Differences in automated testing on MeeGo and Android mobile platforms

Oleksandr Kachur Vladimir Sayenko



MeeGo tools

- Testrunner-lite tool that executes tests on real device.
- Testrunner GUI front end for testrunner-lite.
- Tdriver automation test API for QT applications.
- Testplanner tool for writing and executing test plans

Testrunner

> 😡	Testrunner	- test.xml 💿 🔿 🛞	
File Test Definition Test Run V	/iew Help		File Test (
🗎 💾 🕨 🔻 🏟			
Suite / Set / Case / Step	Result Comment	Test Definition - /home/asher/test.xml • ×	Suite / Set
- my-multimedia-tests	Passed (Total: 3, Pass	Total 3 Passed 3	- my-mu
- meego-bash-tests	Passed (Total: 3, Pass	Failed 0	- mee
+ echo on all terminals	Passed	N/A 0	+6
+ where am I?	Passed	Test Case: Execute calculator test	+-1
+ Execute calculator test	Passed		+
		Suite: my-multimedia-tests Passed	
		domain Multimedia started: 13:01:31	
		Set: meego-bash-tests ended: 15:01:34 duration: 00:00:03	
		feature Video Playback environments hardware, scratchbox	
		Case: Execute calculator test	
		requirement - type Functional	
		subfeature - level Component	
		timeout 90 automatic	
		Steps:	
		1 ruby /etc/tdriver/visualizer/calculator.rb Passed	
		returned 0 (ctdout) started: 13:01:31	
		expected 0 stderr duration 00.00.03	
			/lib/gems/1.8
			verify { calc_
			lc_app.close
			ed exception
		(tmp/tortrupper output 9623 log	n /var/lib/gen
		Debug Z Info Step output	n /etc/tdriver
		(TNEO) 13:01:34 Distribut	
		[INFO] 15:01:34 Finisned!	
<) <>	V	<

File Test Defi

> 💿			Testrunner	- test.xml		00	8
File Test D	efinition Test Run V	ew Help					
	> 🍸 🚳						
Suite / Set	/ Case / Step	Result	Comment	Test Def	nition - /home/asher/test.xml	0	×
- my-mult	timedia-tests	Failed	(Total: 3, Pass	Total	3		
- mee	go-bash-tests	Failed	(Total: 3, Pass	Failed	2		
+ e	cho on all terminals	Passed		N/A	0		
- w	/here am I?	Passed					
+ E	xecute calculator test	Failed		Test Cas	e: Execute calculator test	0	×
				Suite: n	ny-multimedia-tests Fa	iled	
				d	omain Multimedia started:	13:07	:49
				Set: n	neego-bash-tests ended: feature Video Playback duration: nvironments hardware, scratchbox	13:08 00:00	:02 :13
				Case:	vecute calculator test		
				re	equirement - type Functional subfeature - level Component timeout 90 automatic		
				Steps:			
				1 ru	by /etc/tdriver/visualizer/calculator.rb	Failed	
					returned 1 stdout started: 1: expected 0 stderr duration: 0	3:07:4 3:08:0 0:00:1	9 2 3
lib/gems/1.8/	gems/testability-driver-1	.0.4/lib/tdrive	r/verify/verify.rb:1	L33:in`ve	ify': Verification at /etc/tdriver/visualizer/calculat	or.rb:1	3 fail
verify { calc a	app.QLineEdit(:text =>	4') }					
Ic_app.close ed exception: n /var/lib/gem n /var/lib/gem n /etc/tdriver/	The test object (id: 202 s/1.8/gems/testability-d s/1.8/gems/testability-d visualizer/calculator.rb:1	05, type: "ap river-1.0.4/lib/ river-1.0.4/lib/ 3	plication", name: tdriver/verify/veri tdriver/verify/veri	"calculato ify.rb:95:in ify.rb:95:in	r") has no child object with type or behaviour m `loop' `verify'	ethod	with
				[INFO]	13:08:02 Finished!	1	Î

TDriver

Testability Driver TDriver is a testing tool open sourced by Nokia. It will make test automation possible for Qt applications running on any platform that runs Qt



Android problem

110	I'd like to unit test my Android ap moment.	plication but I found that test driven development in	Android is far from trivial at the
	Any tips, tricks, war stories for bu	uilding light weight and preferably fast running tests	?
5~5	android unit-testing tdd		
111	link edit flag	edited Mar 8 at 9:36	community wiki
			2 revs nyenyec

Android tools

- ADB android debug bridge. A versatile tool lets you manage the state of an emulator instance or Androidpowered device.
- ddmlib java lib from Dalvik Debug Monitor Service (need to recompile due to localhost hardcode)
- Robotium is a test framework created to make it easy to write automatic black-box test cases for Android applications. Open source. Released by Jayway
- Android Emulator a QEMU-based device-emulation tool that you can use to design, debug, and test your applications in an actual Android runtime environment.
- Logcat Lets you read system log messages that are output on an Android device or emulator.
- Junit/TestNG unit testing lib



Eclipse Junit view

Package B	Explorer	Hierarchy	ູ ທີ ່ ງເ	Jnit 🗙				• 🗖	
Finished a	fter 0,01	l seconds		÷	t	- Q		•	
Runs:	1/1	🛛 Error	s: 0		⊠ F	ailures:	0		
∎ [⊠] Failur	res 📑	lierarchy							
= =-::-:							->i	29	
= Fallu	re Trace						-) ≣		

増 Package Explore	er 🔂 JUnit 🖾		
Finished after 14,79	seconds		~
-0 0 ∞ 6	<u> </u>		
Runs: 1/1	Errors: 0	I Failures:	1
🖻 🔚 org.test.Re	allyBlackboxTest [Ru hOpenSettings	nner: JUnit 3]	
Failure Trace			
Joint.framework	AssertionFailedError		
at org.test.Rea at android.test.	llyBlackboxTest.test(InstrumentationTest(lanOpenSettings() Case.runMethod()	ReallyBlack Instrument
at android.test.	InstrumentationTest	Case.runTest(Inst	rumentatio
at android.test.	ActivityInstrumentat	ionTestCase2.run	Test(Activi
at android.test.	Android TestRunner, r Android TestRunner, r	un lest(Android le un Test(Android Te	stRunner.
at android test	InstrumentationTest	Rupper opStart(In	strumenta

at android.app.Instrumentation\$InstrumentationThread.run(Instru

Android instrumentation



MeeGo and Android host based testing architectures



MeeGo and Android testing features

	Android	MeeGo
Execution on real device	Impossible to interact without PC	+
System interaction	DDMS	bash
AUT interaction	Instrumentation	TDriver
Test plan execution	-	Testrunner
Semi-automated testing	-	+
Verify images	-	+
Unit testing	Junit/TestNG	Test::Unit, MiniTest
Monkey testing	+	+
Reporting	No out-of-box solution. Few open-source projects to generate junit report.	Testrunner (QA reports format) TDiver (several formats)

MeeGo testing fails when

- You write app without QT
- Your app doesn't have TDriver support
- You try to run app inside emulator on workstation with non-Intel hardware

Android testing fails when

- You don't have key for already packed app (but you have a chance to resign it)
- You are not familiar with Java
- You need to run tests remotely:
 - Hardcoded "localhost" in ddmlib
 - Hardcoded "localhost" in emulator (port forwarding rule is required)

MeeGo test flow

- Start test case (testrunner) dev machine
- Call TDriver test (via SSH) SUT
- Retrieve results (via SSH)
- Generate report

MeeGo demo test

Android test flow

- Start test case (IDE/Java) dev machine
- Call Instrumentation test (via adb) SUT
- Retrieve results (via adb)
- Generate report

Android demo test

Summary

- MeeGo testing fails if you don't have a real device or Intel hardware.
- Android doesn't provide out-of-box solution for host based automated testing but it can be easily implemented.
- With significant differences in tools and languages there is similar design in Android and MeeGo automation test frameworks.