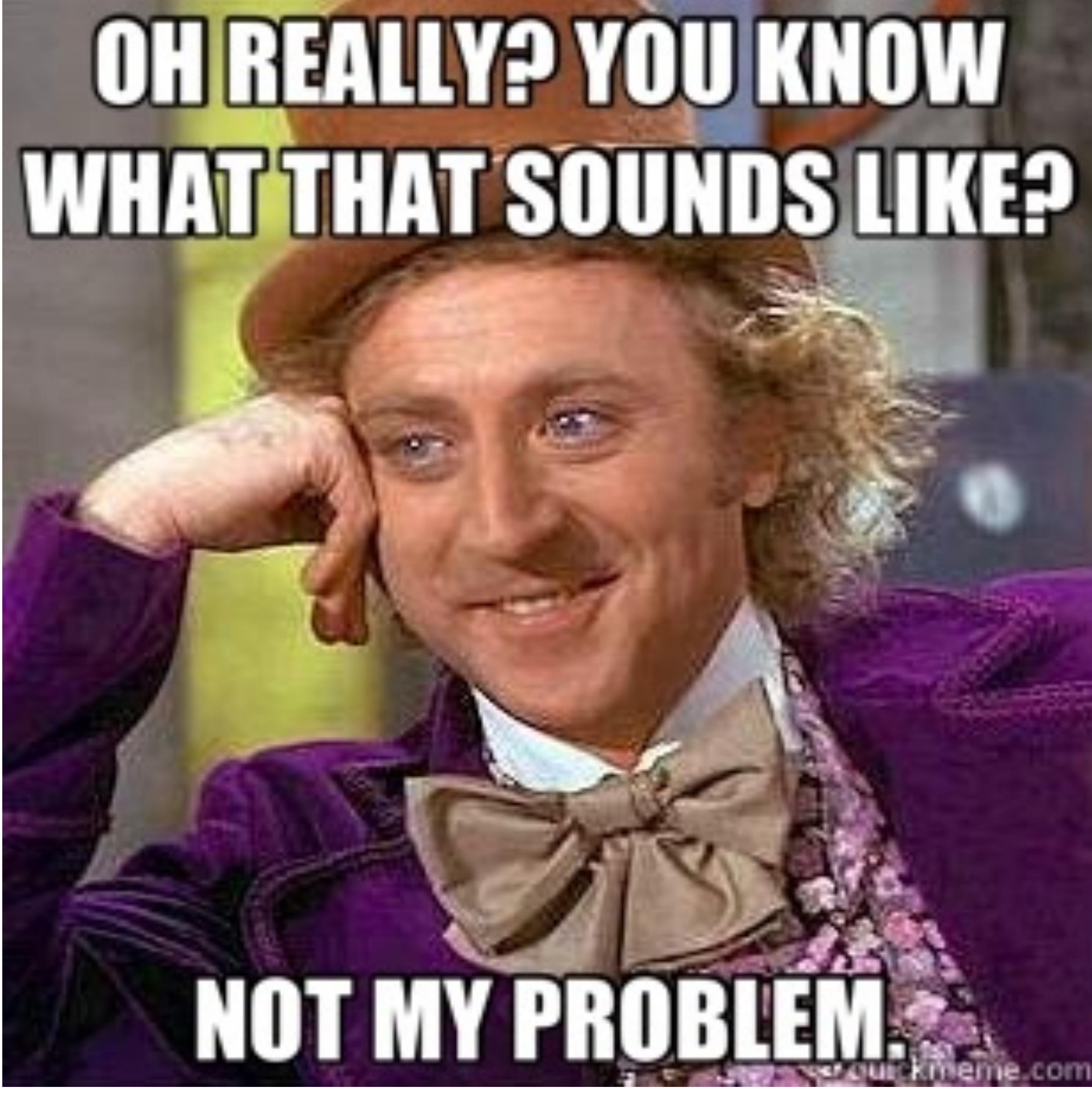
- App downloads are growing
- Mobile teams started growing
- Testing app manually by QA

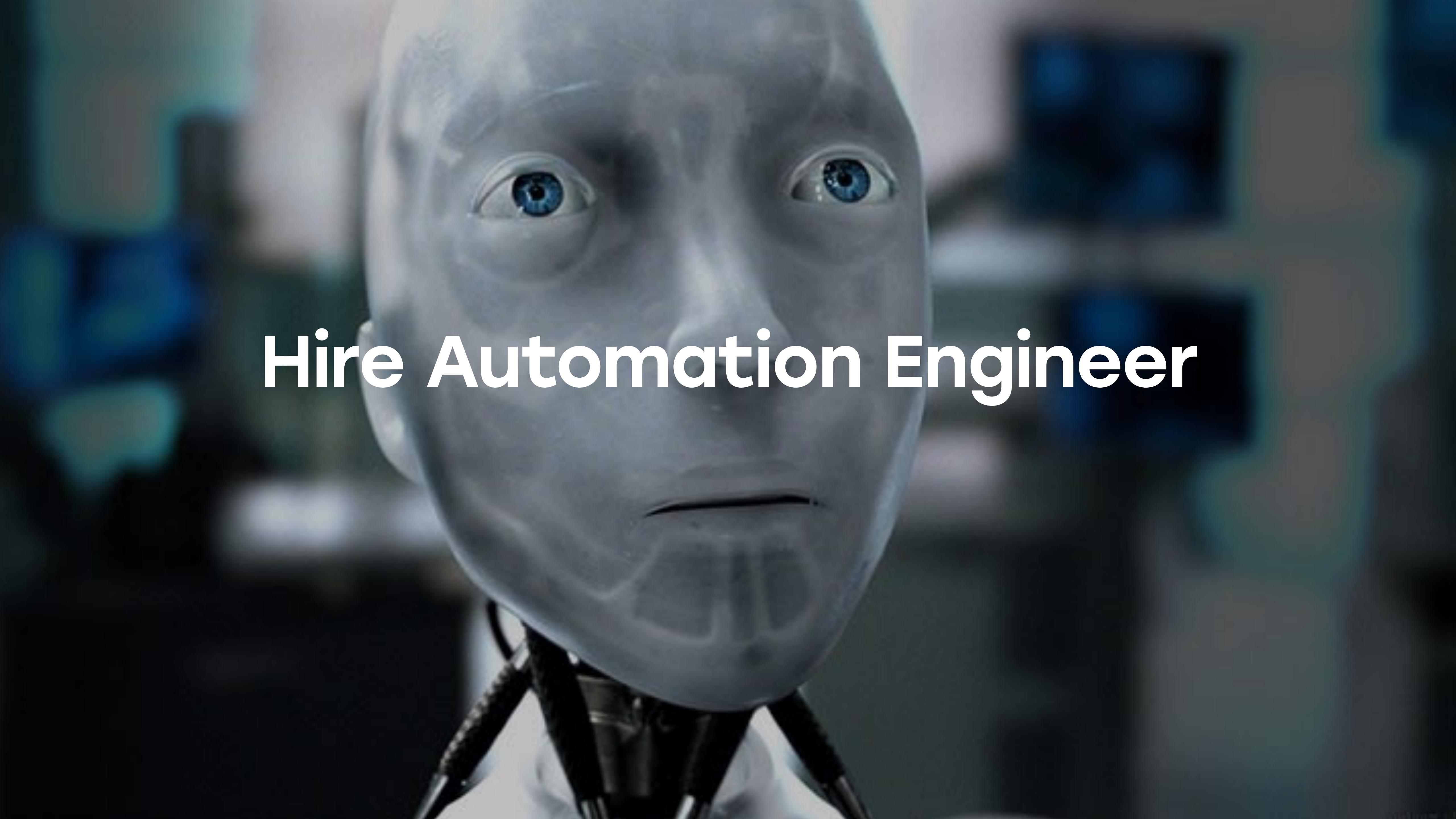


Let's automate the tests!

Year 2012 Test Automation

- Developers: No time
- It's a QA problem!



A close-up, slightly blurred portrait of a silver humanoid robot with blue eyes and a neutral expression.

Hire Automation Engineer

Year 2012 Initial tools



Robotium



Calabash.sh



eggPlant
built by TestPlant



WE HAVE

"AUTOMATED TESTING"



Unified tests Android & iOS ?

Unified tests for Android & iOS ?

- App code bases are different
- App behaviour is not exactly the same!
- Different features on each platform!



Unified tests for Android & iOS ?

- Can't use both App programming languages!
- Can't get support from developers!



Unified tests for Android & iOS ?

- **Test design: same!**
- **Test scope: same!**
- **Test data set-up: same!**
- **Server interaction: same!**



Unified tests for Android & iOS ?

- **Actions: different!**
- **Verifications: different!**



Unified tests for Android & iOS ?

- Efforts can be optimised
- Team can be unified



Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & Manual testers Involvement
- 100% test coverage
- Summary



Cross Platform Test Automation: Calabash



Calaba.sh



* 2018: It is official, Calabash no longer has support from Xamarin/Microsoft

Cross Platform Test Automation: Why Calabash ?

- Not many open source tools for mobile platforms (Year 2012)
- Appium was unstable
- Best among iOS automation tools

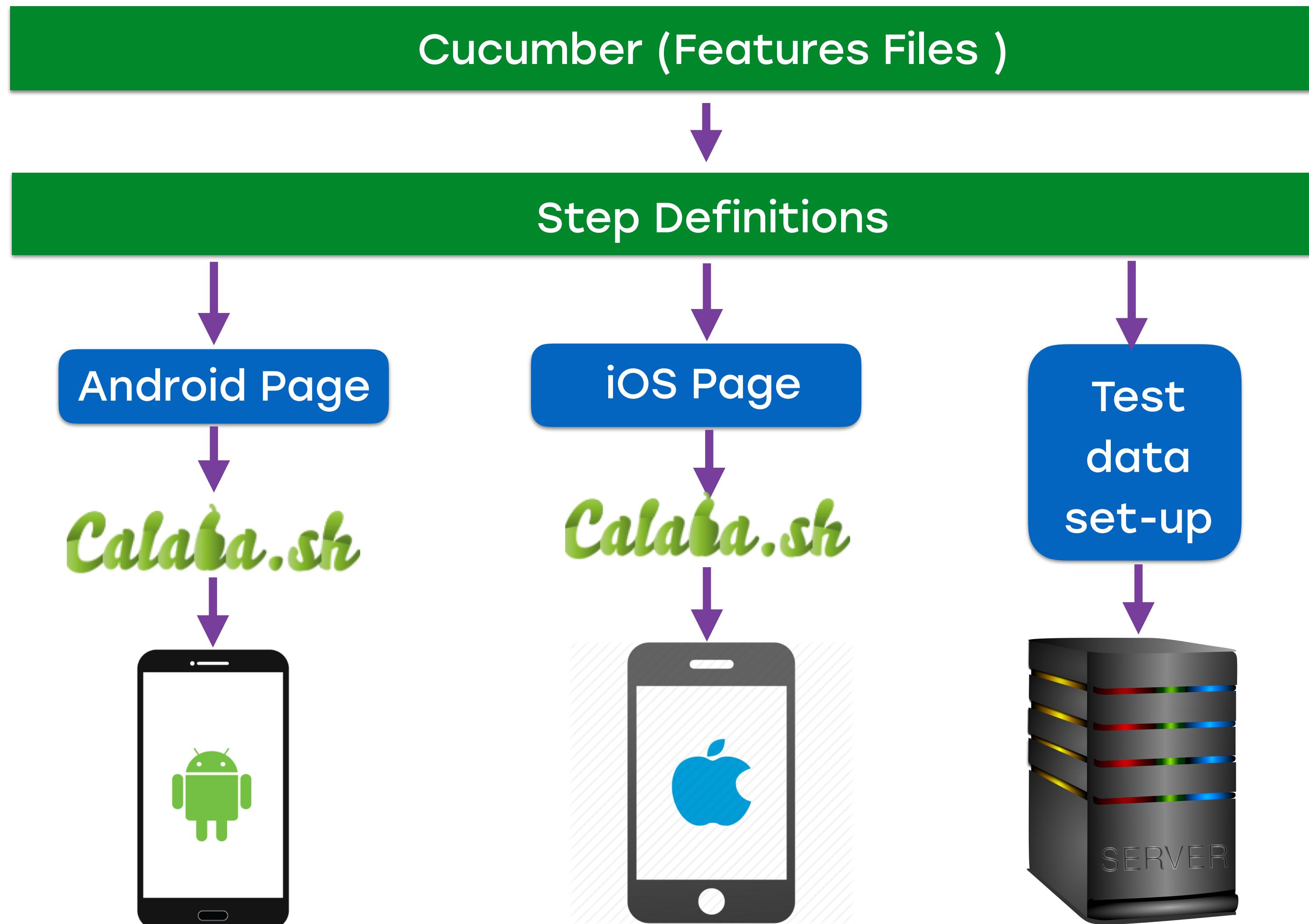


A team is born!

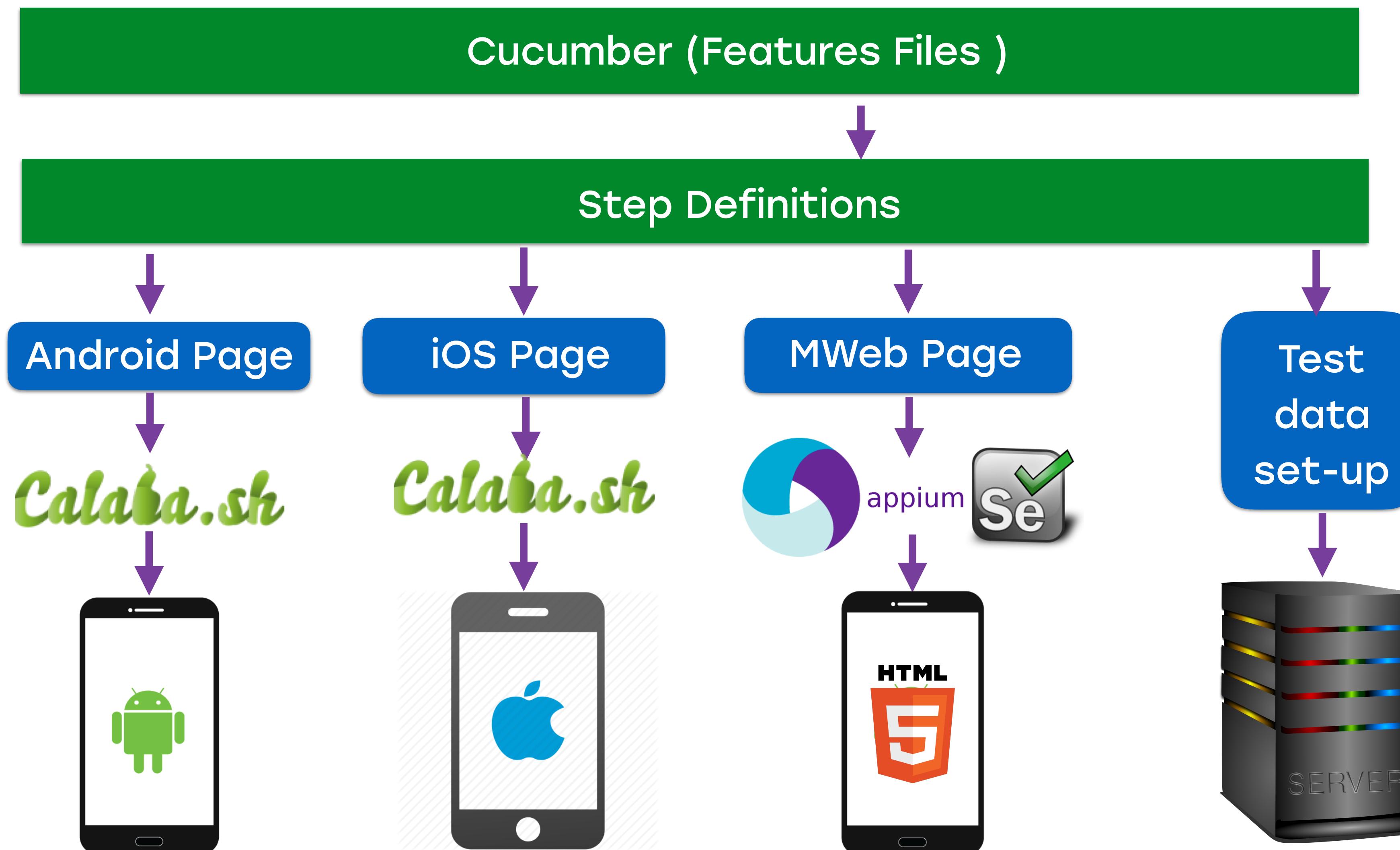
Mobile Test Automation: Goals

- Focus only on automating the tests
- Initial goal is to cover the backlog!
- Responsible for integrating tests with development process

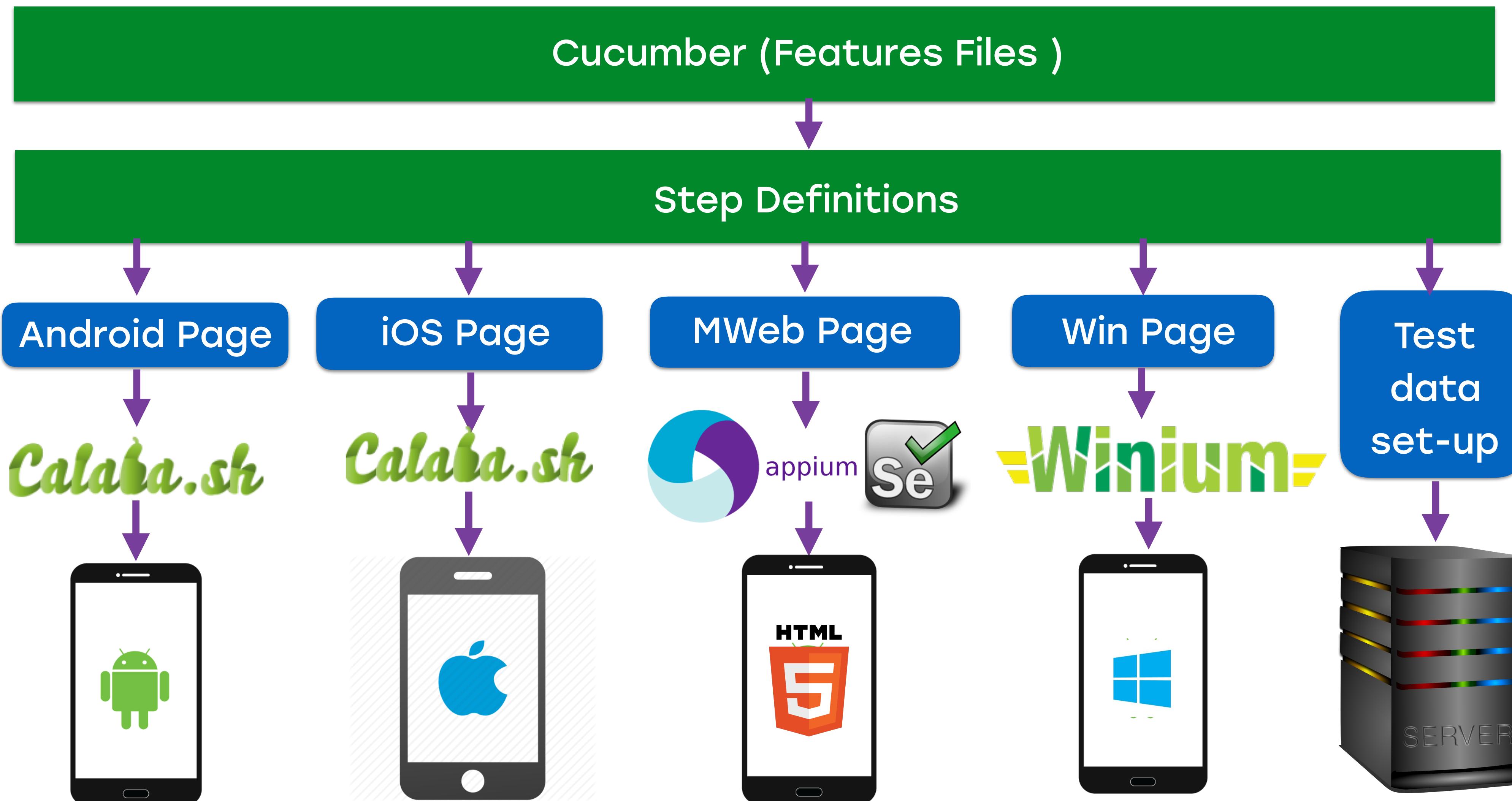
Cross Platform Test Automation: Architecture



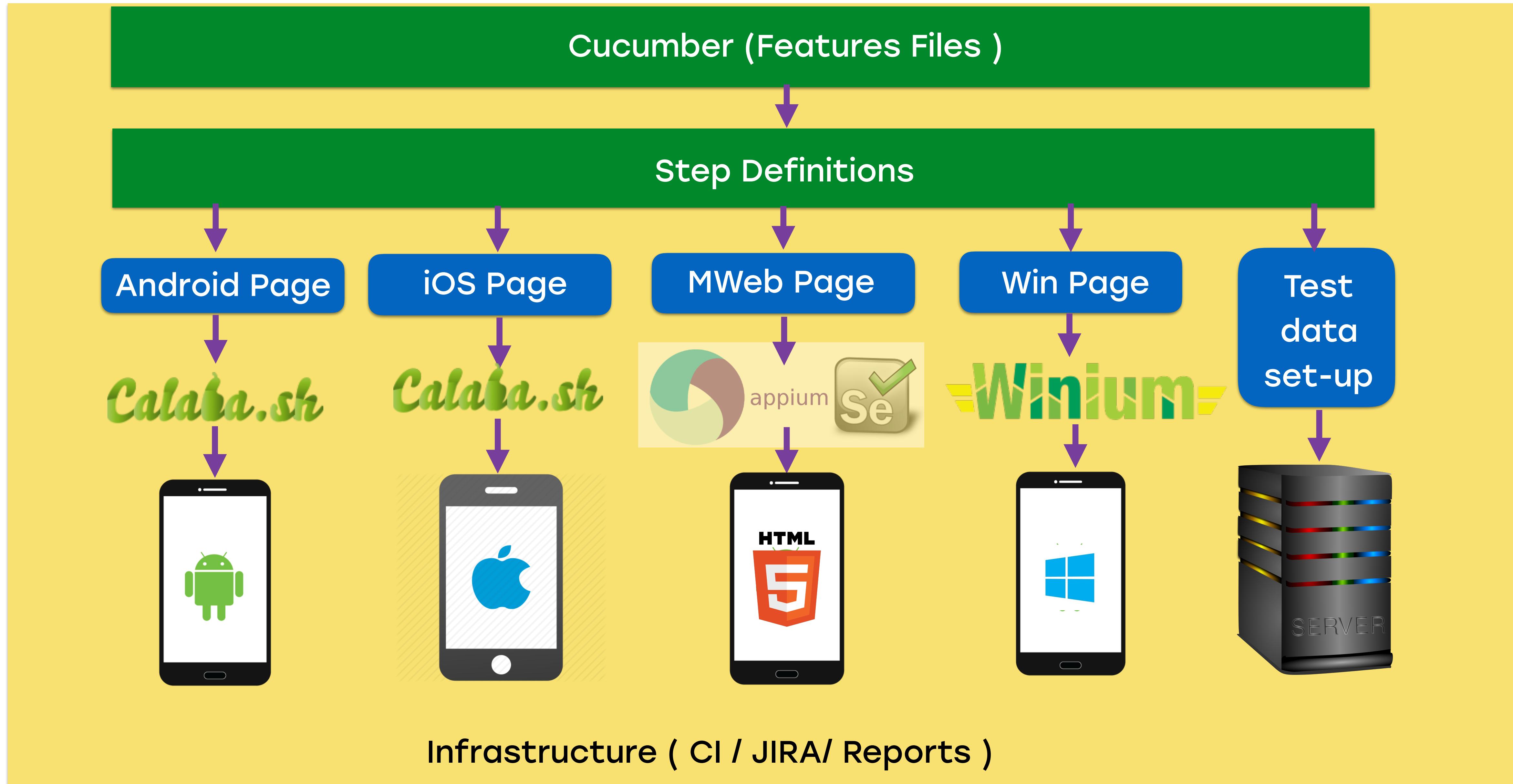
Cross Platform Test Automation: Architecture



Cross Platform Test Automation: Architecture



Cross Platform Test Automation: Architecture



Mobile Test Automation: Early Achievements!

- Responsible for one thing: Test Automation
- Tests running on CI
- Test count is growing
- Regression bugs through automation

Mobile Test Automation: Early Achievements!

- Knowledge sharing
- Learning curve!
- Team work

Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & QA Involvement
- 100% test coverage
- Summary

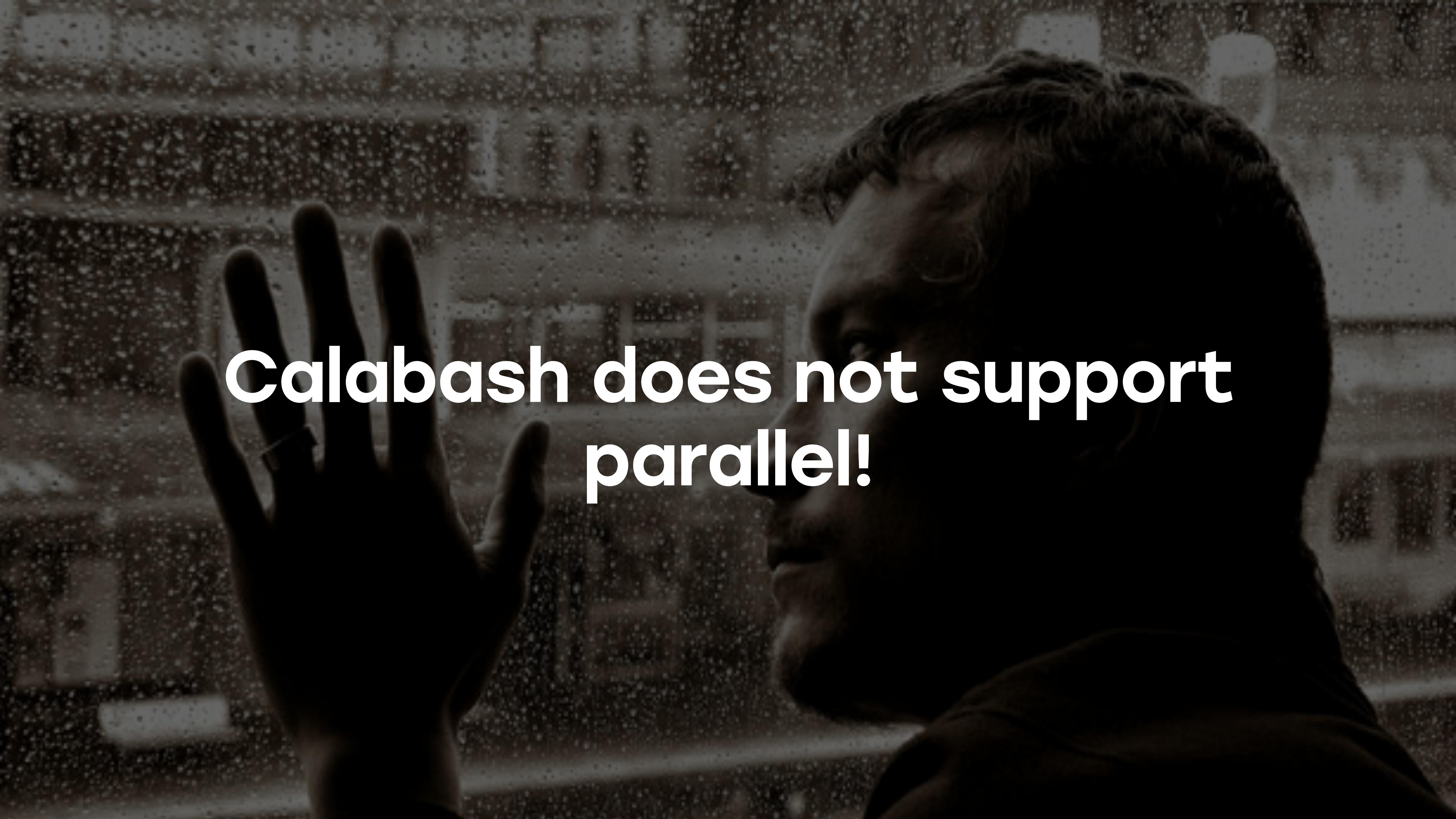


Speed & Resilience: Problem

- Test results are slow
- Runs on a single thread
- Runs nightly
- Runs only on release



Runs tests in parallel!!



Calabash does not support
parallel!

Android

Speed & Resilience: Android

- “**parallel_calabash**” Ruby gem on top of Calabash
- Device lab is expanded
- Tests run in parallel on multiple devices



[https://github.com/rajdeepv/
parallel_calabash](https://github.com/rajdeepv/parallel_calabash)

Hired

More speed!

Speed & Resilience: Android

- Physical space



Speed & Resilience: Android

- Physical space
- Device maintenance



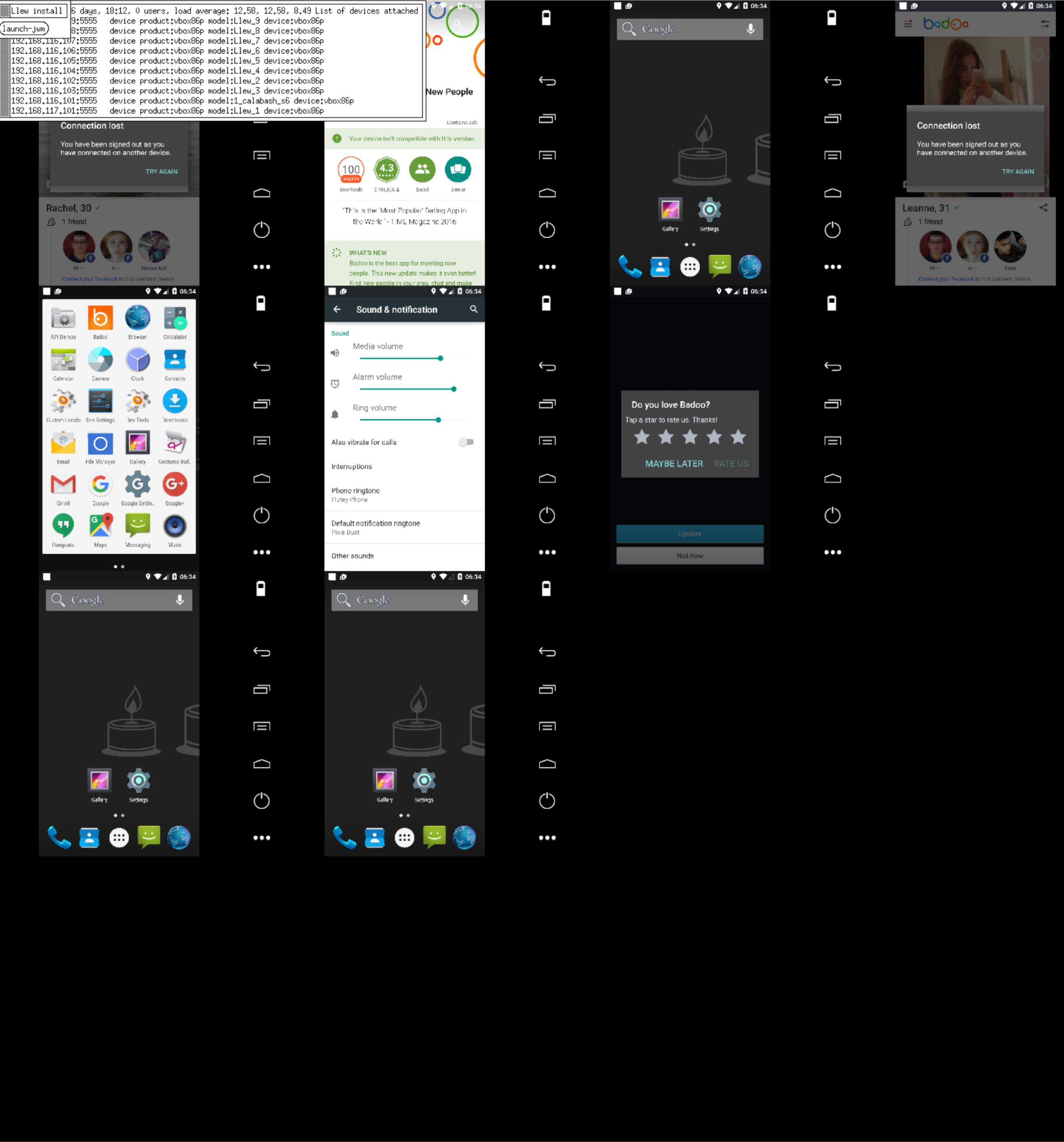
Speed & Resilience: Android

- Physical space
- Device maintenance
- Cable hell



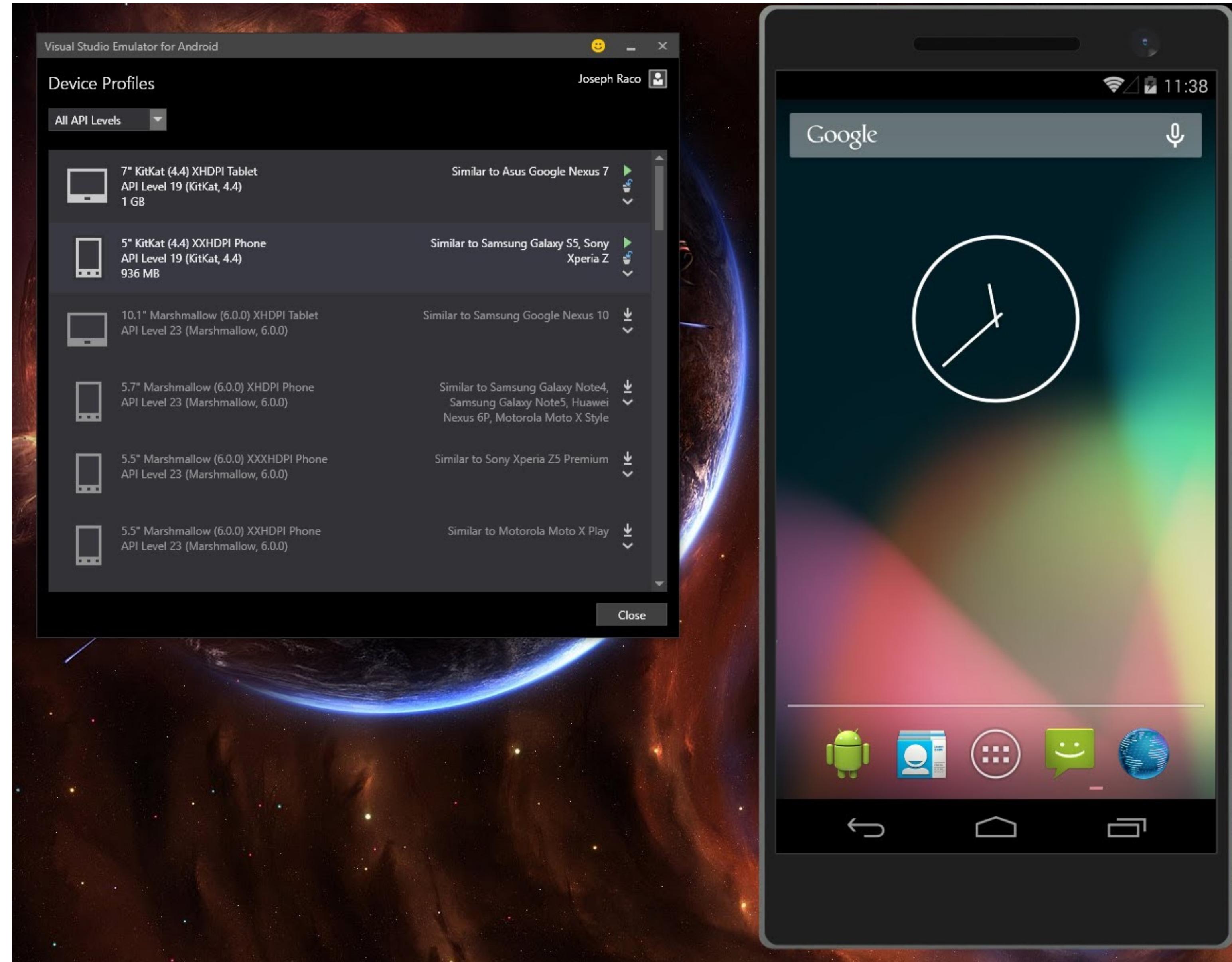
Speed & Resilience: Android

- Initially Genymotion
- Had Google play
- Fastest emulator
- Dockerise the deployment



Speed & Resilience: Android

- **Switch to Google emulators**
- **Smaller image**
- **Headless**
- **No license cost**



Speed & Resilience: Android

- **Linux server (56 Core) can run 32 emulators (Cost £6k)**
- **Scalable**



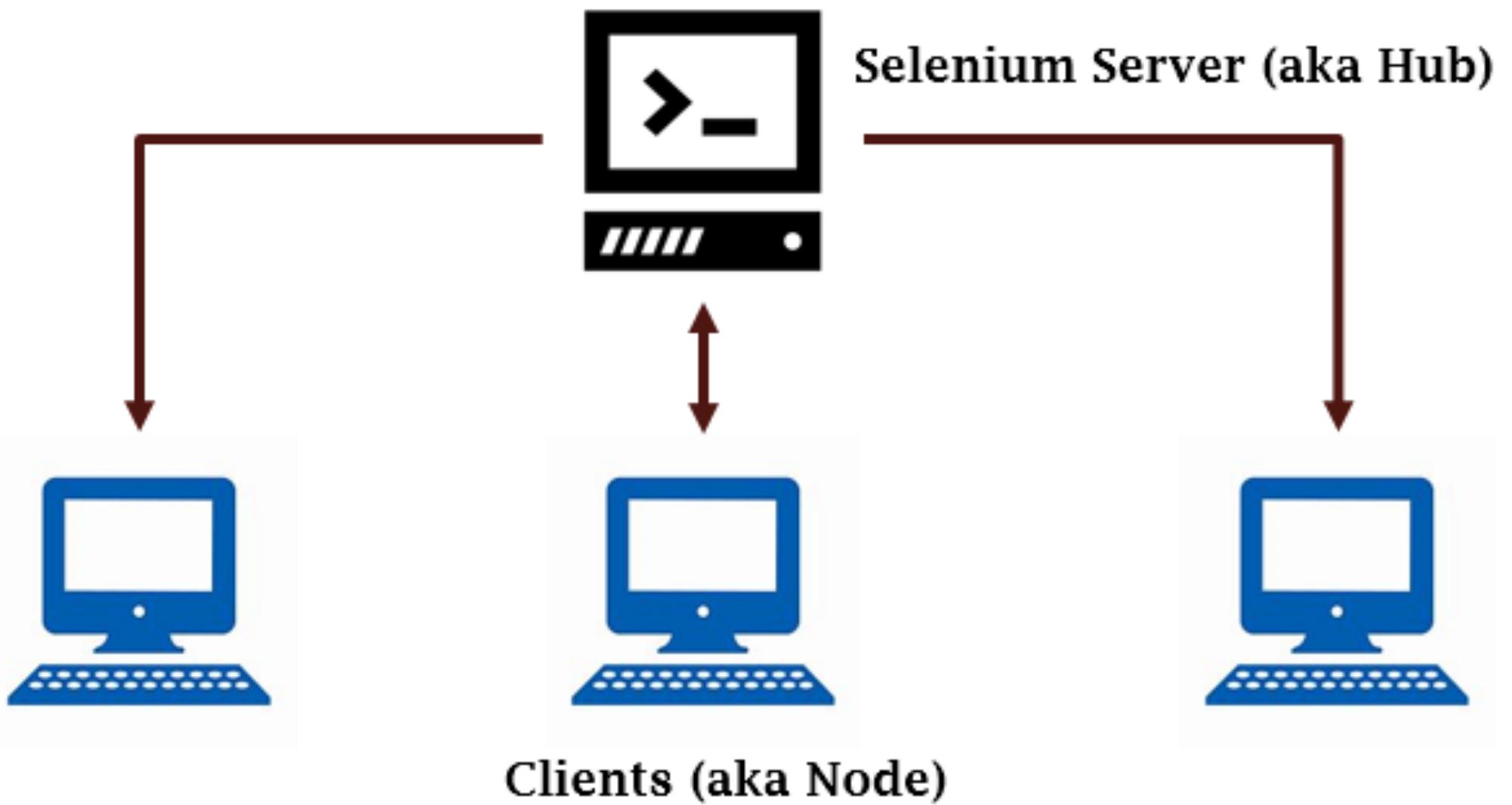
Speed & Resilience: Android

- 1100 tests in 20 minutes
- Tests run for every feature branch on emulator
- Tests run on devices nightly

Mobile Web

Speed & Resilience: Mobile Web

- Extended “parallel_calabash” for mobile web
- Selenium grid makes things easy
- Fewer resources compared to Android



Speed & Resilience: Mobile Web

- 1100 tests in 15 minutes
- No manual testing before release
- Mobile web release every day

iOS

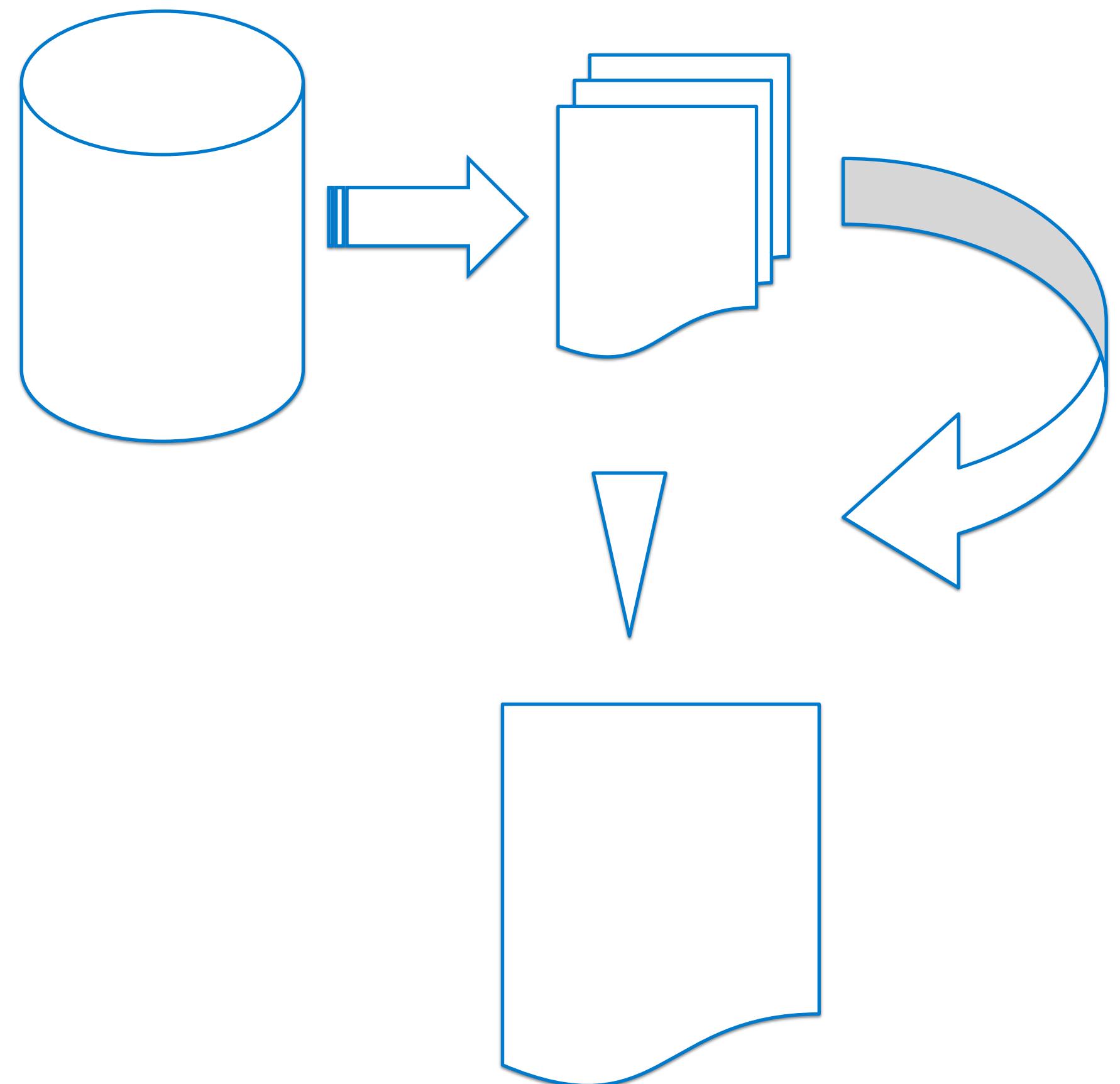
Speed & Resilience: iOS

- **Running tests on device is a nightmare!**
- **Simulator is preferred choice!**
- **Xcode did not support multiple simulators on single desktop**



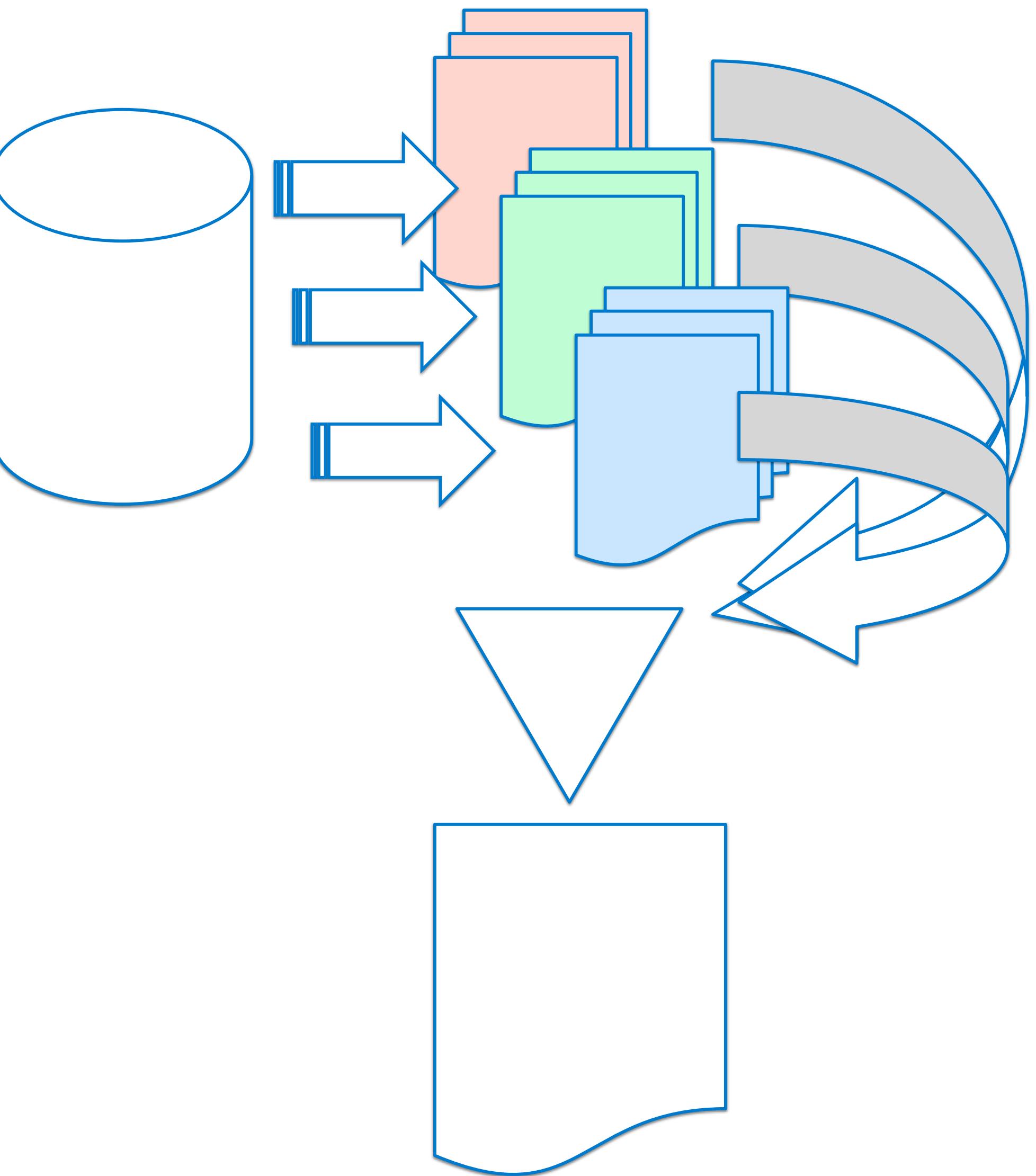
Speed & Resilience: iOS

- **Multiple desktops on a single macOS**
- **1 simulator per 1 desktop**
- **Tests executed via SSH**



Speed & Resilience: iOS

- **Multiple Hosts (ssh)**
- **Multi-parallel testing**



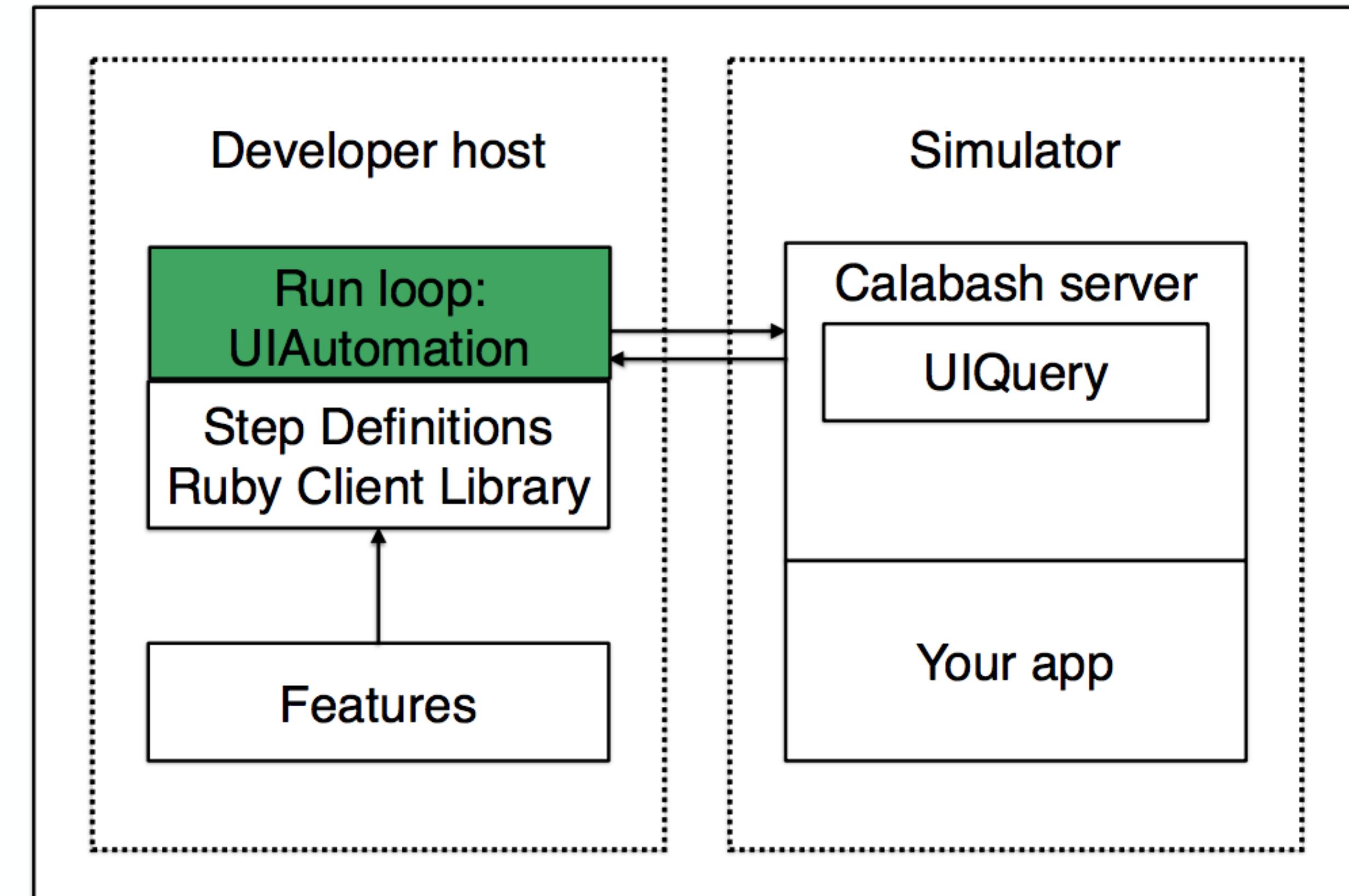
Speed & Resilience: iOS

- Multi-desktops start up is flaky
- Simulators often getting stuck



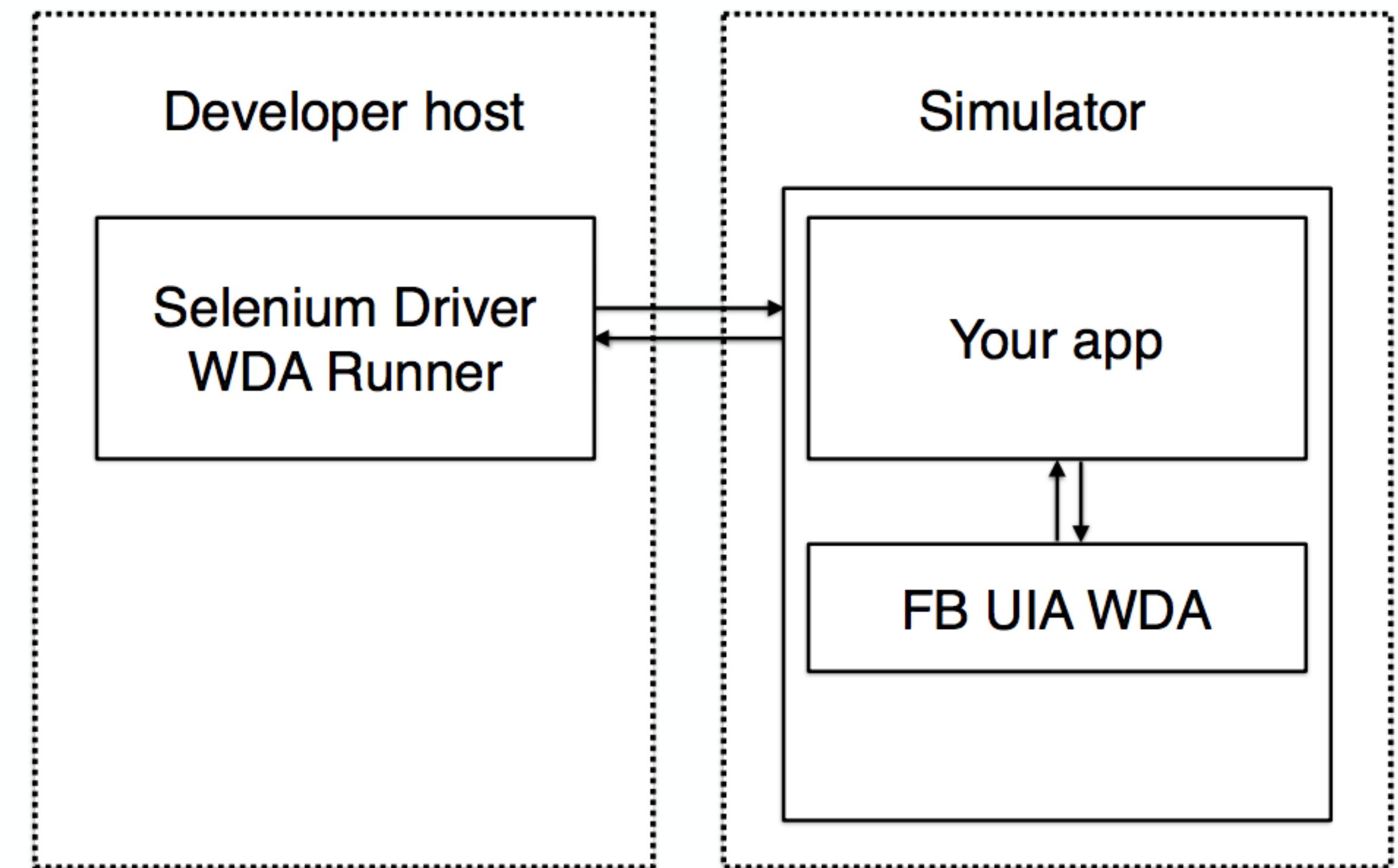
Speed & Resilience: iOS

- **Xcode finally has support for multiple simulators**
- **But Calabash has limitations**
 - **it is runloop which does touch operation**



Speed & Resilience: iOS

- **Web Driver from Facebook**
- **Supports multiple simulator on one desktop**

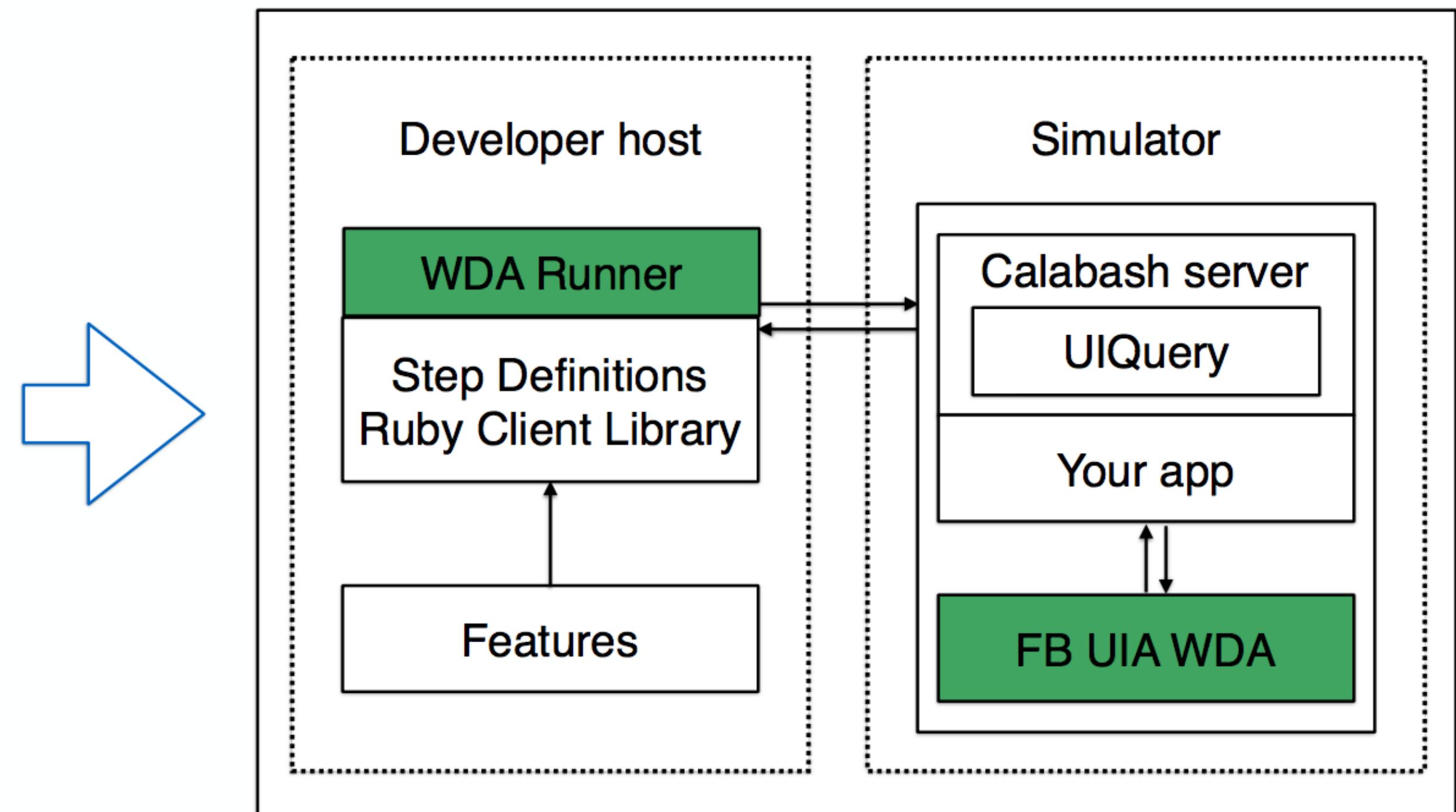
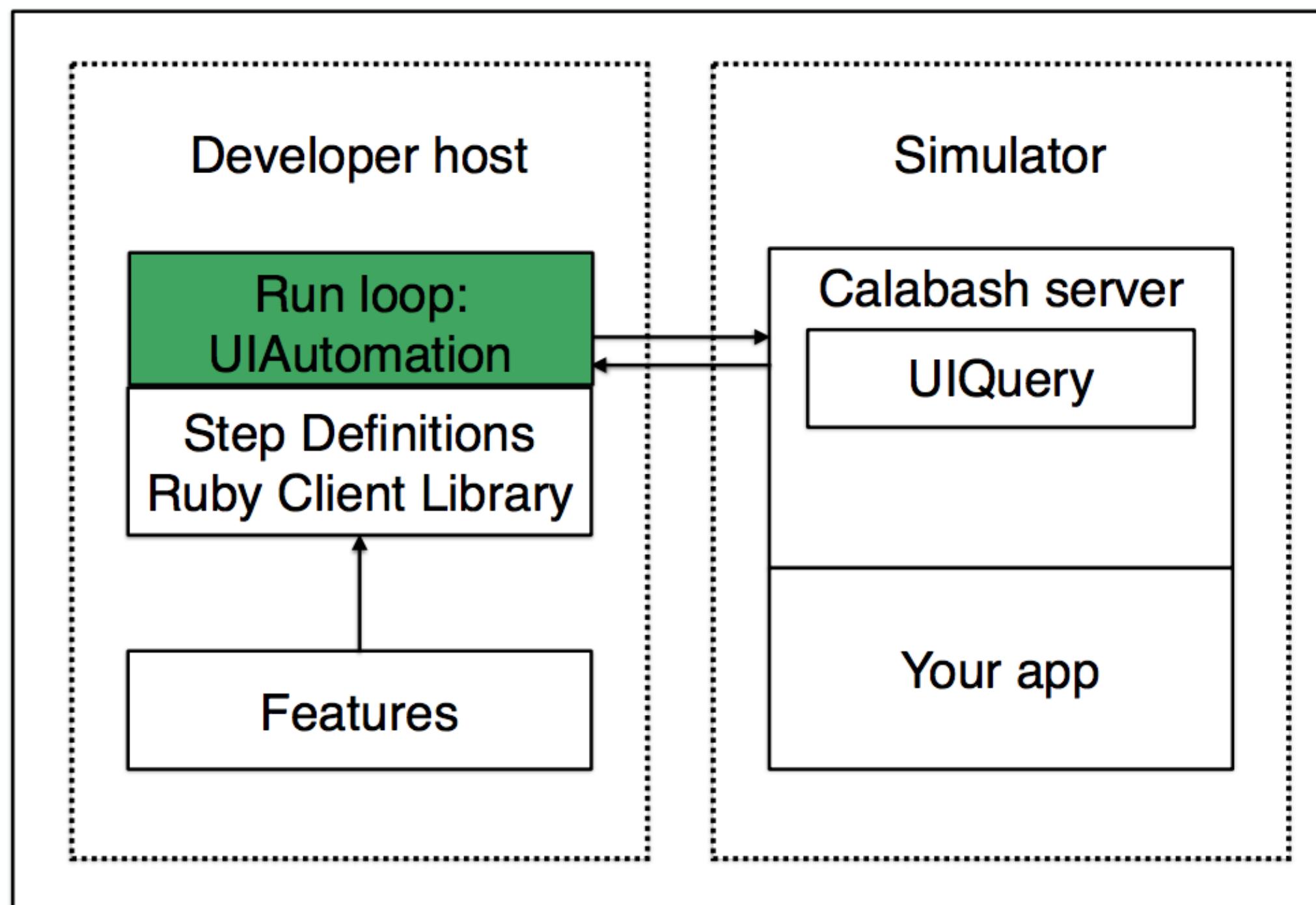


Speed & Resilience: iOS

- Calabash queries different from FB Web driver
- Full migration cost huge
- FB Web driver does not support backdoors
- Calabash faster in 99% cases

Speed & Resilience: iOS

Calabash + Facebook => Facebash!



Speed & Resilience: iOS

- **Fewer desktops = more simulators**
- **Relatively stable**
- **Far from target speed**



Speed & Resilience: iOS

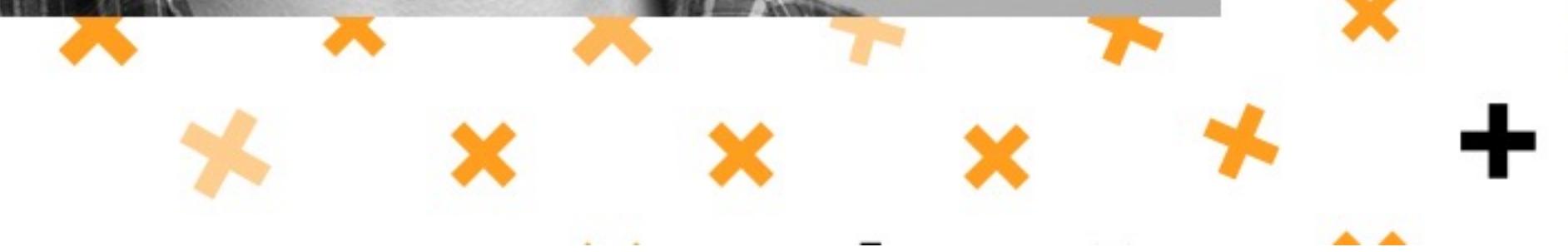
HEISENBUG

// 2018 Piter

Николай Абалов

Badoo

Когда нужны скорость
и масштабирование:
сервер распределенных
iOS-устройств



Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & Manual testers Involvement
- 100% test coverage
- Summary



Test Reliability: Problem

- End to End tests are flaky by nature
- More tests meant more Flaky tests
- Tests are often broken by app feature changes



Test Reliability: Problem

- **Problem with culture**
- **More focus on new tests**
- **Less focus on tests stability**



Test Reliability

95% tests should pass

Test Reliability

- Fixing failed tests became the top-priority
- Report failed tests every day
- No feature branch merge when tests failed.

Mobile Test Automation

NITROID

JINBA Automation.
Common Places
Visitors
HON white Label

DS

Running Tests on devices
iOS 9
Photo of the day.
HON white Label

Mobile Web

Own profile
Payments
Spotlight
HON white Label

① iOS 8
② iOS 7
③ iPad
④ Phone-sim
⑤ native
⑥ Appium

Engineering

Running Tests powered on iOS
"Deep Linking" in iOS
Adding more devices to "Deep Linking"

Date	Android	iOS	MW
11/09	101/477	8/429	64/366
14/09	93/477	1/429	24/366
15/09	X	9/432	15/365
16/09	163/477	2/432	8/372
17/09	19/477	0/432	9/372
18/09	18/477	18/432	29/372
21/09	15/477	0/432	36/384
22/09	12/480	7/432	24/385
23/09	15/480	15/432	9/385
24/09	08/480	7/432	1/387
25/09	16/480	1/432	23/387
28/09	54/482	7/432	15/400
29/09	18/482	X	63/400
30/09	15/490	15/432	28/406
1/10	13/490	2/432	X
2/10	6/490	2/432	30/418

October 2015

Test Reliability

95% tests should pass

99% tests must pass

Test Reliability

- Improved test infrastructure stability & monitoring**
- Improved App server infrastructure for tests**
- Better APIs for test data**

Date	Android	iOS	MOBILE WEB	WINDOWS PHONE	BUMBLE ANDROID	BUMBLE iOS
1/11/2016		88/829	9/768	3/108	1/56	10/56
2/11/2016	3/805	86/829	9/768	0/108	2/56	7/56
3/11/2016	4/806	89/827	9/768	0/108	1/56	2/56
4/11/2016	45/806	87/827	0/768	6/108	x	2/56
7/11/2016	5/806	82/826	0/768	3/118	1/56	3/56
28/11/2016	36/806	88/826	3/768	8/118	1/56	3/56
09/11/2016	x	86/828	0/768	5/118	5/66	4/60
10/11/2016	9/806	87/829	1/768	10/118	1/66	4/60
11/11/2016	4/806	x	1/771	8/118	5/66	6/60
14/11/16	8/809	86/829	3/771	6/118	2/66	4/60
15/11/16	4/809	88/829	8/771	x	3/66	2/59
16/11/16	10/809	x	3/771	5/118	x	6/62
17/11/16	9/809	88/829	2/771	8/126	3/66	2/62
18/11/16	2/811	82/829	0/770	12/126	1/66	2/62
21/11/16	5/811	80/838	-	7/126	-	-
22/11/16	3/816	89/838	x	7/126	1/66	x
23/11/16	x	89/838	x	7/126	10/67	x
24/11/16	5/824	21/838	6/789	8/136	2/67	3/62
				10/136	1/67	2/62

Test Reliability

Branch	#	Results	Artifacts	Changes	Started	Duration
<default>	#7020	⚠ Tests failed: 3 (3 new), passed: 1090, muted: 24 ▽	View ▽	Changes (139) ▽	10 May 18 12:15	24m:44s
<default>	#6993	⚠ Tests failed: 7 (6 new), passed: 1089, muted: 25 ▽	View ▽	Changes (25) ▽	09 May 18 13:34	23m:08s
<default>	#6992	⚠ Tests failed: 3 (3 new), passed: 1089, muted: 25 ▽	View ▽	Changes (91) ▽	09 May 18 12:14	24m:14s
<default>	#6987	⚠ Tests failed: 1, passed: 1093, muted: 23 ▽	View ▽	No changes ▽	09 May 18 08:00	23m:09s
<default>	#6986	⚠ Tests failed: 1 (1 new), passed: 1091, muted: 25 ▽	View ▽	Changes (128) ▽	09 May 18 06:00	23m:20s
<default>	#6966	⚠ Tests failed: 1 (1 new), passed: 1094, muted: 22 ▽	View ▽	Changes (20) ▽	08 May 18 12:12	25m:07s
<default>	#6959	⚠ Tests failed: 3 (2 new), passed: 1089, muted: 25 ▽	View ▽	No changes ▽	08 May 18 08:00	24m:48s
<default>	#6958	⚠ Tests failed: 4 (2 new), passed: 1088, muted: 25 ▽	View ▽	Changes (31) ▽	08 May 18 06:00	23m:36s
<default>	#6953	⚠ Tests failed: 5 (2 new), passed: 1088, muted: 24 ▽	View ▽	aida <aida@cor... (7) ▽	07 May 18 12:09	27m:33s
<default>	#6950	⚠ Tests failed: 3 (2 new), passed: 1088, muted: 26 ▽	View ▽	No changes ▽	07 May 18 08:00	24m:19s
<default>	#6949	⚠ Tests failed: 1, passed: 1091, muted: 25 ▽	View ▽	Changes (4) ▽	07 May 18 06:00	23m:36s
<default>	#6948	⚠ Tests failed: 6 (4 new), passed: 1084, muted: 25 ▽	None ▽	No changes ▽	06 May 18 12:00	22m:55s
<default>	#6947	⚠ Tests failed: 7 (5 new), passed: 1085, muted: 23 ▽	None ▽	Changes (18) ▽	06 May 18 08:00	23m:48s
<default>	#6942	⚠ Tests failed: 2, passed: 1088, muted: 25 ▽	None ▽	No changes ▽	05 May 18 12:00	21m:56s
<default>	#6940	⚠ Tests failed: 4 (3 new), passed: 1087, muted: 24 ▽	None ▽	Changes (260) ▽	05 May 18 08:00	21m:46s
<default>	#6908	⚠ Tests failed: 1, passed: 1093, muted: 25 ▽	None ▽	Changes (121) ▽	04 May 18 08:00	22m:03s
<default>	#6879	⚠ Tests failed: 1 (1 new), passed: 1094, muted: 24 ▽	None ▽	Changes (161) ▽	03 May 18 08:00	23m:26s

Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & Manual testers Involvement
- 100% test coverage
- Summary



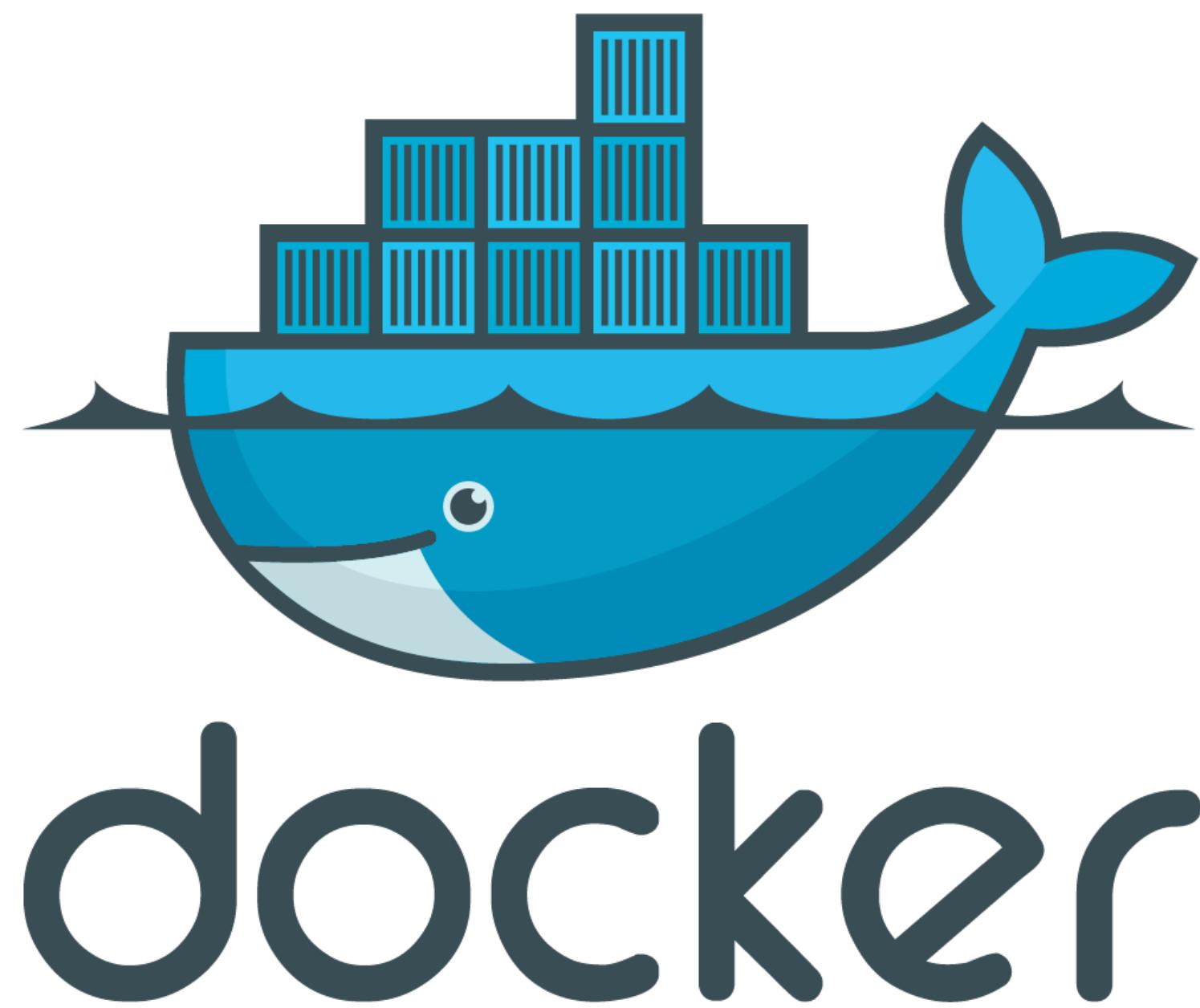
Developers: Lack of idea on tests

- Developers see it as a black box
- Different tools
- Different programming language
- Long list of instructions to run locally



Developers: Lack of idea on tests

- Dockerise the test-run environment



Developers: Lack of idea on tests

- Dockerise the test-run environment
- Failed test report provides the docker command

```
===== SCENARIO_WITH_LINE: =====
features/functional/ownprofile/ownprofile_profile_photo.feature:59
=====

===== LOCAL RUN WITH Docker: =====
./docker_droid.sh path/to/BadooApp.apk features/functional/ownprofile/ownprofile_profile_photo.feature:59
=====
Add 'OBFUSCATED=1 MAPPING_PATH=~/Desktop/BadooApp-internal-badoo-QA.mapping.txt' for obfuscated apk
Add 'NO_APP_INSTALL=true' to save time by avoiding app install everytime (will work only if tests are run once without using it)
=====

===== DEVICE DETAILS: =====
Android SDK built for x86_64 => emulator-5612 tcp:automation1.d4:5037
=====
```

Mobile Test Automation Team Home

Created by Alexander Izmaylov, last modified by Nikolai Abalov on Mar 06, 2018

Pages

- Set up Test Developer environment
 - Platform Android
 - Platform iOS
 - iOS: Basic Set Up
 - iOS: Virtual Machine Set Up
 - iOS Device server Incidents
 - Platform Mobile Web
 - Mobile Web: Test Automation Setup
- Set up real devices for automation
 - iOS devices
- Test Write Practice
 - Guidelines: how to write?
 - Code Health
 - Cucumber Feature
 - Coding guidelines for mobile test automation
 - Naming Guidelines
 - Test Code on Ruby
 - Calabash Code Standards / Tips
- Run tests
 - Android
 - Run a test on Android: via Docker
 - iOS
 - Local Run on iOS: Run tests on test developer host
 - Run iOS tests on physical devices
 - Teamcity: Choose components to run functional tests

Developers: Lack of idea on tests

- Dockerise the test-run environment
- Failed test report provides the docker command
- Documentation

Developers: Lack of idea on tests

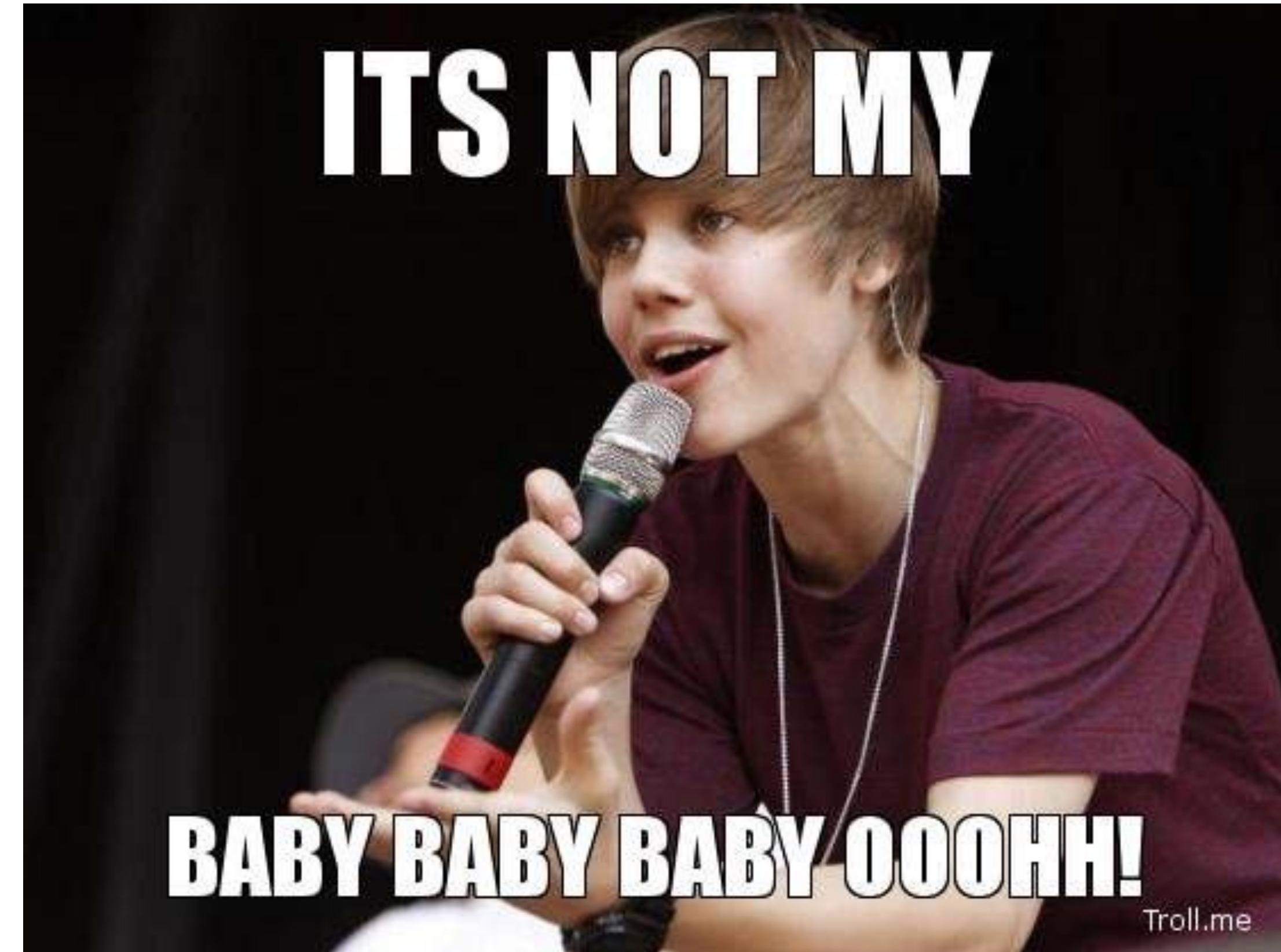
- Checking automation test failures
- Some even fixing the test code!
- Developer deploying tasks without QA help



Manual testers

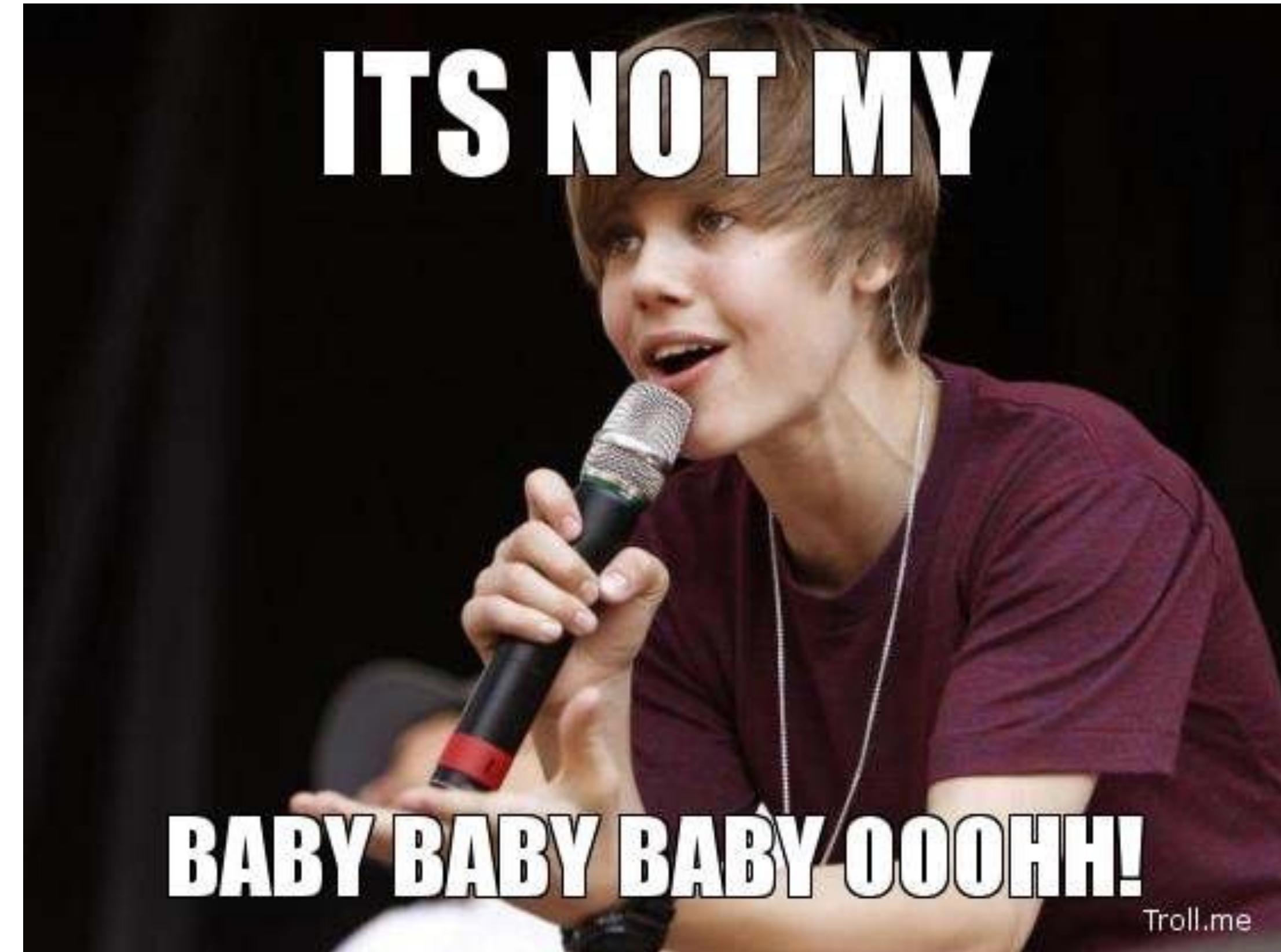
Manual testers: Lack of involvement

- **Tests automated and maintained by separate team**
- **QA lost the responsibility for tests**
- **QA never had confidence in automation**



Manual testers: Lack of involvement

- QA testing what we are automated
- Automated tests not getting the advantage of QA experience



TIME FOR REVIEW

Manual testers: Lack of involvement

- QA Reviewed the tests
- Added missing flows
- Improved the test verifications



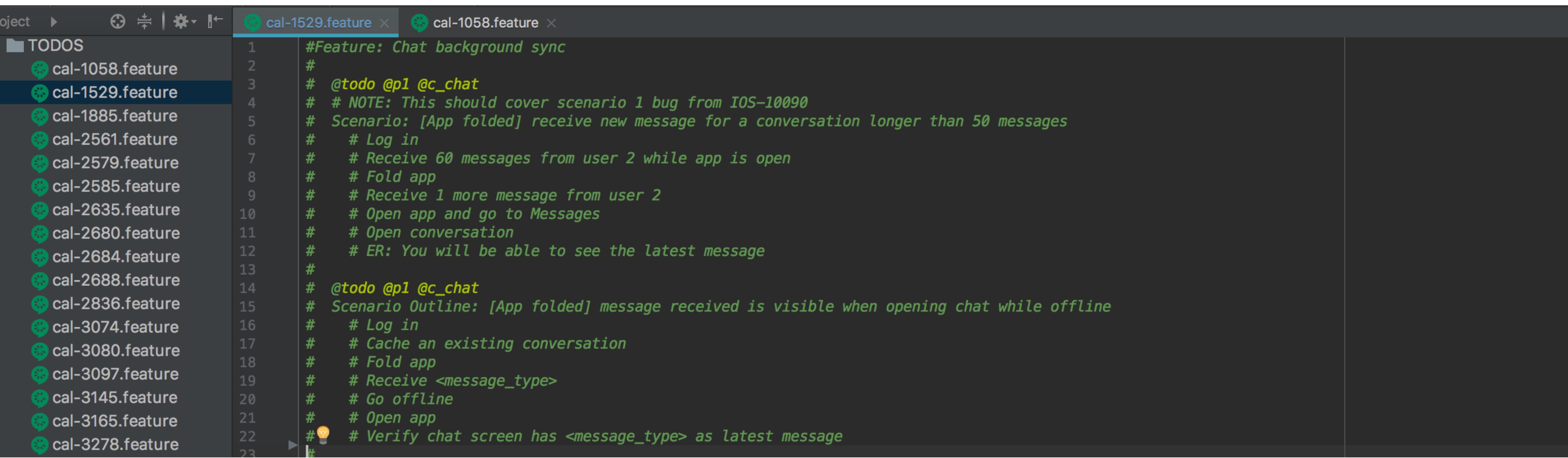
YEAH OK,

TRY TO FIND COVERAGE



Manual testers: Functional Coverage

- Adding missing tests as TODO/@todo in feature files
- Adding TODO/@todo tests for new features by QA



```
Object  Project  +  ⌂  ⚙  ⌛  cal-1529.feature ×  cal-1058.feature ×
[TODOS]
cal-1058.feature
cal-1529.feature
cal-1885.feature
cal-2561.feature
cal-2579.feature
cal-2585.feature
cal-2635.feature
cal-2680.feature
cal-2684.feature
cal-2688.feature
cal-2836.feature
cal-3074.feature
cal-3080.feature
cal-3097.feature
cal-3145.feature
cal-3165.feature
cal-3278.feature

1 #Feature: Chat background sync
2 #
3 # @todo @p1 @c_chat
4 # # NOTE: This should cover scenario 1 bug from IOS-10090
5 # Scenario: [App folded] receive new message for a conversation longer than 50 messages
6 #   # Log in
7 #   # Receive 60 messages from user 2 while app is open
8 #   # Fold app
9 #   # Receive 1 more message from user 2
10 #   # Open app and go to Messages
11 #   # Open conversation
12 #   # ER: You will be able to see the latest message
13 #
14 # @todo @p1 @c_chat
15 # Scenario Outline: [App folded] message received is visible when opening chat while offline
16 #   # Log in
17 #   # Cache an existing conversation
18 #   # Fold app
19 #   # Receive <message_type>
20 #   # Go offline
21 #   # Open app
22 #   # Verify chat screen has <message_type> as latest message
23 #
```

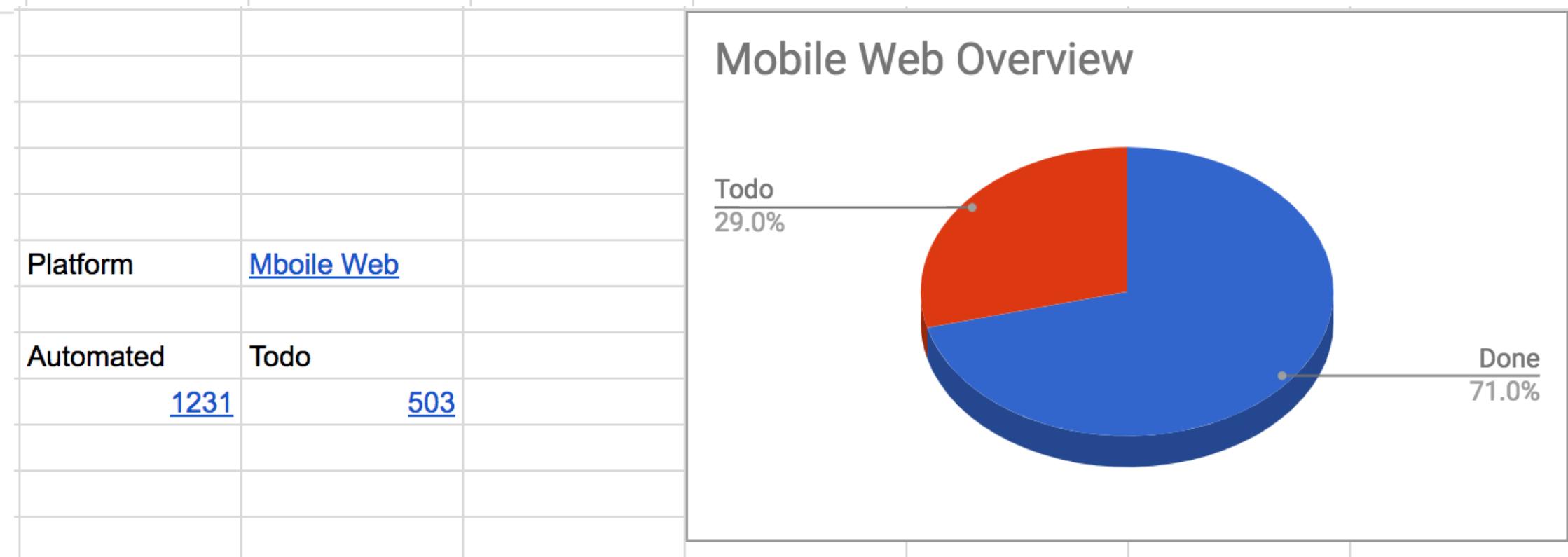
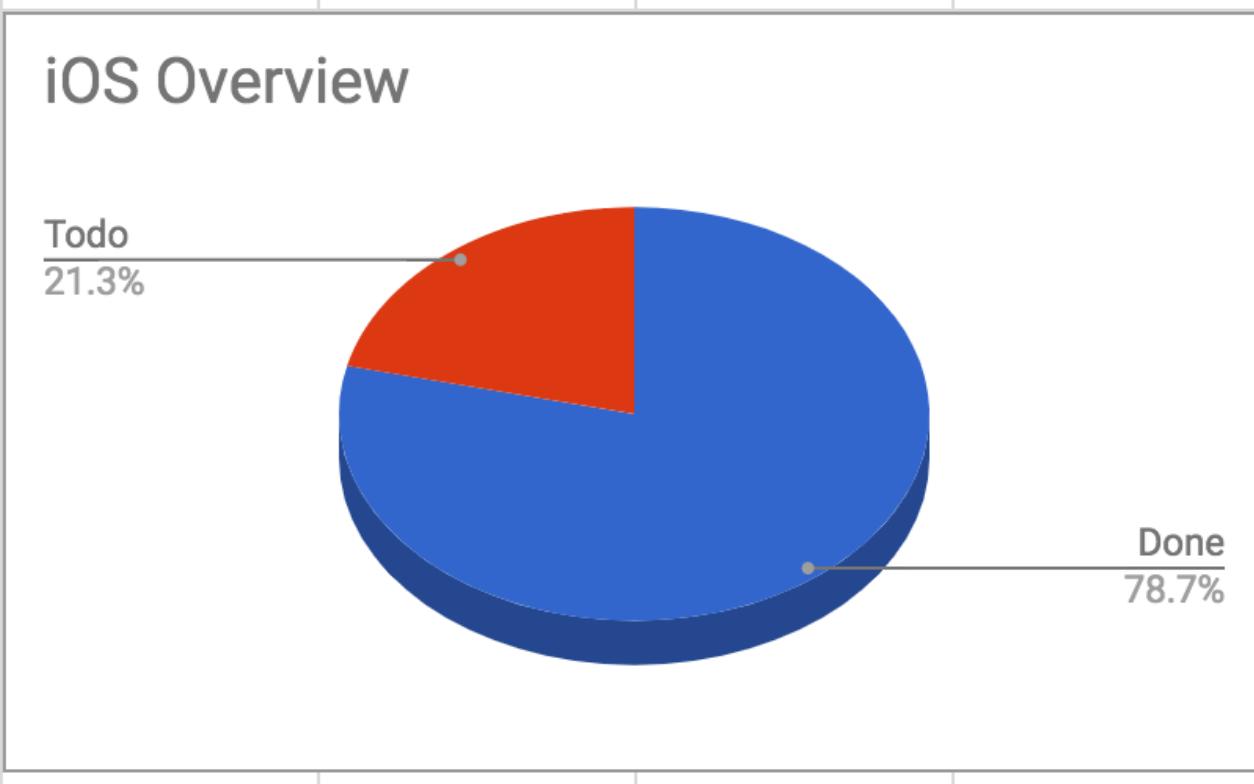
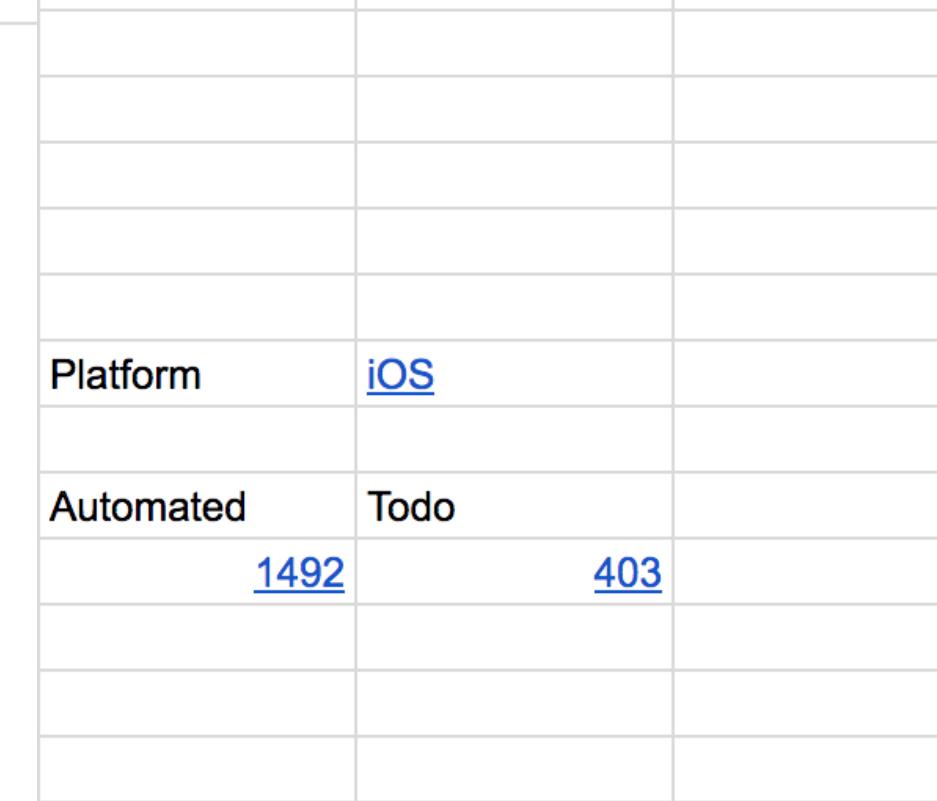
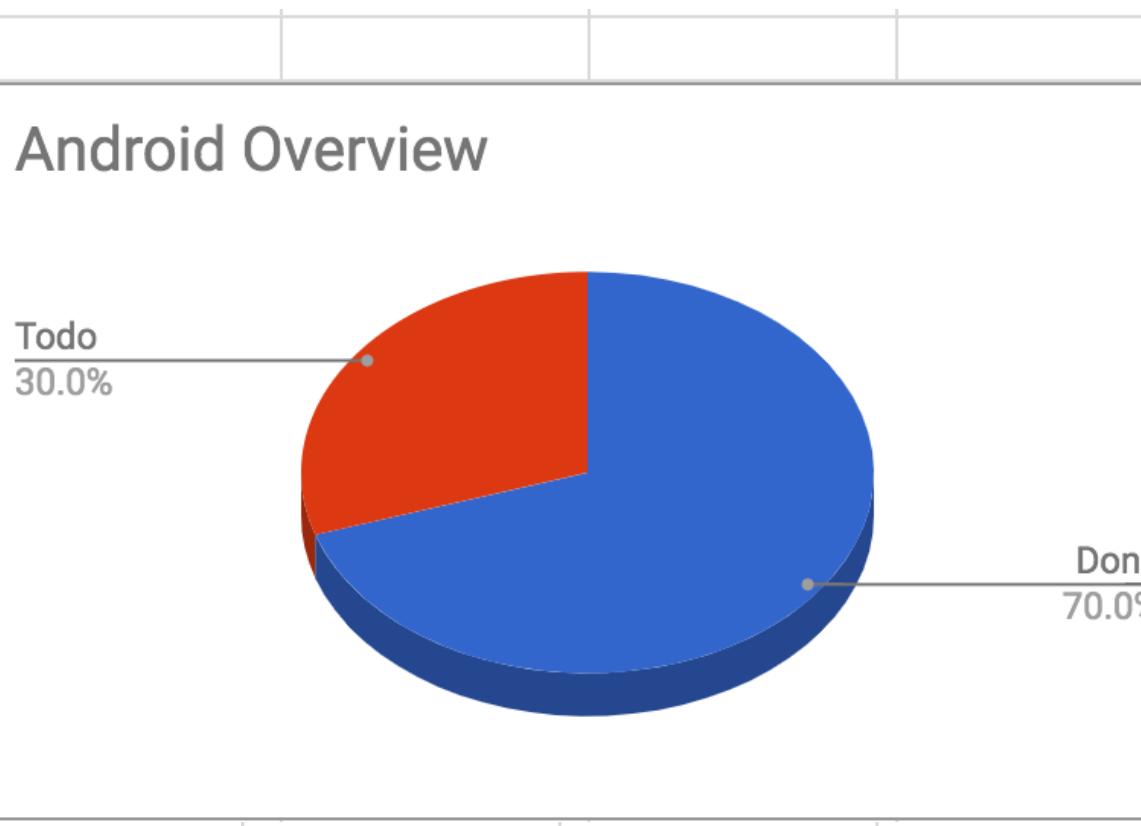
Manual testers: Functional Coverage

**Functional coverage = Test Automated /
(Tests automated + TODO tests)**

The screenshot shows a code editor interface with a dark theme. On the left, there's a sidebar titled 'TODOS' containing a list of feature files: cal-1058.feature, cal-1529.feature, cal-1885.feature, cal-2561.feature, cal-2579.feature, cal-2585.feature, cal-2635.feature, cal-2680.feature, cal-2684.feature, cal-2688.feature, cal-2836.feature, cal-3074.feature, cal-3080.feature, cal-3097.feature, cal-3145.feature, cal-3165.feature, and cal-3278.feature. The file 'cal-1529.feature' is currently selected and its content is displayed in the main pane. The content of 'cal-1529.feature' is as follows:

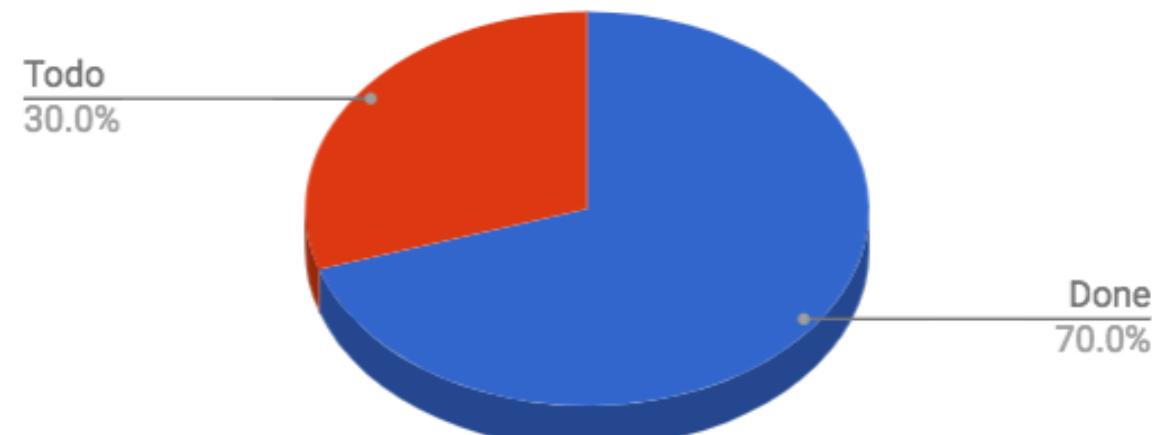
```
1 #Feature: Chat background sync
2 #
3 # @todo @p1 @c_chat
4 # # NOTE: This should cover scenario 1 bug from IOS-10090
5 # Scenario: [App folded] receive new message for a conversation longer than 50 messages
6 #   # Log in
7 #   # Receive 60 messages from user 2 while app is open
8 #   # Fold app
9 #   # Receive 1 more message from user 2
10 #   # Open app and go to Messages
11 #   # Open conversation
12 #   # ER: You will be able to see the latest message
13 #
14 # @todo @p1 @c_chat
15 # Scenario Outline: [App folded] message received is visible when opening chat while offline
16 #   # Log in
17 #   # Cache an existing conversation
18 #   # Fold app
19 #   # Receive <message_type>
20 #   # Go offline
21 #   # Open app
22 #   # Verify chat screen has <message_type> as latest message
23 #
```

Manual testers: Functional Coverage



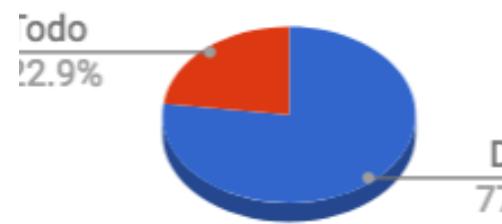
Manual testers: Functional Coverage

Android Overview



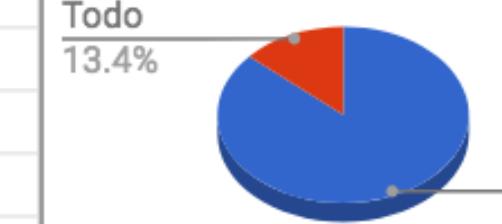
Game	350	104	Game: Encounters & Others Profile
Login	174	27	Login, Registration, Onboarding and Verification
Messenger	715	132	Messenger: Chat, Connections, Folders, Spotlight
Payments	332	256	Payments, Purchases & Billing
People	104	94	People Nearby, Lookalikes, Friends of friends
Profile	368	135	Own profile, Settings, Popularity, Blocked Users, Help & Feedback
Server	272	94	Startup, Pushes, Landing, Deeplinks
Social	90	43	Social, Sharing, Friends & Invites
Uncategorised	2	1	
P2P3	17	127	
Overview	1346	576	All categorised tests

Android Game



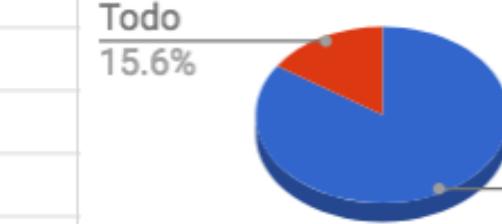
Game: Game: Encounters & Others Profile

Android Login



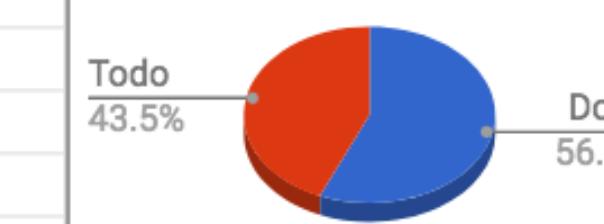
Login: Login, Registration, Onboarding and Verification

Android Messenger



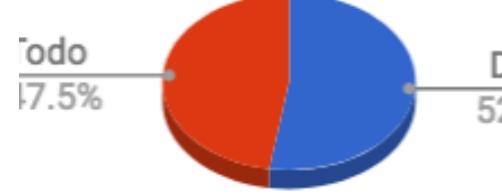
Messenger: Messenger: Chat, Connections, Folders, Spotlight

Android Payments



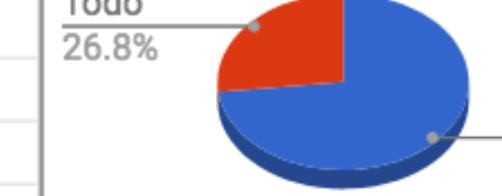
Payments: Payments, Purchases & Billing

Android People



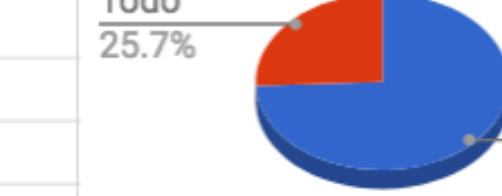
People: People Nearby, Lookalikes, Friends of friends

Android Profile



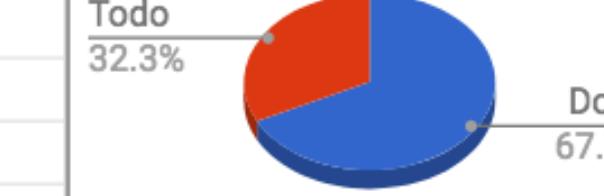
Profile: Own profile, Settings, Popularity, Blocked Users, Help & Feedback

Android Server



Server: Startup, Pushes, Landing, Deeplinks

Android Social



Social: Social, Sharing, Friends & Invites

Manual testers: Functional Coverage

- It brings transparency
- It brings confidence
- Helps to drive what to automate
- It is our baby now!

©Victoria Ford's Photography

Manual testers: Writing tests!

- **Manual testers started analysing failures**
- **Manual testers fixing the tests!**
- **Manual testers automating new features!**



mak

Manual testers: Writing tests!

- **Developing new tools on top of test automation**
- **Frameworks improvements**
- **Still automate complex features**



mak

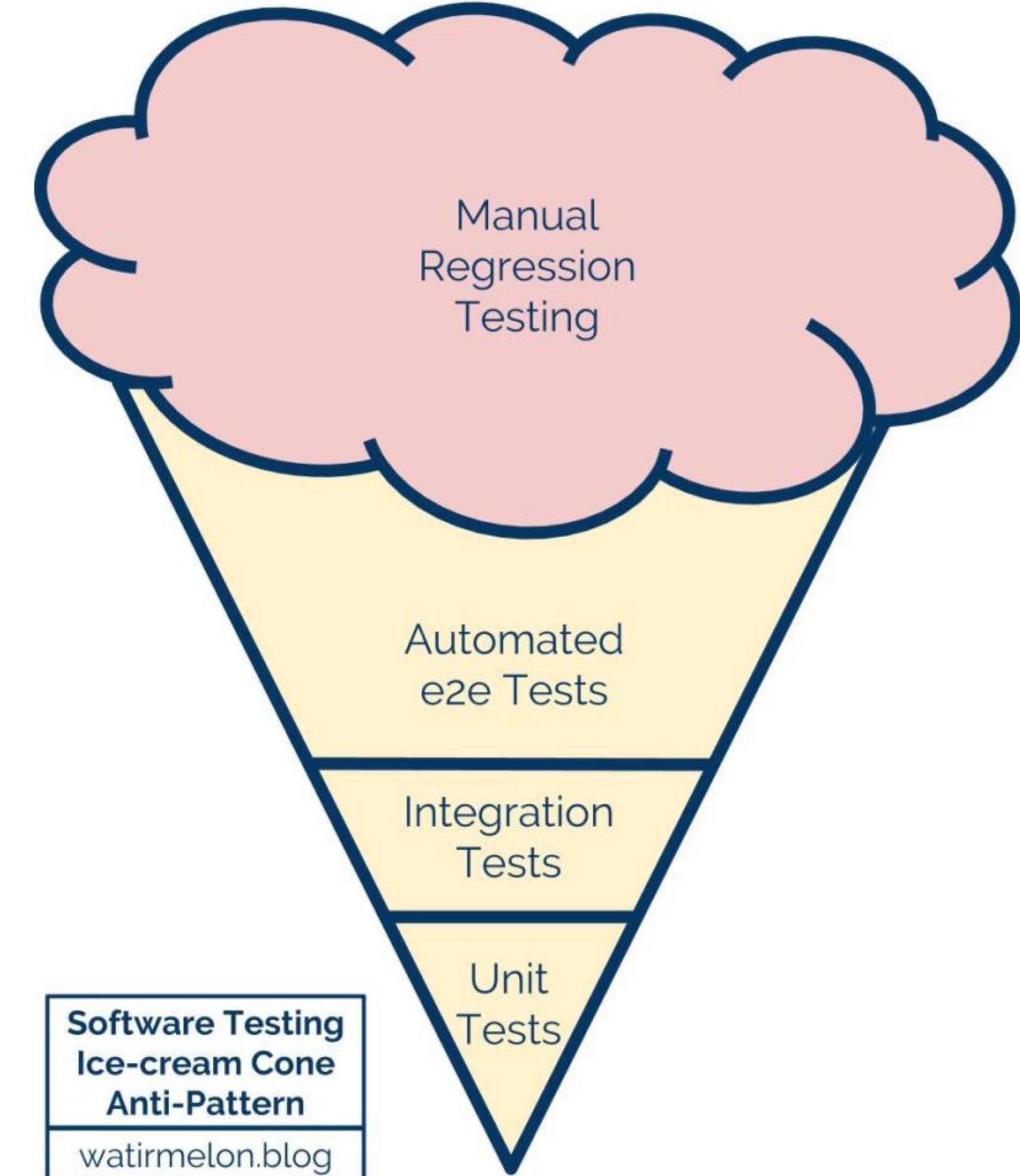
Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & QA Involvement
- 100% test coverage
- Summary



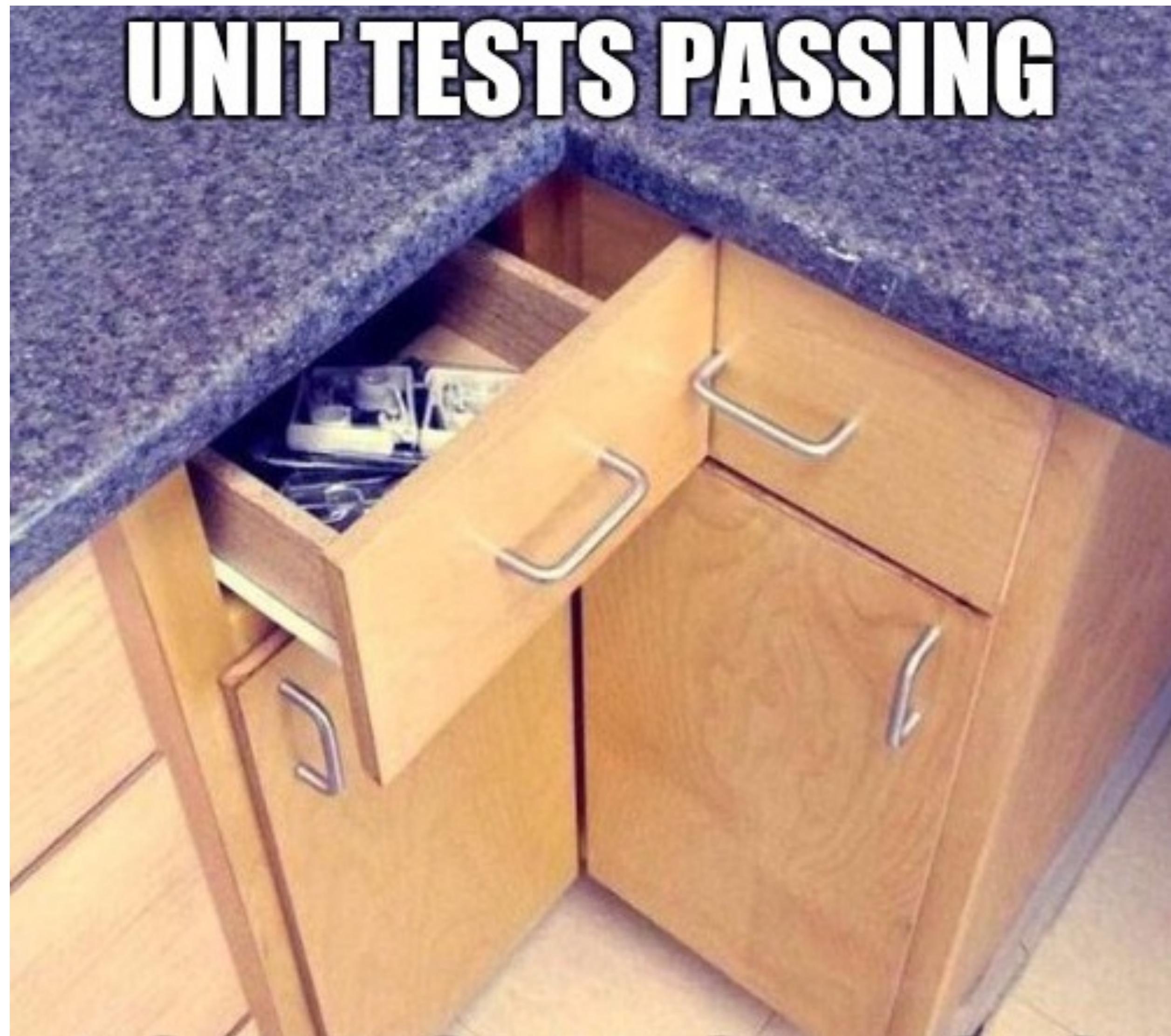
100% coverage: Is that a problem?

- Not a test-pyramid, it is an ice-cream
- Ice-cream can melt!



100% coverage: Is that a problem?

- Unit tests exist
- Lack of integration layer



100% coverage: Is that a problem?

- Develop tools for integration layer
- Integration tests by Developer
- Gradually move End to End tests to integration layer



Agenda

- A bit about where we are
- 2012: First step
- Cross-platform test automation
- Speed & Resilience
- Tests Reliability
- Developers & QA Involvement
- 100% test coverage
- Summary



Summary

- Choose your test strategy
 - Cross platform Vs platform specific
- Align with development process
- Don't ignore flaky tests
- Faster feedback is always important

Summary

- Developers & QA involvement is crucial
- Spend time on your infrastructure
- Encourage unit and integration tests

Summary

“

If you are test automation
engineers you will exist forever,
but you can do much more than
automating tests

”

– Not Alan Page



Cheers!

techblog.badoo.com