How to add test cases on JTA

Copyright (C) 2015 FNST LIMITED Copyright (C) 2015 FUJITSU TEN LIMITED

Abstract

This document is used to demonstrate how to add a regular test case to JTA. The newly added test case, as an example for this document, is used to test "touch" command. That is to say, "touch <file>" will be executed on the target machine. If "<file>" is created successfully, the test passes; otherwise it fails.

Important note

This document is a bit obsolete and needs to be updated.

Important details:

- JTA was renamed to Fuego
- Jenkins is configured (by default) to start web interface at http://IPADDR:8080/fuego

Keeping given changes in mind this document is still useful.

- 1. To make explanation easier, we make some assumptions here:
 - a) The machine with JTA installed on it will be called "JTA machine" below. The IP address of JTA machine is 192.168.30.71.
 - b) The machine, on which the test is supposed to be tested, will be called "target machine" below. The IP address of target machine is 192.168.30.64.
- 2. Login to JTA machine as "root" user.
- Use the following command to check whether "Jenkins" service is working.
 # /etc/init.d/jenkins status

If message, like "Jenkins Continuous Integration Server is not running", is showed, please use the following command to start "Jenkins" service. #/etc/init.d/jenkins start

4. The following table lists the files that should be added or fixed in order to add a test case for "touch" command.

file	usage
(optional)	used for selecting "spec" for test cases, so
/home/jenkins/overlays/testplans	that some variables in test_specs will be set
	to satisfy the requirement of the test.
(optional)	used for defining some variables for test.
/home/jenkins/overlays/test_specs	These variables are organized as "spec". In
	different "spec", variables will be defined
	differently
/home/jenkins/tests/Functional.touch/touch-	test start point that will be used to setup
script.sh	the test environment, execute the test and
	grab test result from target machine
/home/jenkins/tests/Functional.touch/touch-	test program that will be executed on the
device.sh	target machine to test "touch" command
/home/jenkins/overlays/boards/porter.board	configuration of target machine, touch-
	script.sh needs this to setup test
	environment
/home/jenkins/scripts/tools.sh	defining variables used to cross-build
	programs for target machine

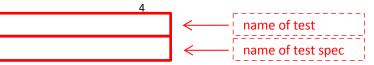
"/home/jenkins/overlays/testplans" and "/home/jenkins/overlays/test_specs" are optional, only used when some special variables are needed for certain tests.

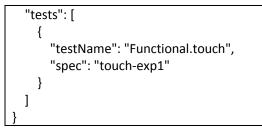
More detailed information will be demonstrated in the next several steps.

5. Add "test plan" (optional)

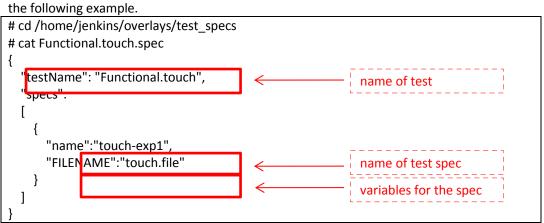
Add "testplan_touch.json" under "/home/jenkins/overlays/testplans", and write it as the following example.



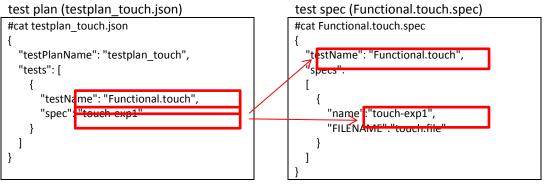




 Add "test spec" (optional) Add "Functional.touch.spec" under "/home/jenkins/overlays/test_specs", and write it as



7. Relationship between "test plan" and "test spec"



8. Add test script

Create folder "Functional.touch" under "/home/jenkins/tests", and under the folder add two files, "touch-script.sh" and "touch-device.sh".

# cd /home/jenkins/tests	
# mkdir Functional.touch	test name
# cat touch-script.sh	
#!/bin <mark>/bash <</mark>	test start point
function test_build { <	function used to build test
echo "test compiling (should be here)"	
}	program
function test_deploy { 🧹	
put \$TEST_HOME/touch-device.sh \$JTA_HOME/jta.\$	STESTDIR/
}	program to the target machine
function test_run {	
assert_define FUNCTIONAL_TOUCH_FILENAME <	function used to execute test
report "cd \$JTA_HOME/jta.\$TESTDIR; ./touch-device	sprogram on the target machine
\$FUNCTIONAL_TOPCH_FILENAME	confirm variables are
}	
function test_processing { <	
log_compare "\$TESTDIR" "1" "PASS\$" "p"	·
log_compare "\$TESTDIR" "0" "FAIL\$" "n"	function used to handle the log
}	of executing test program to
	decide the result of the test
. \$JTA_ENGINE_PATH/scripts/functional.sh <	

Follow the example below to write "touch-script.sh".

Follow the example below to write "touch-device.sh". Be-careful, "touch-device.sh" should script that will call above gain the executable permission in order to be run on target machine.

0		functions to do the test	
# cat <mark>touch-device.sh</mark> #!/bin/bash		 test program 	·='
echo "Touch Founcti	on Test!"		
file=/tmp/\$1			
rm –f \$file	<	file that will be "touch"ed	!
touch \$file	1	clean environment to avoid	
		the file already exists	
if [-f \$file];then echo "PASS"	<	"touch" the file	
else echo "FAIL"		Lifebo filo outoto outouto	
fi	<u> </u>	 if the file exists, output "PASS", otherwise, output 	
rm –f \$file		"FAIL"	

9. Eix configuration of larget machine

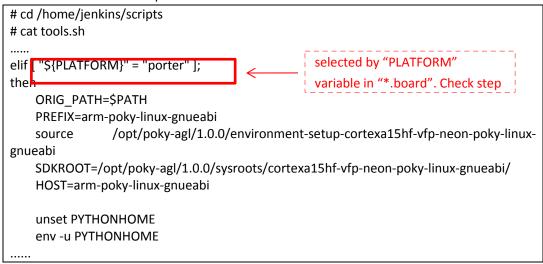
Follow the example below to fix "/home/jenkins/overlays/boards	porter's related configuration, "porter.board" under
# cd /home/jenkins/overlays/box	ards
# cat qemu-arm.board	
inherit "base-board"	
include "base-params"	
	IP address or hostname of target machine
€	user name for ssh login
	password for ssh login, not needed for AGL
	yet

IPADDR="192.168.30.64"	
LOGIN="root"	
JTA_HOME="/home/a"	
#PASSWORD="root"	
PLATFORM="porter"	
TRANSPORT="ssh"	
ARCHITECTURE="arm"	
#SATA_DEV="/dev/sda1" #SATA_MP="/mnt/sata"	
#USB_DEV="/dev/sdb1"	
#USB_MP="/mnt/usb"	
—	
#MMC_DEV="/dev/mmcblk0p2"	
#MMC_MP="/mnt/mmc"	

If you want to execute the test on other target machine, fix the related "*.board" file. You can also refer to "jta-guide.pdf" for more detailed information.

10. Fix variable definition used for corss-building

Fix "tools.sh" under "/home/jenkins/scripts". Variables, like SDKROOT, PREFIX, HOST, and "source" are used to setup cross-build environment.



11. Logon to JTA web interface. The URL should be "192.168.30.71:8080" here:

🧕 0. History [Test Auto	0. History [Test Auton	nation Framev	vork] – Mozilla Fi	refox	_ 8
(3 192.168.30.71 :8080	0	~ 0	🗄 🔹 🚺 🖌 Goog	gle 🔍	☆ 🖻 » 🗏
		LONG TERM SUPPORT INITIATIV	RENES	SAS Search	1
Home				ENA	BLE AUTO REFRESH
 See Test People Test Run History Edit Dashboard Manage Jenkins 	0. History Benchmarks Latest tests runs		nework	(add description
📄 Scriptler	Test	Run ·	Time	Platform SDK	Device
Exclusion administration	Functional.LTP.Net Stress	<u>●#1</u>	Nov 1, 2015 3:01:26 AM		porter
Test Run Queue	Functional.LTP.Syscalls	<u>_#1</u>	Nov 1, 2015 2:38:14 AM		porter
No test runs in the queue.					
Targets Status # Master	Test Run statistics Status of the test run	Des	scription N	lumber of test rui	IS

12. Click "Functional" tag, then click "New Test" to create a new test case

	Functio	onal [Test /	Automation Framework]	– Mozilla Fire	fox	_ =
🧕 Functional [Test Aut	. × 🕂					
• @ 192.168.30.71:808	30/view/Fur	nctional/	× Ċ 🔹	8 ✔ Google	Q 🏠 🖻	€ ↓ »
			LONG TERM	RENES	search	
Home Functional					ENAB	LE AUTO REFRESH
 Beople Test Run History Edit Dashboard Delete Dashboard 	0. History	Benchmarks	Functional all batch runs	+	2	add description
💥 Manage Jenkins 📴 Scriptler	s	w	Test Name w∕ Status Color ↓	Test Priority	Last Duration	Device
Exclusion administration		ø	Functional.aiostress	360	N/A	N/A
Test Run Queue		ø	Functional.arch_timer	360	N/A	N/A
No test runs in the queue.		Ð	Functional.bzip2	360	N/A	N/A
Targets Status						
Targets Status # <u>Master</u>		Ð	Functional.cmt	360	N/A	N/A

13. Input "Functional.touch" for "Test name". Then check "Copy existing Test", input "Functional.bzip2". After all, click "OK".

	New Test [Test Automation Framework] - Mozilla Firefox
🙎 New Test [Test Auto	× +
(192.168.30.71 :808	30/view/Functional/newJob 🗸 🖒 🖌 🖉 🖉 Google 🔍 🛠 🛍 🖊 » 🚍
Home Functional	
쯜 New Test	Test name Functional touch
🍓 People	O Inheritance Project
📄 Test Run History	This is a project that allows basic inheritance of properties to occur. It is broadly compatible with the free-style-project class for most purposes, but only allows inheritance to occur between projects of this inheritance-enabled class.
🚫 Delete Dashboard	Test a free-style software project
💥 Manage Jenkins 📄 Scriptler	This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.
Exclusion administration	🔿 Build a maven2/3 project
Test Run Queue	Build a maven 2/3 project. Jenkins takes advantage of your POM files and drastically reduces the configuration.
No test runs in the queue.	O Monitor an external test
Targets Status # Master 1 Icle	This type of job allows you to record the execution of a process run outside Jenkins, even on a remote machine. This is designed so that you can use Jenkins as a dashboard of your existing automation system. See the documentation for more details.
2 Idle	Suitable for projects that need a large number of different configurations, such as testing on multiple
koelsch	environments, platform-specific builds, etc.
1 Idle	Copy existing Test
1 Idle	Copy from Functional.bzip2 Functional.bzip2
1 Idle	ОК

14. Fix configurations related to the test

1) test description:

Fu	nctional.touch Config	Test Automation Framev	vork] – Mozilla	Firefox	-	• ×
G Functional.touch Con	×+					
(192.168.30.71 :8080/	view/Functional/job/Fund	tional.tc 🗙 🥑 💣 🚺	Google 🖸	☆ 🖻	∔ n »	≡
		LONG TERM SUPPORT INITIATIVE	RENES	search		
Home Functional Functional.t	ouch configuration					
🛧 Back to Dashboard	Project name	Functional	.touch]
🔍 Status	Description	touch f	unction test su:	ite		
🔁 Changes						
🔚 Workspace						
🔊 Run Test Now		Preview				
🚫 Delete Test	💞 Discard Old Time Runs					
💥 Configure Test	Strategy	Log Rotati	ion		~	
Test Run History (<u>trend</u>)	Grategy		eep builds		Ť	^
RSS for all RSS for failures		Days to k				×
			if not emp clays	ty, build records are on	ly kept up to this number o	of .
		Max # of	builds to keep 300			0
	Save Apply					

2) test name:

	c 🕆 C 🔹 🚺 🗙 Google	Q ☆ 自 ∔ ⋒	>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
me Functional Functional.touch configuration			
	Dynamic Choic	ce Parameter (Scriptler)	
	Name	TESTPLAN	2
	Description		0
	Remote Script		
	Choice Type	Single Select 🗸	
	Readonly Input Fie	id 🗌	
	Script	getTestplans 🗸	
	Parameters	Parameter name	
	~	test_name Parameter	
		alue	
		Functional.touch	

3) test start point, it should be "touch-script.sh" here:

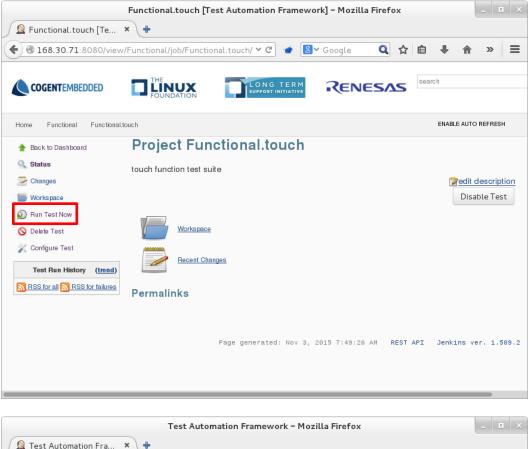
	Functional.touch Config [Test Automation Framework] - Mozilla Firefox	_ = ×
🖉 🖉 Functional.touch Co	on × 👌	
€ € 192.168.30.71:8	1080/view/Functional/job/Functional.tc 🌱 🕑 🐲 🚺 🗙 Google 🛛 🔍 😭 🖶	r
Home Functional Func	ctional.touch configuration	
	Test Run	
	Execute shell	0
	Command if [! -d "/logs/\$JOB_NAME"]; then mkdit -p "/logs/\$JOB_NAME"; echo \$TESTFIAN >/logs/\$JOB_NAME/last_used_testFlow; TESTFIAN=testFlows/\$TESTFIAN.ison source/tests/\$JOB_NAME touch-script.gb See the list of available environment variables Delu Add test execution step Post-test actions.	
	Post build task	0
	Log text ssh	0
	Save Apply	

4) click "Apply", then the new test case is created:

Functional.touch Config [Test Automation Framework] – Mozilla Firefox	_ 🗉 ×
📓 Functional.touch Con × 🕂	
🗲 🖲 192.168.30.71:8080/view/Functional/job/Functional.tc 🗸 😋 🔹 🔞 🗸 Google 🔍 😭 🖨 🖡 🏦	» ≡
Home Functional Functional.touch configuration Add another task Delete Groovy Postbuild Groovy Script: Binding binding = new Binding(); binding.setVariable('manager', manager'); Groovy Shell shell = new Groovy Shell(binding); shell.evaluate(new File('/home/jenkins/scripts/postbuild.groovy').text); If the script fails: Do nothing	•
Add post-test action Save Apply	_
Page generated: Nov 3, 2015 7:34:56 AM REST API Jenkins ve	r. 1.509.2

15. Clike "Run Test Now" on the left side.

Choose "porter" for "Device", check "Rebuild" and choose "testpaln_touch" for TESTPLAN. Then click "Run test" to start the test. The test progress will be showed in "Test Run History".



🙎 Test Automation Fra 🗴 💠			
(168.30.71 :8080/view	/Functional/job/Functional.touch/ 🗸 🍲 🕙 🗸 Google 🔍 🏠 🖨 🔸 🎓 » 🚍		
Home Functional Functional.touch			
摿 Back to Dashboard	Project Functional.touch		
🔍 Status	This test run requires parameters:		
🔁 Changes	Device porter V		
👹 Workspace	(target)		
🔊 Run Test Now	Reboot		
🚫 Delete Test	I checked larget device will be rebooled <u>before</u> running test.		
💥 Configure Test	Rebuild 🧭		
Test Run History (<u>trend</u>)	If checked all existing build instances of the lest suite will be deteled and test suite will be rebuilt from tarball.		
<u> RSS for all 💦 RSS for failures</u>	Target Cleanup		
	TESTPLAN testplan_touch		
	Run test		
	Page generated: Nov 3, 2015 7:49:37 AM Jenkins ver. 1.509.2		
	Fage generated: NOV 3, 2015 7:49:37 AM Senkins Ver. 1.309.2		

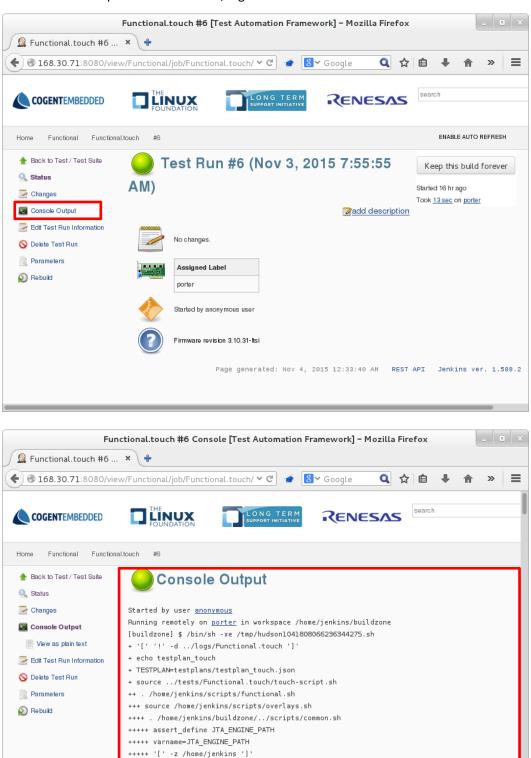
	Functional.touch [Test Automation Framework] – Mozilla Firefox	_ = ×
🙎 Functional.touch [Te	× +	
(168.30.71 :8080/view	v/Functional/job/Functional.touch/ 🌱 🧭 🔞 🗸 🖨 🖡	⋒
	FOUNDATION FOUNDATION Search	
Home Functional Functional.	Llouch ENABLE	AUTO REFRESH
摿 Back to Dashboard	Project Functional.touch	
Status Changes Workspace Run Test Now Delate Test Configure Test	touch function test suite	edit description Disable Test
Test Run History (trend) #6 Nov 3, 2015 7:55:55 AM RSS for all RSS for failures	Recent Changes Permalinks Page generated: Nov 3, 2015 7:55:55 AM REST API Jenki	ns ver. 1.509.2

16. If the test succeeded, a line with a green icon in front of it will be showed; otherwise, a red icon will be showed.

Click this line to get more information about this test.

Functional	.touch [Test Automation Framework] – Mozilla Firefox
📓 Functional.touch [Te 🗙 🖶	
• 168.30.71:8080/view/Functional/j	ob/Functional.touch/ 🗸 🎓 🛣 😸 🗸 Google 🔍 🏠 🖨 🔸 🎓 🔉 🚍
Home Functional Functional.touch	ENABLE AUTO REFRESH
📤 Back to Dashboard	Project Functional.touch
🔍 Status	touch function test suite
Changes	Zedit description
Workspace	Disable Test
🔊 Run Test Now	
🚫 Delete Test	Workspace
💥 Configure Test	Recent Changes
Test Run History (trend)	Pecelit Gladiges
🥥 #6 Nov 3, 2015 7:55:55 AM porter / 3.10.31-ltsi	Permalinks
RSS for all RSS for failures	
	Page generated: Nov 3, 2015 7:55:55 AM REST API Jenkins ver. 1.509.2

17. Click "Console Output" on the left side, log of the test will be showed.



+++++ export JTA_PARSER_PAT\#>/home/jenkins/scripts/parser +++++ JTA_PARSER_PAT\#>/home/jenkins/scripts/parser +++++ export JTA_LOGS_PAT\#>/home/jenkins/logs

+++++ export JTA_ENGINE_PAT⊢/home/jenkins +++++ JTA_ENGINE_PAT⊢/home/jenkins