

MOET

<https://github.com/eing/moet>

Mobile End-to-End Testing
Mobile Exploratory Testing

Eing Ong
Intuit, Inc.

@eingong | eing.ong@intuit.com

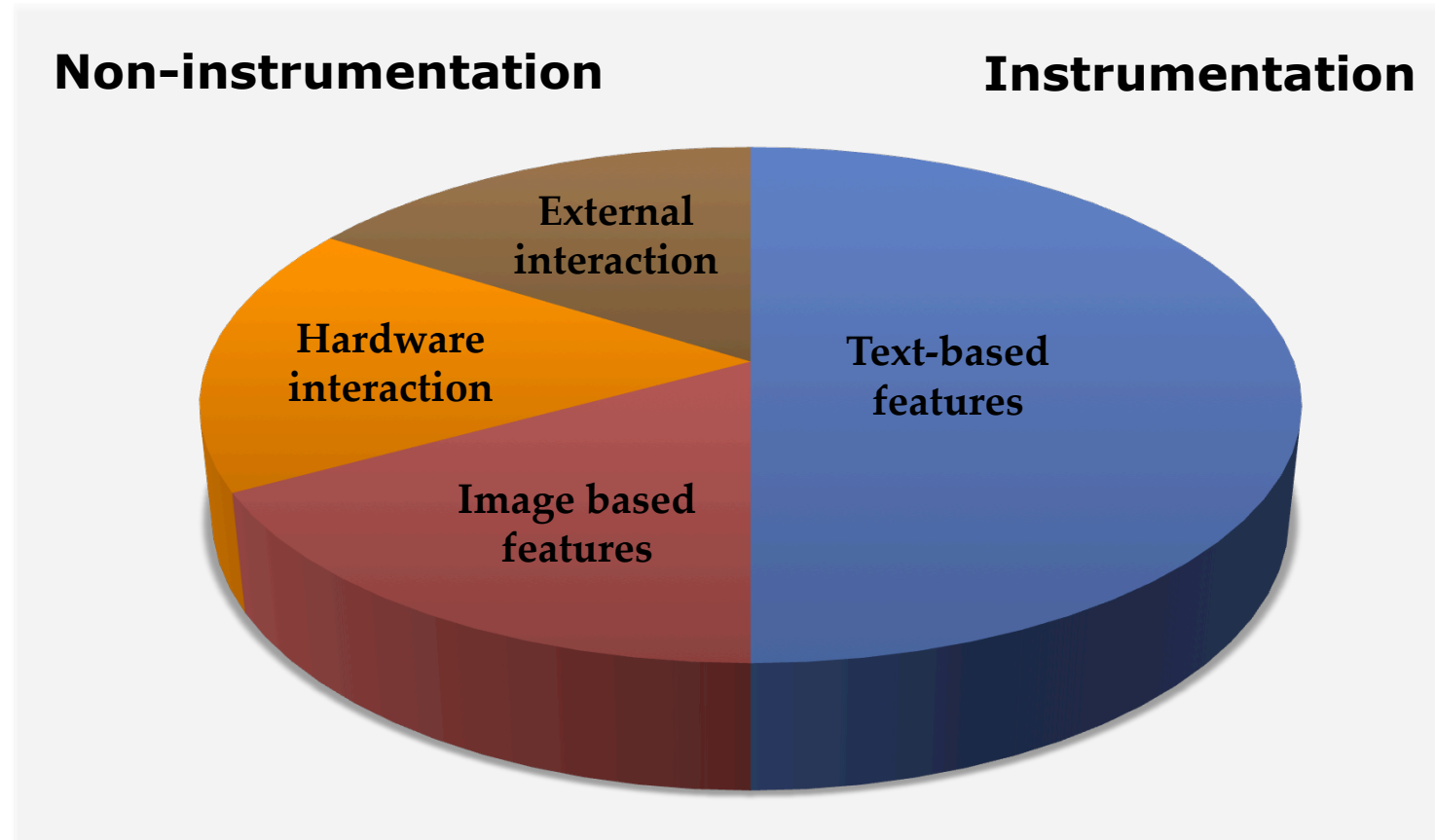
Mobile automation technologies

- **Two categories**
 - Instrumented technique
 - Non-instrumented technique
- **What is instrumentation?**
 - Tests are compiled with the app
 - Tests are installed & launched with the app
 - Source code is required and may need to be modified
 - Only one app can be executed at a time
 - White box approach

Advantages of both techniques

Non-instrumentation	Instrumentation
<ul style="list-style-type: none">• Device platform agnostic• Test code reuse• Test language & test harness autonomy• Support for<ul style="list-style-type: none">Multi-applications testingCustom composite UI elementsDatabase/server API assertionsUse of external libraries (e.g. hardware/image libraries)	<ul style="list-style-type: none">• Elements can be accessed• Debugging ease• Test verification ease• Reduce tools dependencies• Support for<ul style="list-style-type: none">Installing applicationLaunching applicationCleanup (kill application)Test execution on deviceCode coverage

Which technique should I use ?



Mobile automation tools

Mobile OS	Non-instrumentation	Instrumentation
Android	eggPlant, Sikuli, MOET, MonkeyRunner	Robotium, Calabash, MonkeyTalk
iOS	eggPlant, Sikuli, MOET	UIAutomation, KIF, iCuke, Frank, UISpec, Zucchini, Bwoken, Calabash, MonkeyTalk, Applecart, iOS-driver

MOET

- *Think design*
 - Interfaces
 - Creational pattern
- *Think reuse*
 - Device independent tests
- *Think One*
 - IDE
 - Test harness and language

Test

Login("user1","passwd1")

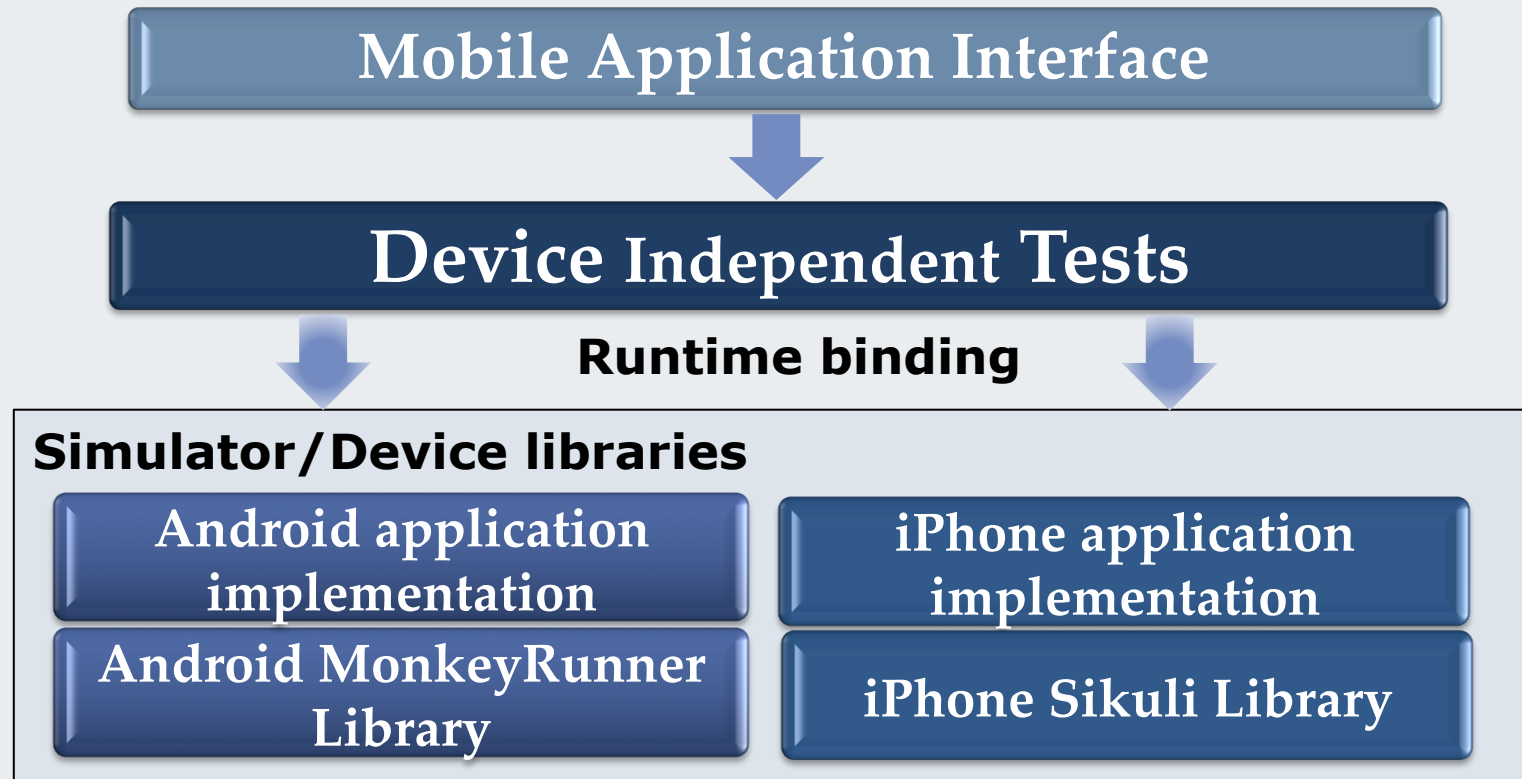
iPhone implementation

touch(100,100)
enter(username)
touch(100,200)
enter(password)
touch(150, 300)

iPhone Sikuli library

void enter()
void touch(x,y)

Anatomy of MOET



Top Ten Wish List
a.k.a.
Mobile test automation
Wall of Pain

Android Wall of Pain

10. There's more to test execution

- Command with report file generation :
adb shell am instrument -w -e reportFile junit-report.xml
com.intuit.test/com.zutubi.android.junitreport.JUnitReportTestRunner
- Date-time fix if using device snapshots :
adb shell date -s `date +"%Y%m%d.%H%M%S"`

Android Wall of Pain

10. There's more to test execution

- Command with report file generation :
`adb shell am instrument -w -e reportFile junit-report.xml
com.intuit.test/com.zutubi.android.junitreport.JUnitReportTestRunner`
- Date-time fix if using device snapshots :
`adb shell date -s `date +"%Y%m%d.%H%M%S"``

9. Where are the test results ?

- Pulling results from device :
`adb pull /data/data/com.intuit.test/files/junit-report.xml junit-report.xml`

iOS Wall of Pain

8. *iOS simulator...the singleton pattern*

- *Multiple instances of simulator*
- *Headless execution*

iOS Wall of Pain

8. *iOS simulator...the singleton pattern*

- *Multiple instances of simulator*
- *Headless execution*

7. What test harness ?

- *Test grouping, test assertions, base test class*
- *Results needs to be displayable in browser*

iOS Wall of Pain

8. *iOS simulator...the singleton pattern*

- *Headless execution*
- *Multiple instances of simulator*

7. *What test harness ?*

- *Test grouping, test assertions, base test class*
- *Results needs to be displayable in browser*

6. **How about a CI plugin**

- Test execution
- Test results
- Code coverage

iOS Wall of Pain

5. *Where's the IDE plugin ?*

- Code formatting
- API completion
- Breakpoints, debugger
- “javadoc” help
- One click test execution of test method/testsuite

All mobile platforms

4. Problem ? Search ?

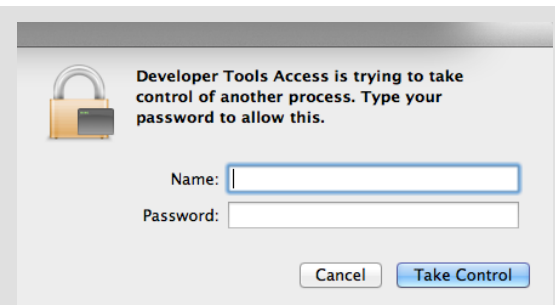
- Blogs, stackoverflow, developer groups for devices, etc.
- Mobile testing one-stop tips & tricks (not questions)
- Community driven effort

Tip – Remove authorization prompt

Administrative rights

Update /etc/authorization

```
<key>system.privilege.taskport</key>
<dict>
  <key>allow-root</key>
  <!-- previous value <false/> -->
  <true/>
</dict>
```



Tip – Clean cache, preferences, SQLite

~/Library/Application Support/iPhone Simulator/{SDK}/Applications/<uuid>

Library: Preferences, Caches

Documents: sqlite3 <app>.sqlitedb

All mobile platforms

3. *How about a mobile test report ?*

- Mobile test report standard e.g. *mobile-test-report.xml*
 - Screenshots and video recording of failed tests
 - Device logs
 - Device detection

All mobile platforms

3. *How about a mobile test report?*

- Mobile test report standard e.g. *mobile-test-report.xml*
 - Screenshots and video recording of failed tests
 - Device logs
 - Device detection

2. **Mobile testing is anything but mobile**

- OTA app installation and execution
- Testing via wifi and carrier networks
- Test tool independent component
- Selenium grid-like test infrastructure

All mobile platforms

1. Yet another proprietary language?

- Open sourced test harness & language for
 - Instrumented tests
 - Non-instrumented tests
 - Hybrid apps tests

Next steps

