Continuous Performance Testing

Shopware Developer Conference

Kore Nordmann (@koredn)

08. June 2013

About Me

Kore Nordmann
@koredn

Co-founder of

Helping people to create high quality web applications.

`http://qafoo.com`

- Expert consulting
- Individual training

Get a training on object oriented design for your team!

Outline

Motivation

Conclusion

Motivation

- Why should we do performance tests?
  - Locate unknown bottlenecks
  - Mesaure behaviour of the full stack
- Why should we do that continuously?
  - Find performance regressions
  - Ensure optimizations are persistent

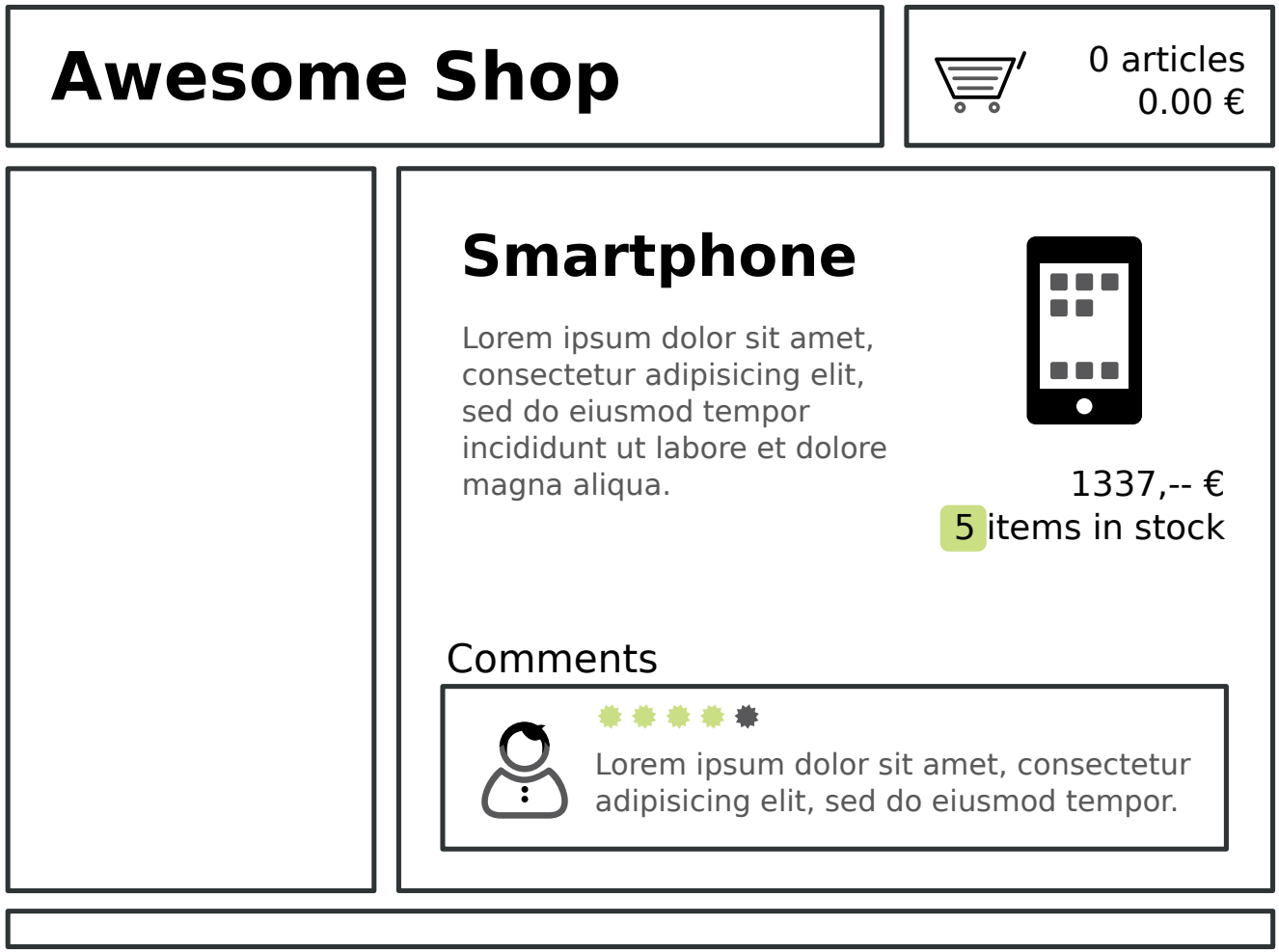
Often used tools

- Often misused tools
  - siege
  - ApacheBench (ab)
- Testing for micro-optimizations
  - Evaluating Hello-World-examples of Frameworks
- Useful tools, I won't talk about
  - xDebug Profiling
  - xhProf
  - Database profiling tools
  - System metrics (Graphite, . . .)

The problem is more complex

- Your task: Create a new webshop
  - Assume it's march
  - The deadline is October this year, right before Christmas

A webshop



A real tool for performance tests

- JMeter
  - Complex user paths
  - Concurrend requests
  - Record on proxy
  - Clustering

JMeter 101

- Thread Group
  - An execution plan (like a user registration)
- Controller
  - Controls how samplers are executed (loop, random, . . .)
- Config Element
  - Configration, optionally from external sources
- Timer
  - Defining timing constraints for sampler execution
- Sampler
  - Perform the actual work (like HTTP requests)

Getting started

- Create a test plan
  - What do users actually do on your site?
- Example:
  - Random browser
  - User registration
  - Sign on
  - Shopping with checkout
  - Commenting products

File Edit Run Options Help

0 / 0

- Random Noise Tesplan
  - HTTP Default Settings
  - HTTP Cookie Manager
  - HTTP Cache Manager
  - HTTP Header Manager
  - Random Noise
    - Random noise urls
    - Loop Controller
      - HTTP Request HTTPClient
      - Gaussian Random Timer
  - Summary Report
- WorkBench

Test Plan

Name: Random Noise Tesplan

Comments:

User Defined Variables

Name:	Value

Add Delete

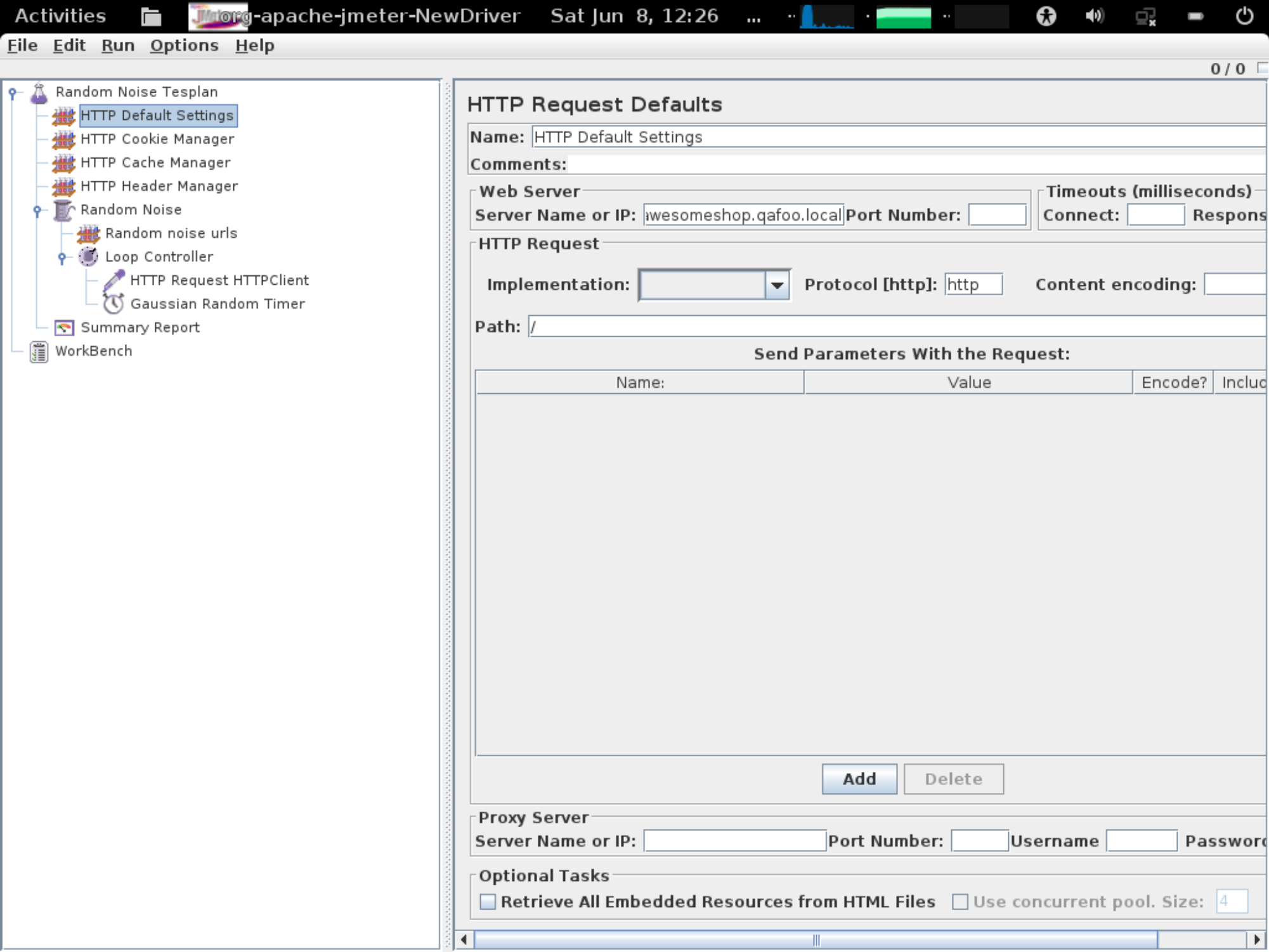
☐ **Run Thread Groups consecutively (i.e. run groups one at a time)**

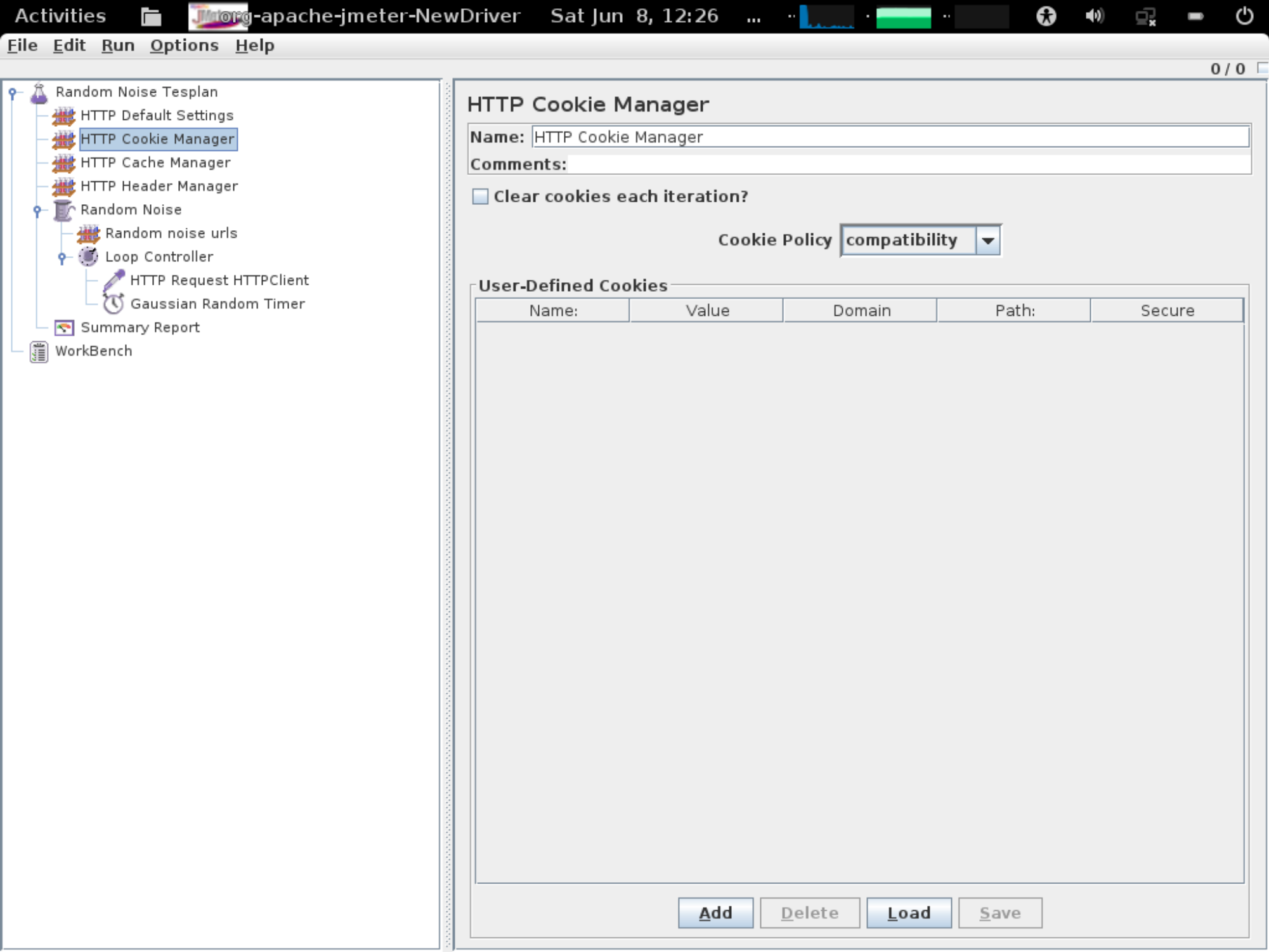
☐ **Functional Test Mode (i.e. save Response Data and Sampler Data)**

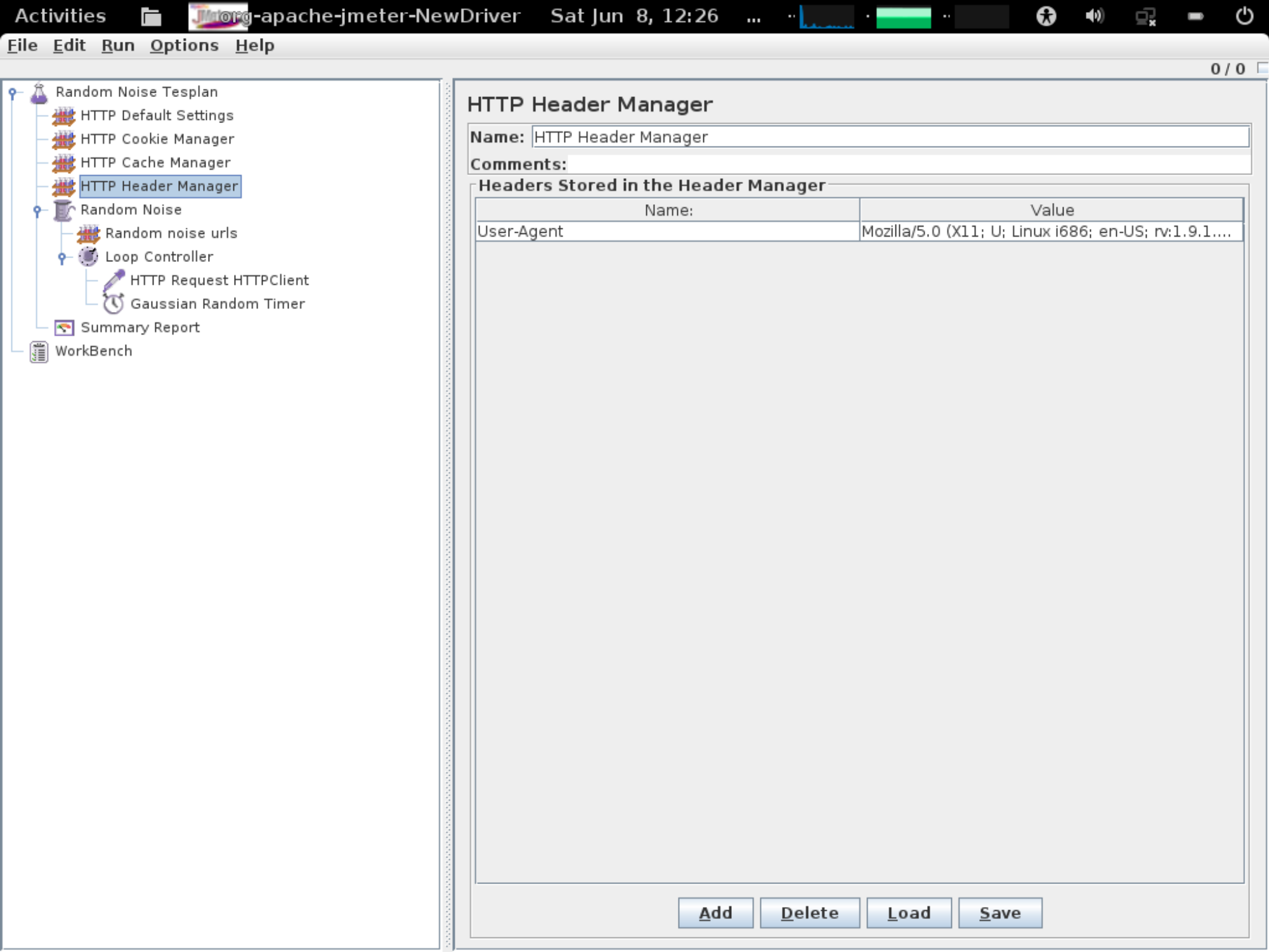
Selecting Functional Test Mode may adversely affect performance.

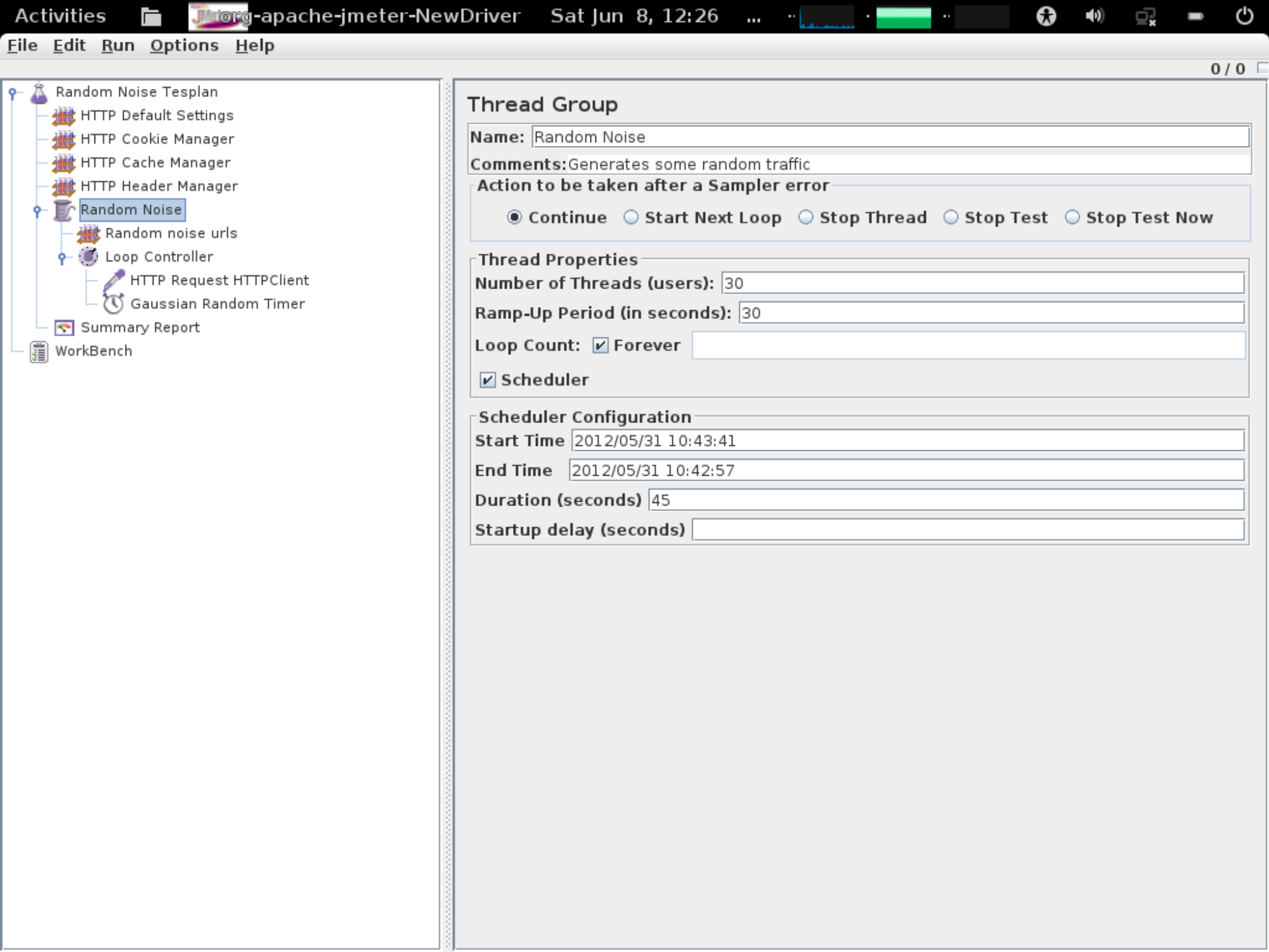
Add directory or jar to classpath Browse... Delete Clear

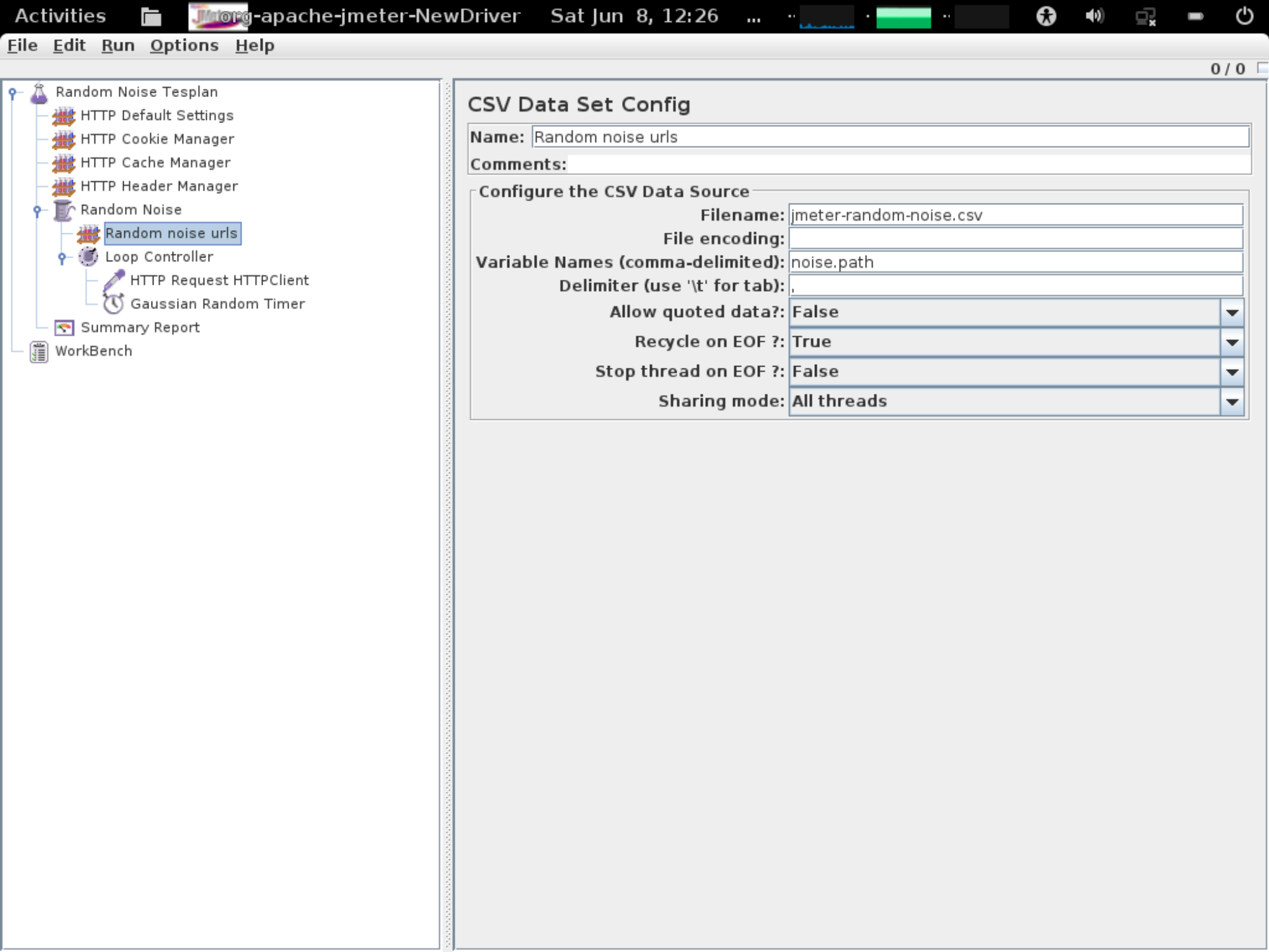
Library

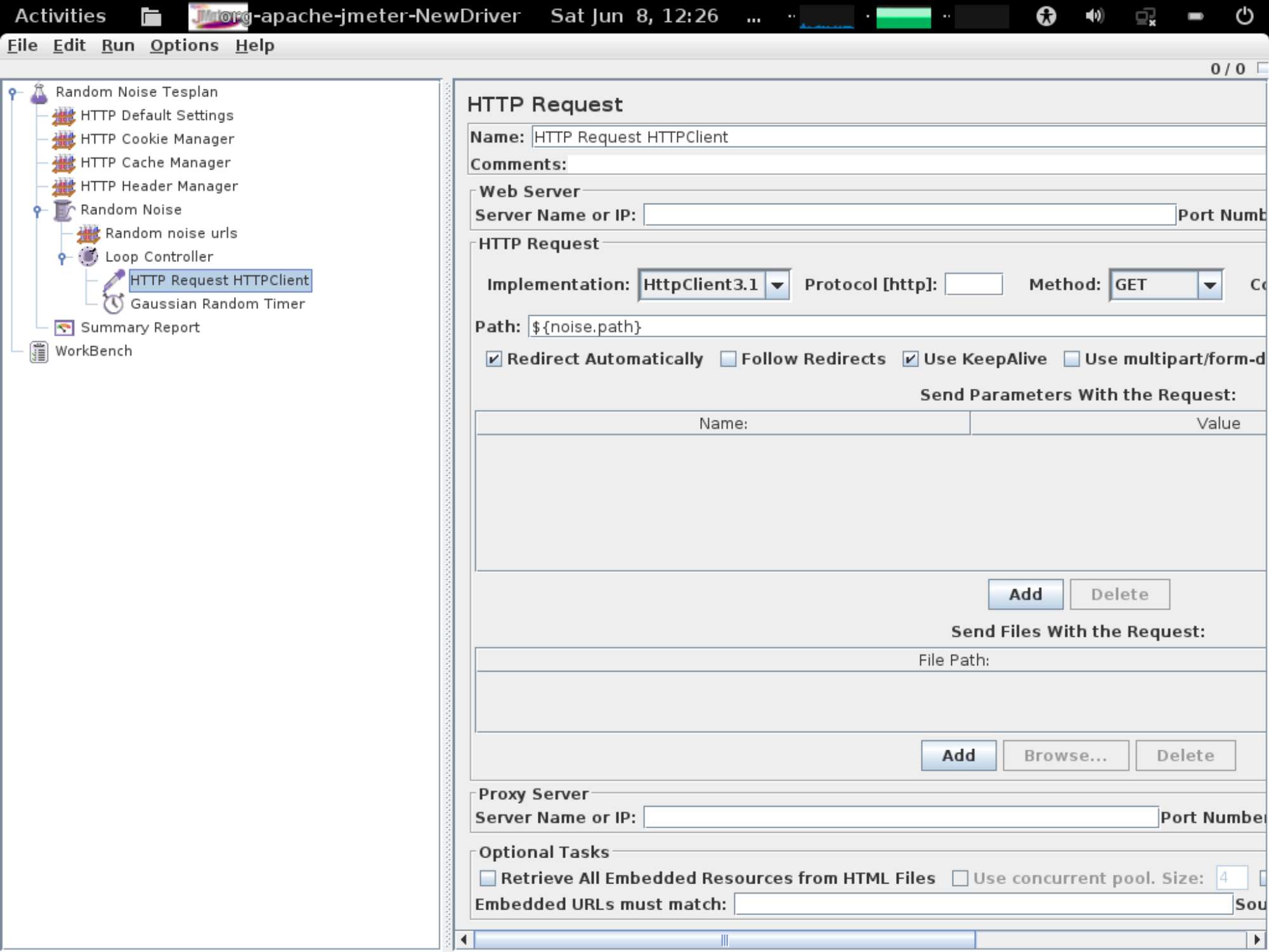

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
HTTP Request Defaults
Name: HTTP Default Settings
Comments:
Web Server
Server Name or IP: awesomeshop.qafoo.local
Port Number:
Timeouts (milliseconds)
Connect:
Respons
HTTP Request
Implementation:
Protocol [http]: http
Content encoding:
Path: /
Send Parameters With the Request:
Name:
Value
Encode?
Add
Delete
Proxy Server
Server Name or IP:
Port Number:
Username
Passwor
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size: 4


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
HTTP Cookie Manager
Name: HTTP Cookie Manager
Comments:
Clear cookies each iteration?
Cookie Policy compatibility
User-Defined Cookies
Name: Value Domain Path: Secure
Add Delete Load Save

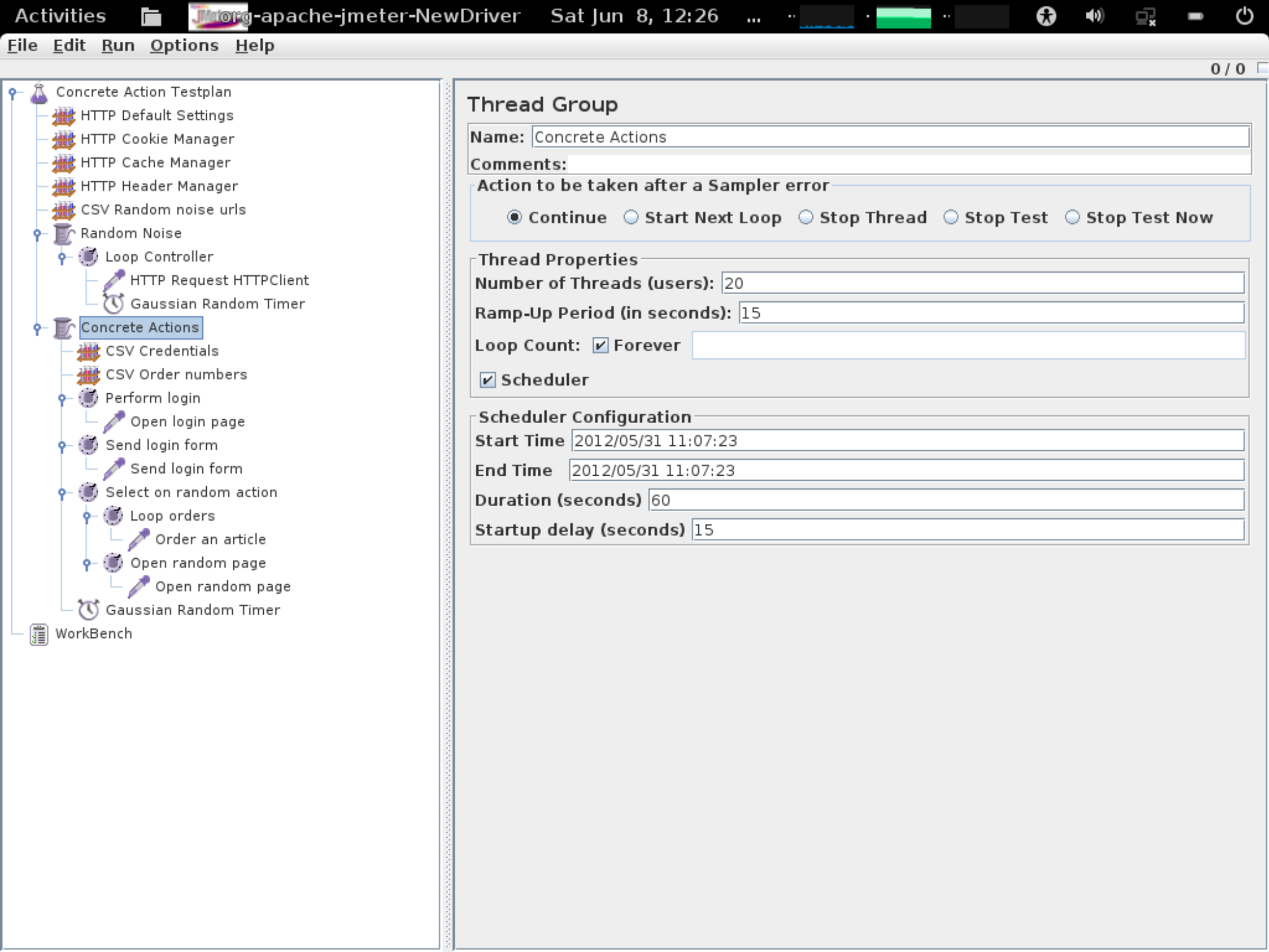

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
HTTP Header Manager
Name: HTTP Header Manager
Comments:
Headers Stored in the Header Manager
Name:
Value
User-Agent
Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.1....
Add
Delete
Load
Save

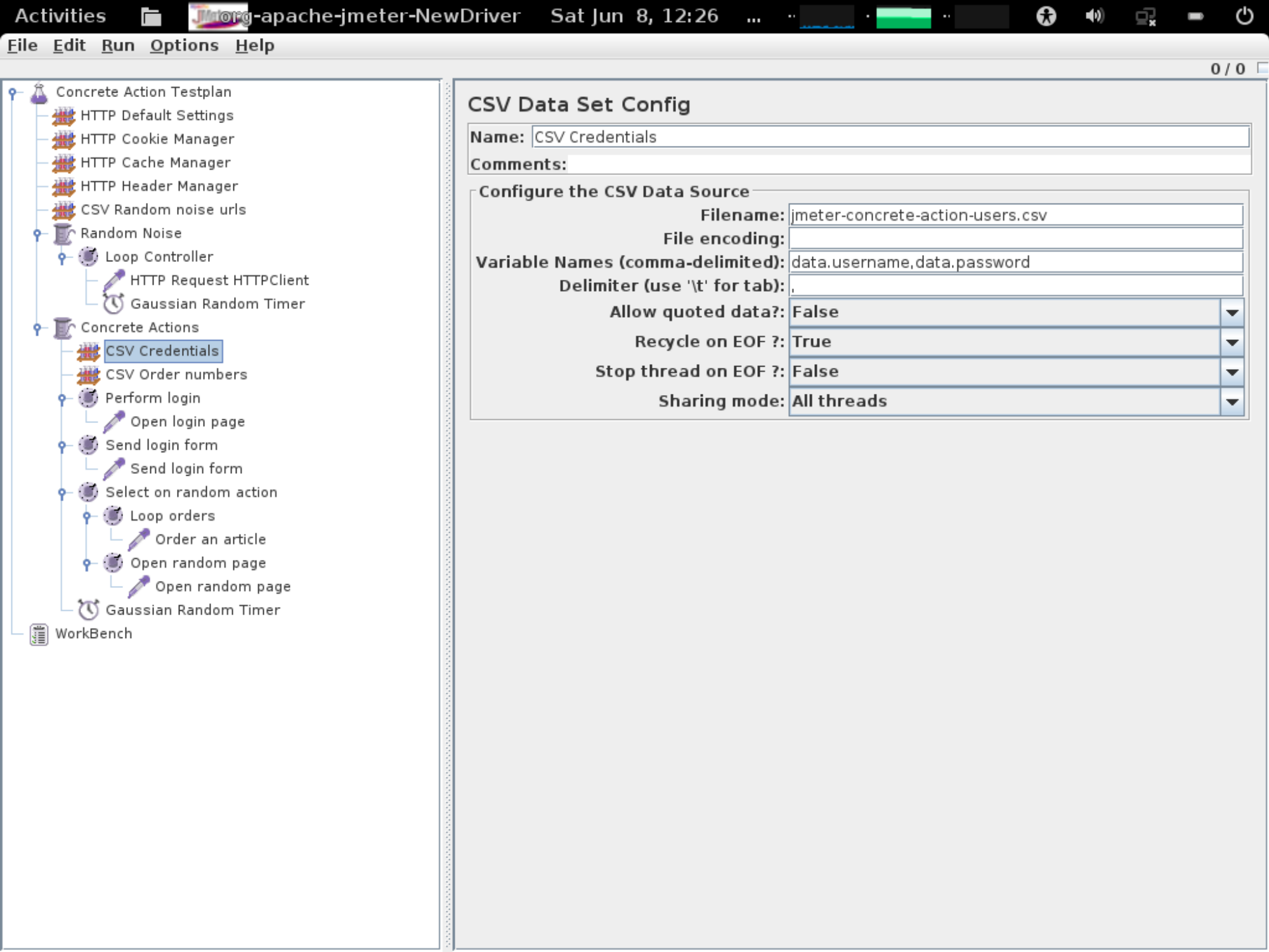

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
Thread Group
Name: Random Noise
Comments: Generates some random traffic
Action to be taken after a Sampler error
Continue
Start Next Loop
Stop Thread
Stop Test
Stop Test Now
Thread Properties
Number of Threads (users): 30
Ramp-Up Period (in seconds): 30
Loop Count: Forever
Scheduler
Scheduler Configuration
Start Time 2012/05/31 10:43:41
End Time 2012/05/31 10:42:57
Duration (seconds) 45
Startup delay (seconds)

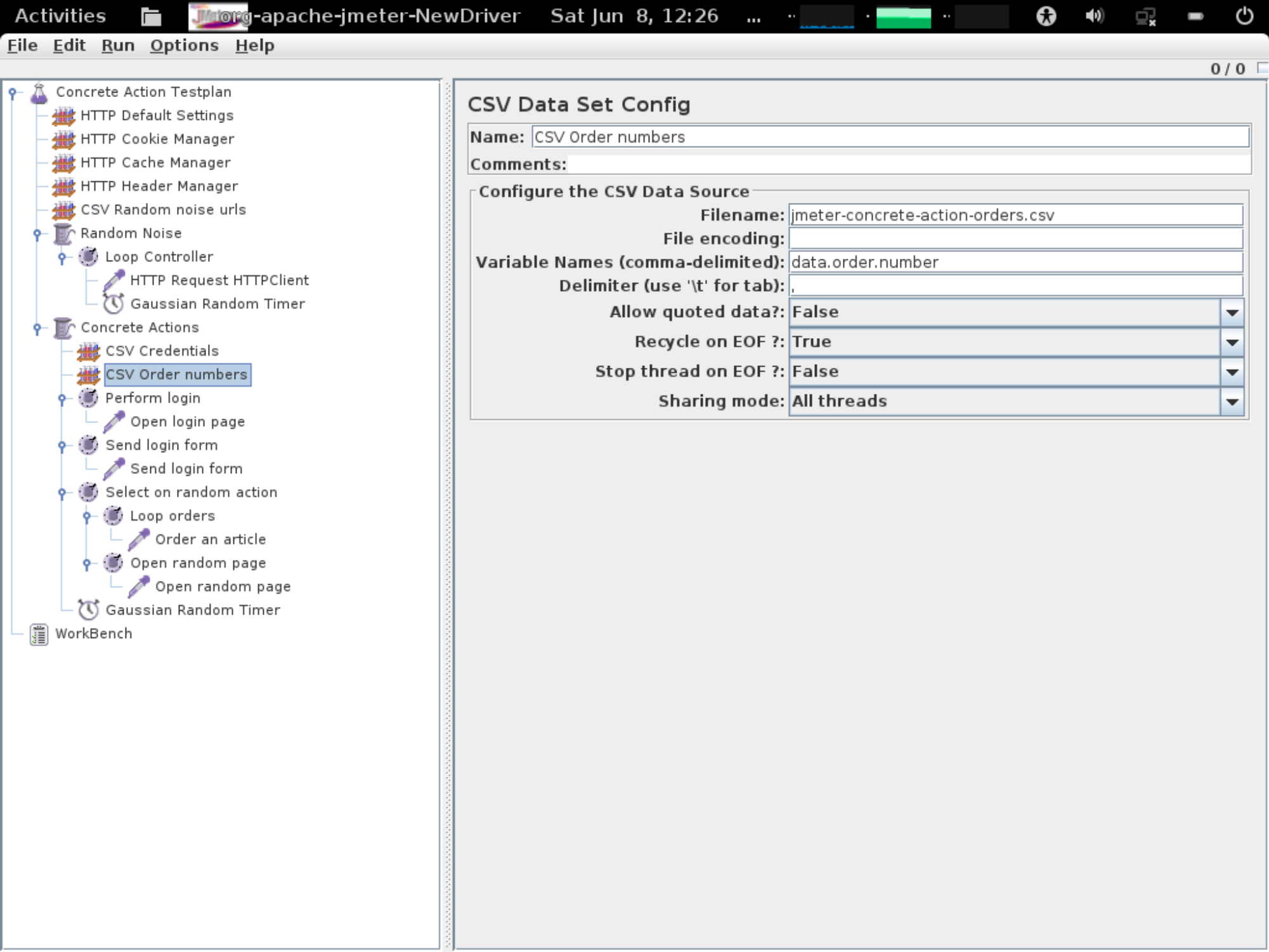

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
CSV Data Set Config
Name: Random noise urls
Comments:
Configure the CSV Data Source
Filename: jmeter-random-noise.csv
File encoding:
Variable Names (comma-delimited): noise.path
Delimiter (use '\t' for tab): ,
Allow quoted data?: False
Recycle on EOF ?: True
Stop thread on EOF ?: False
Sharing mode: All threads

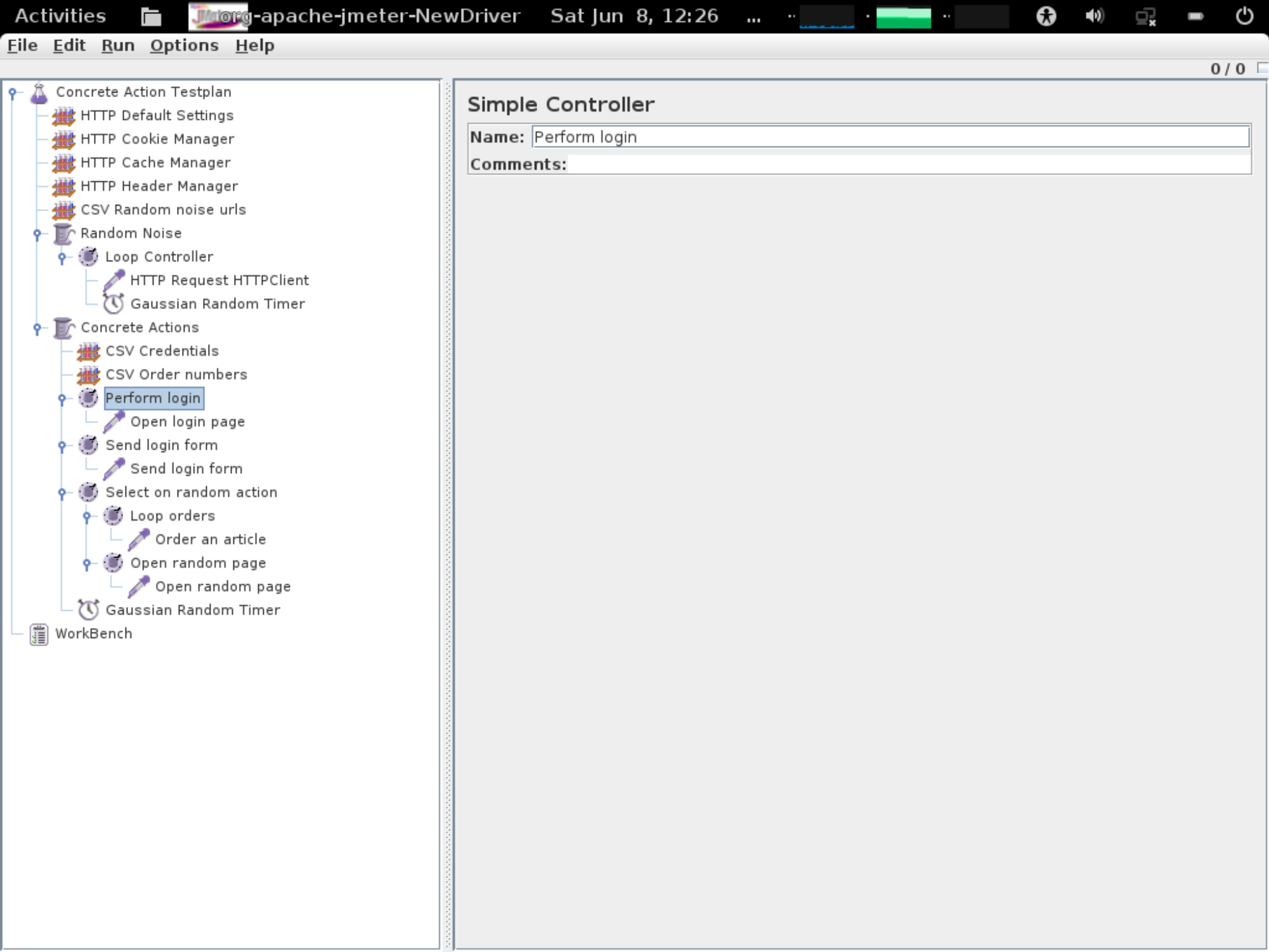

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
HTTP Request
Name: HTTP Request HTTPClient
Comments:
Web Server
Server Name or IP:
Port Numb
HTTP Request
Implementation: HttpClient3.1
Protocol [http]:
Method: GET
Path: ${noise.path}
Redirect Automatically
Follow Redirects
Use KeepAlive
Use multipart/form-d
Send Parameters With the Request:
Name:
Value
Add
Delete
Send Files With the Request:
File Path:
Add
Browse...
Delete
Proxy Server
Server Name or IP:
Port Number
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size:
4
Embedded URLs must match:

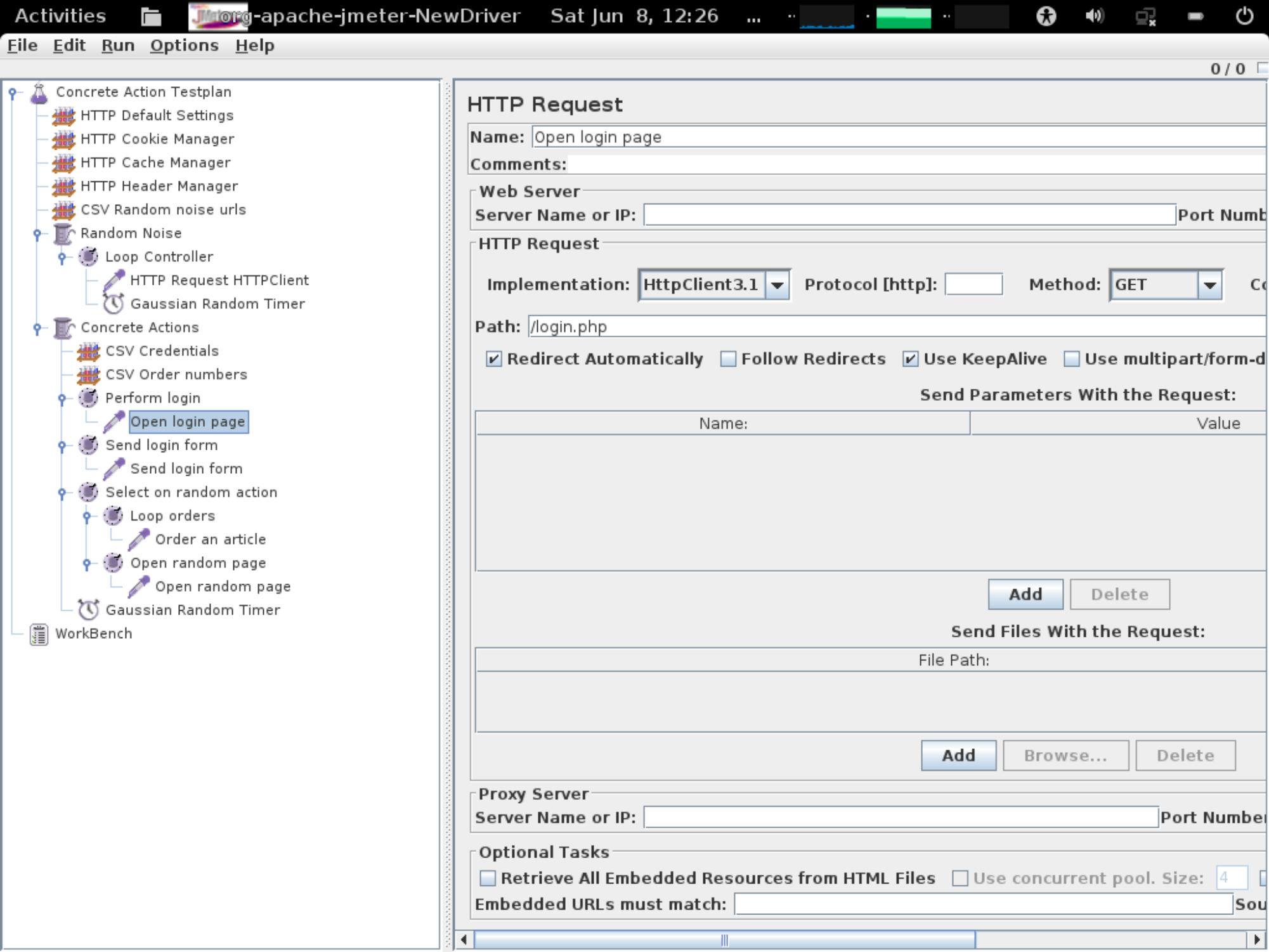
Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Random Noise Tesplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
Random Noise
Random noise urls
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Summary Report
WorkBench
Gaussian Random Timer
Name: Gaussian Random Timer
Comments:
Thread Delay Properties
Deviation (in milliseconds): 500.0
Constant Delay Offset (in milliseconds): 1000

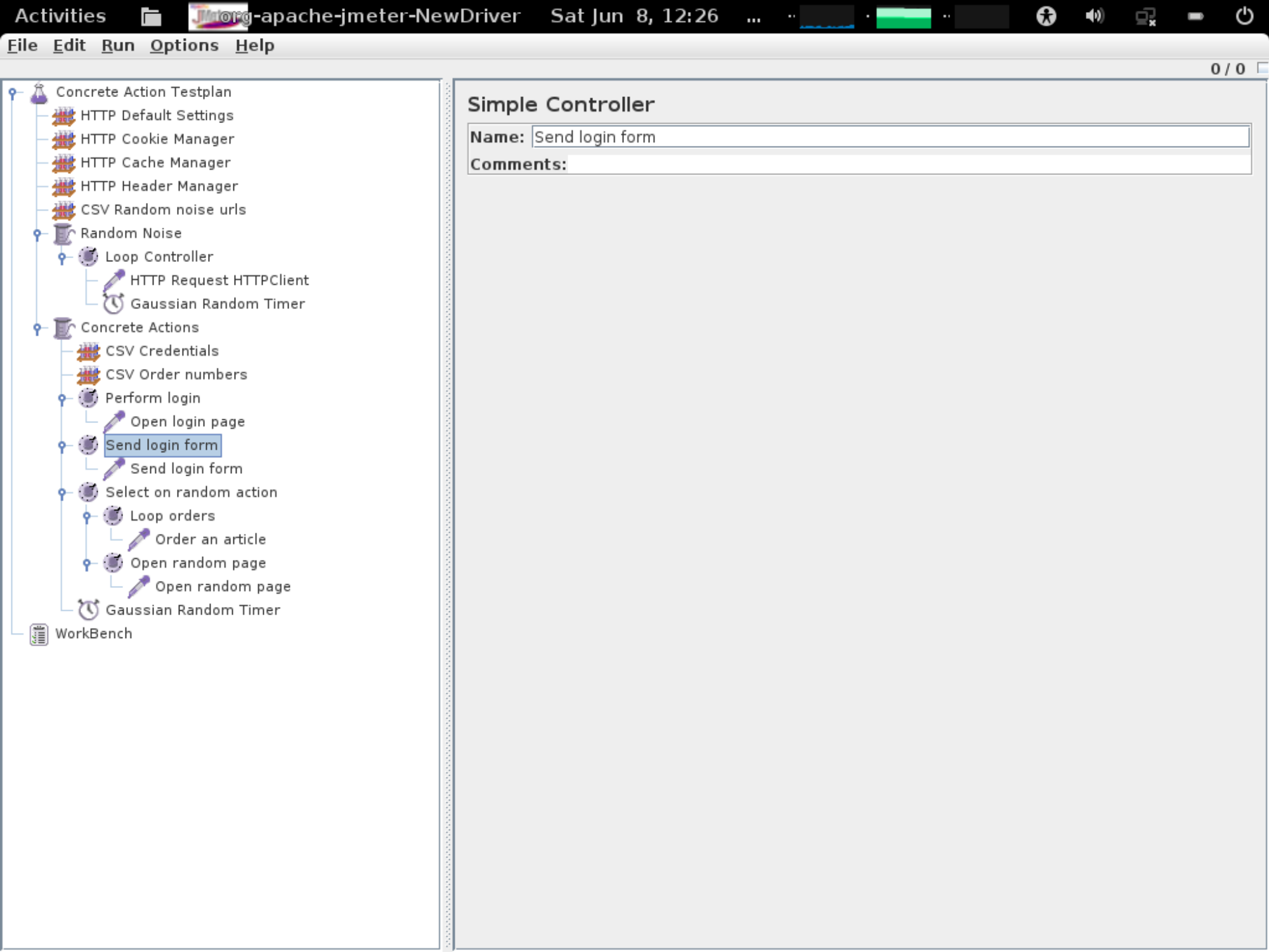

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Thread Group
Name: Concrete Actions
Comments:
Action to be taken after a Sampler error
Continue
Start Next Loop
Stop Thread
Stop Test
Stop Test Now
Thread Properties
Number of Threads (users): 20
Ramp-Up Period (in seconds): 15
Loop Count: Forever
Scheduler
Scheduler Configuration
Start Time 2012/05/31 11:07:23
End Time 2012/05/31 11:07:23
Duration (seconds) 60
Startup delay (seconds) 15

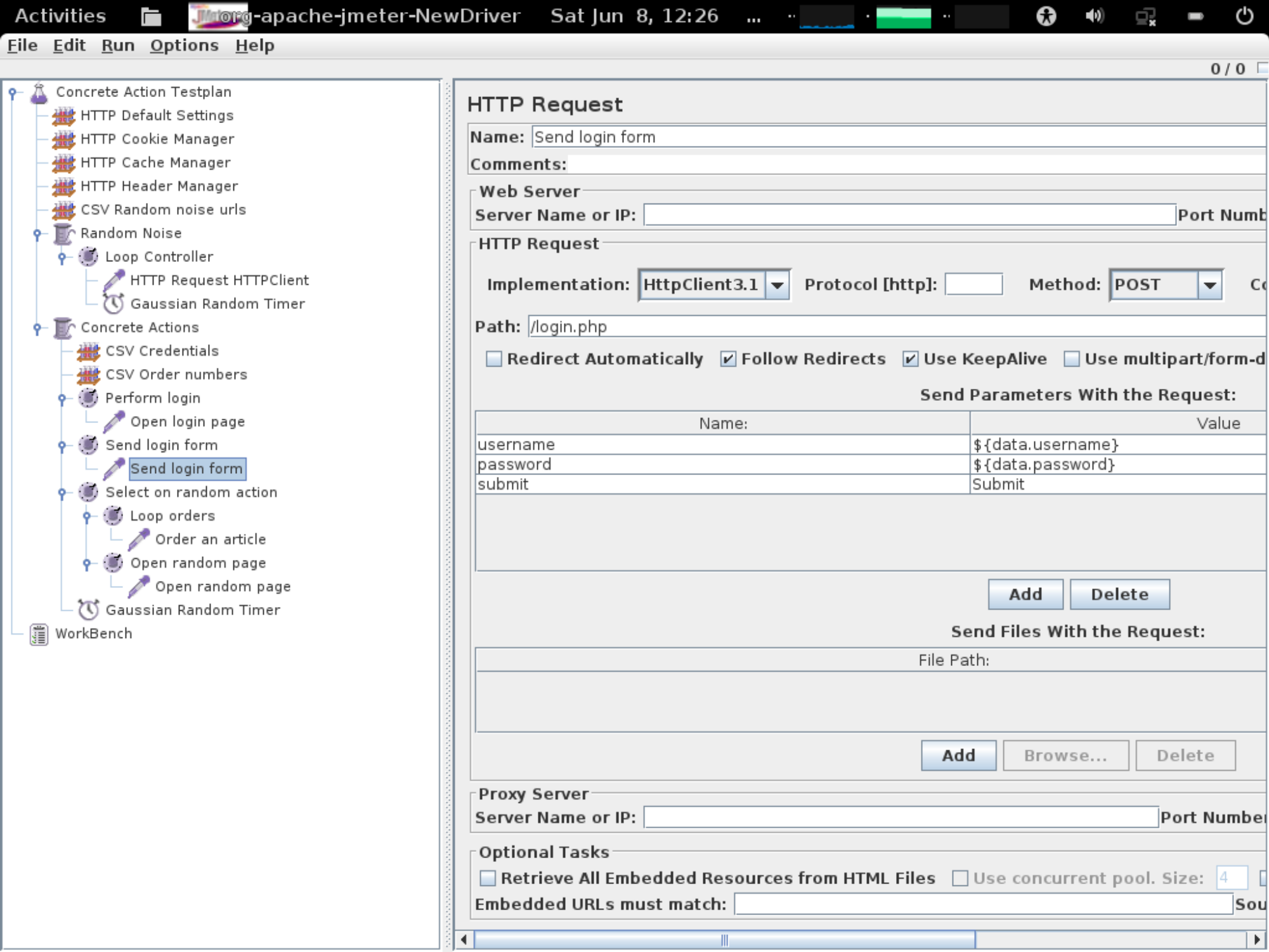

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
CSV Data Set Config
Name: CSV Credentials
Comments:
Configure the CSV Data Source
Filename: jmeter-concrete-action-users.csv
File encoding:
Variable Names (comma-delimited): data.username,data.password
Delimiter (use '\t' for tab): ,
Allow quoted data?: False
Recycle on EOF ?: True
Stop thread on EOF ?: False
Sharing mode: All threads

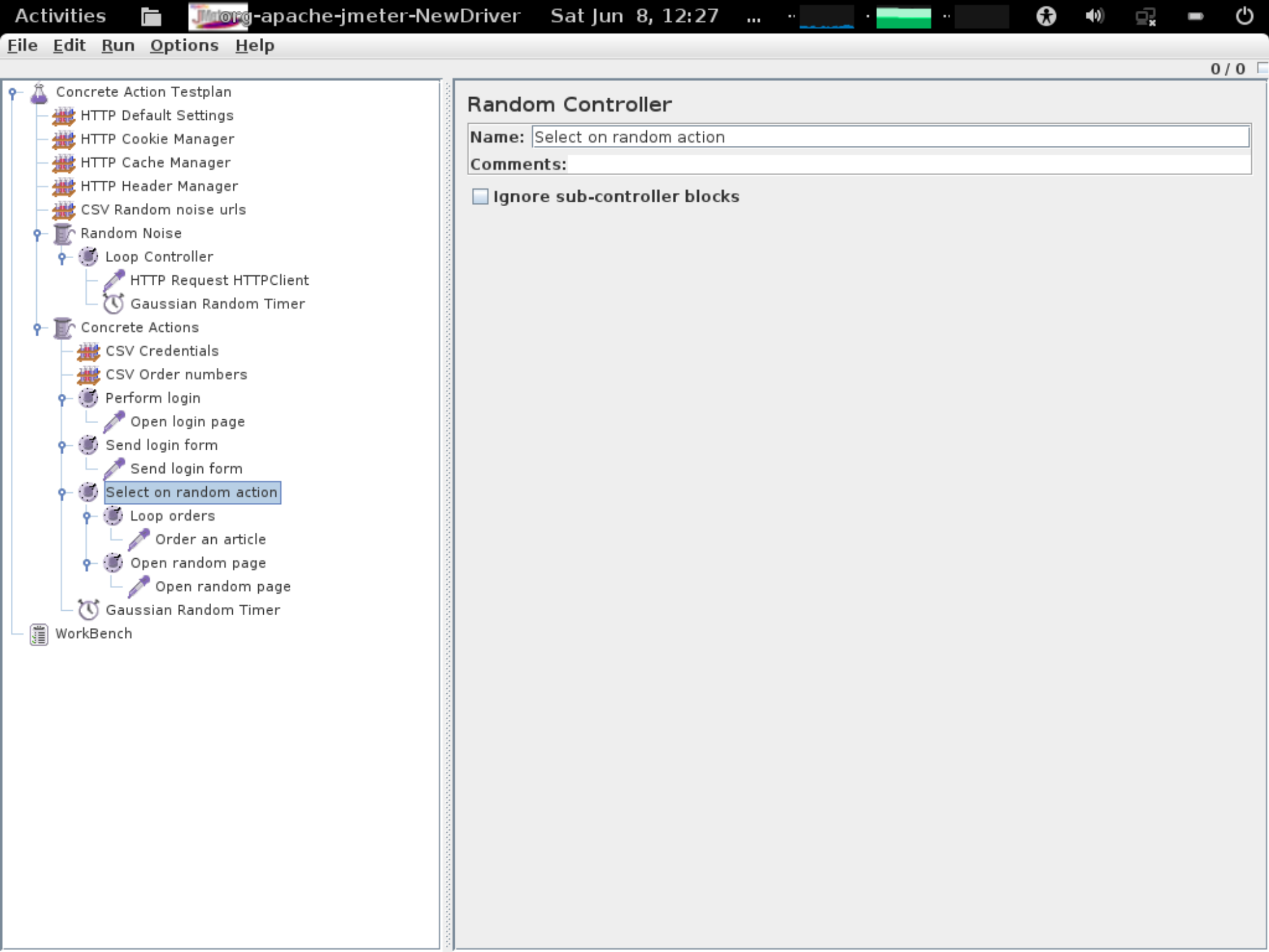

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
CSV Data Set Config
Name: CSV Order numbers
Comments:
Configure the CSV Data Source
Filename: jmeter-concrete-action-orders.csv
File encoding:
Variable Names (comma-delimited): data.order.number
Delimiter (use '\t' for tab): ,
Allow quoted data?: False
Recycle on EOF ?: True
Stop thread on EOF ?: False
Sharing mode: All threads

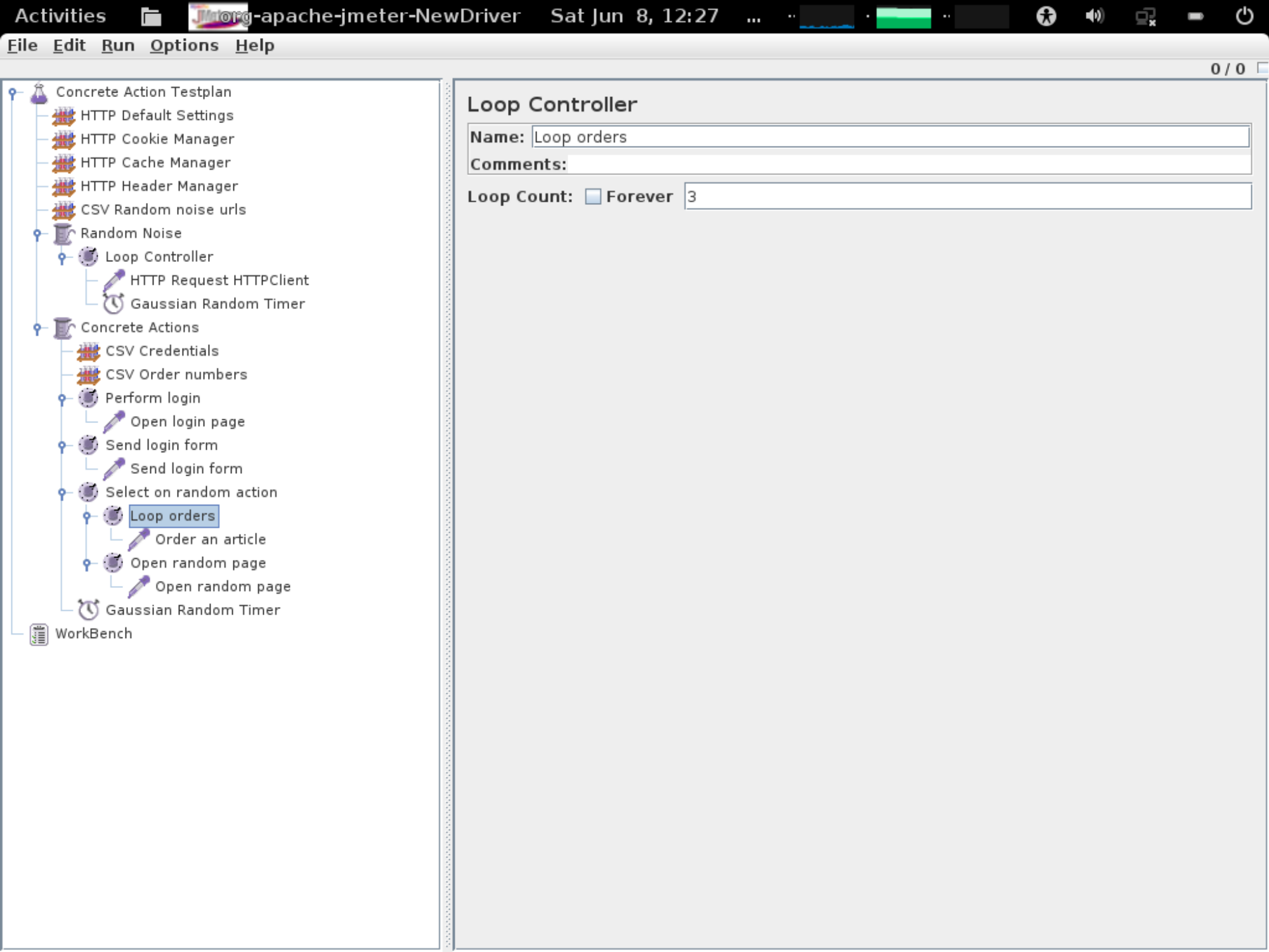

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Simple Controller
Name: Perform login
Comments:

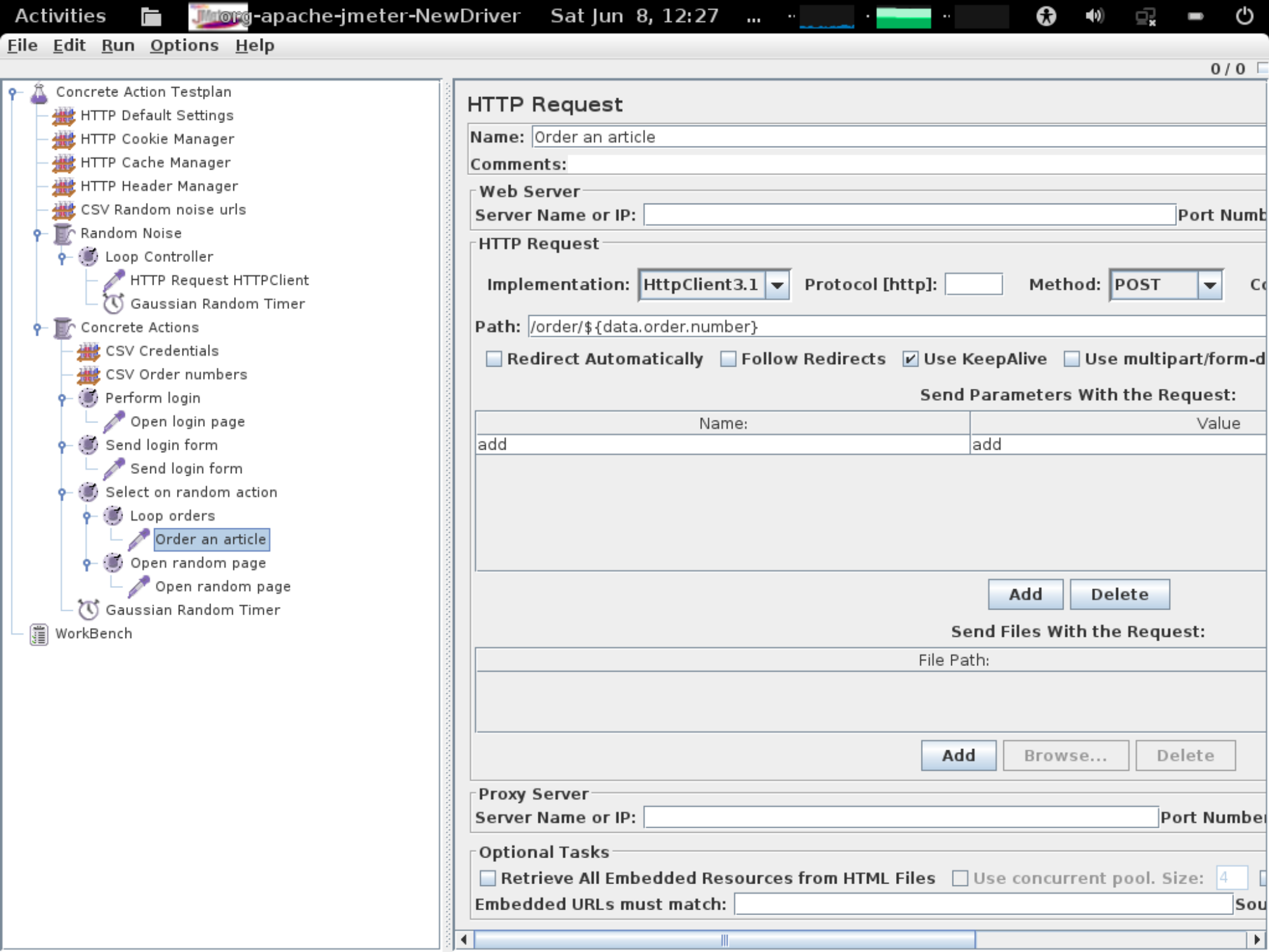

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
HTTP Request
Name: Open login page
Comments:
Web Server
Server Name or IP:
Port Numb
HTTP Request
Implementation: HttpClient3.1
Protocol [http]:
Method: GET
Path: /login.php
Redirect Automatically
Follow Redirects
Use KeepAlive
Use multipart/form-d
Send Parameters With the Request:
Name:
Value
Add
Delete
Send Files With the Request:
File Path:
Add
Browse...
Delete
Proxy Server
Server Name or IP:
Port Number
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size:
4
Embedded URLs must match:

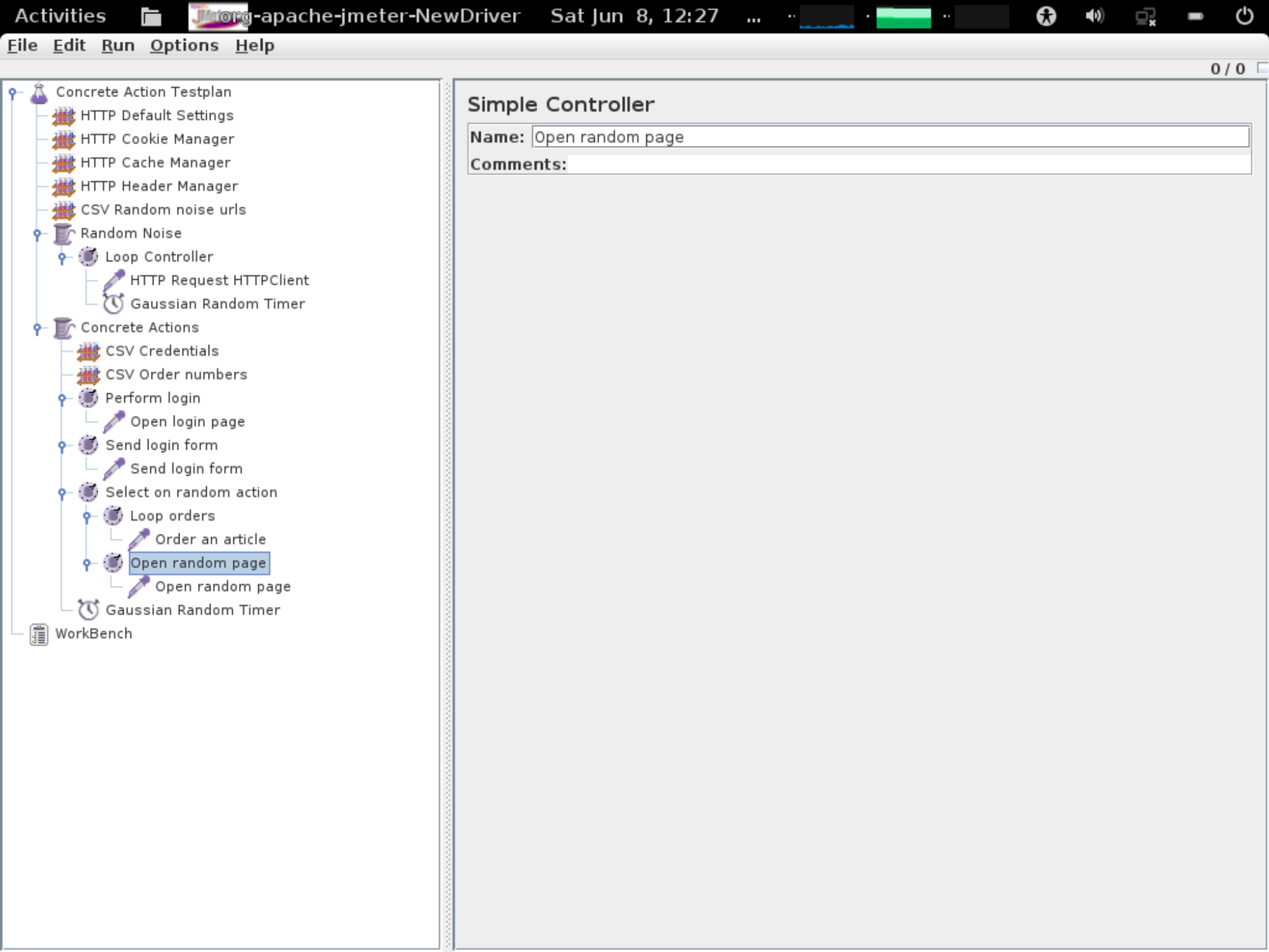

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Simple Controller
Name: Send login form
Comments:

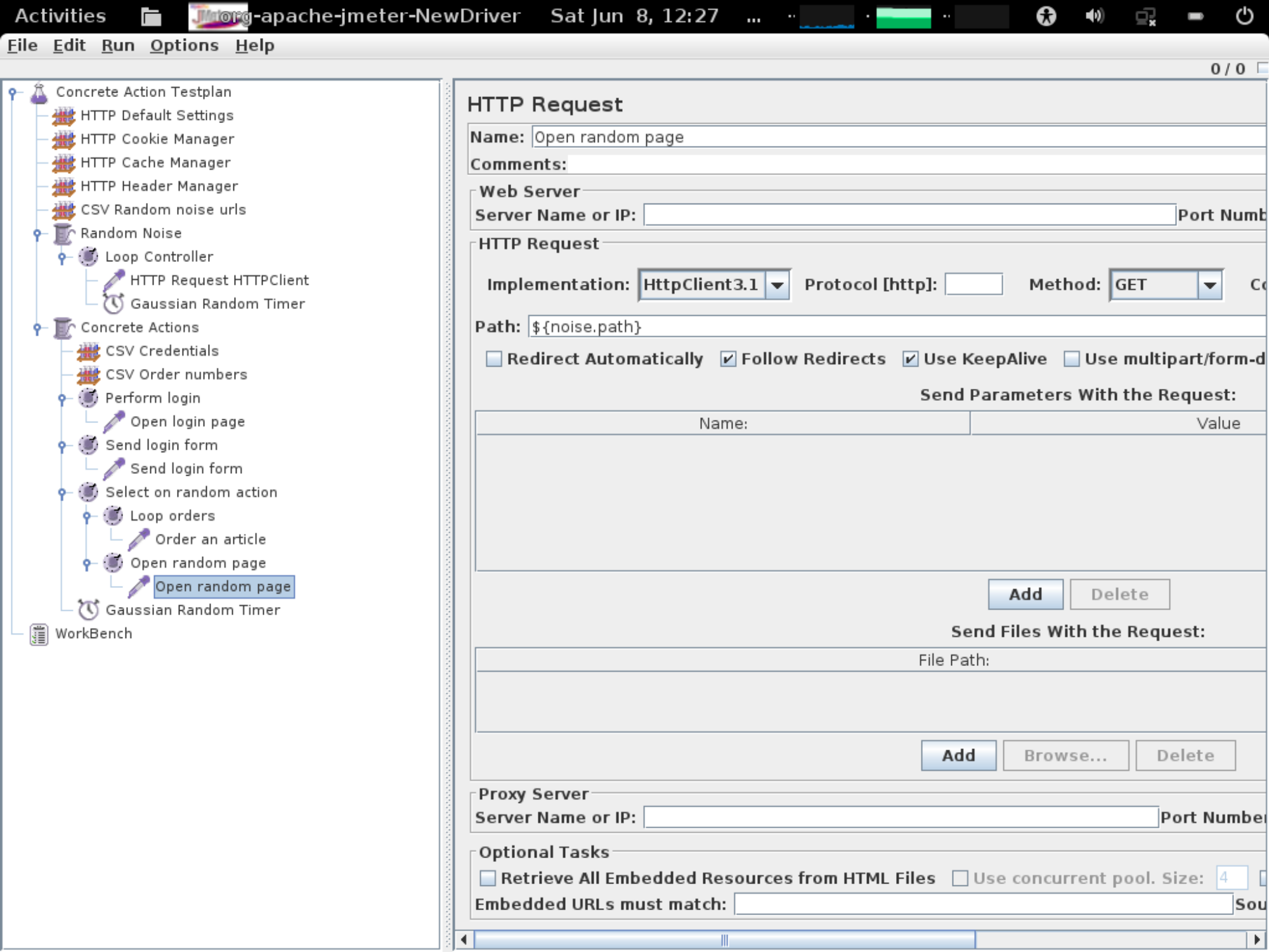

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:26
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
HTTP Request
Name: Send login form
Comments:
Web Server
Server Name or IP:
Port Numb
HTTP Request
Implementation: HttpClient3.1
Protocol [http]:
Method: POST
Path: /login.php
Redirect Automatically
Follow Redirects
Use KeepAlive
Use multipart/form-d
Send Parameters With the Request:
Name:
Value
username
${data.username}
password
${data.password}
submit
Submit
Add
Delete
Send Files With the Request:
File Path:
Add
Browse...
Delete
Proxy Server
Server Name or IP:
Port Number
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size:
4
Embedded URLs must match:

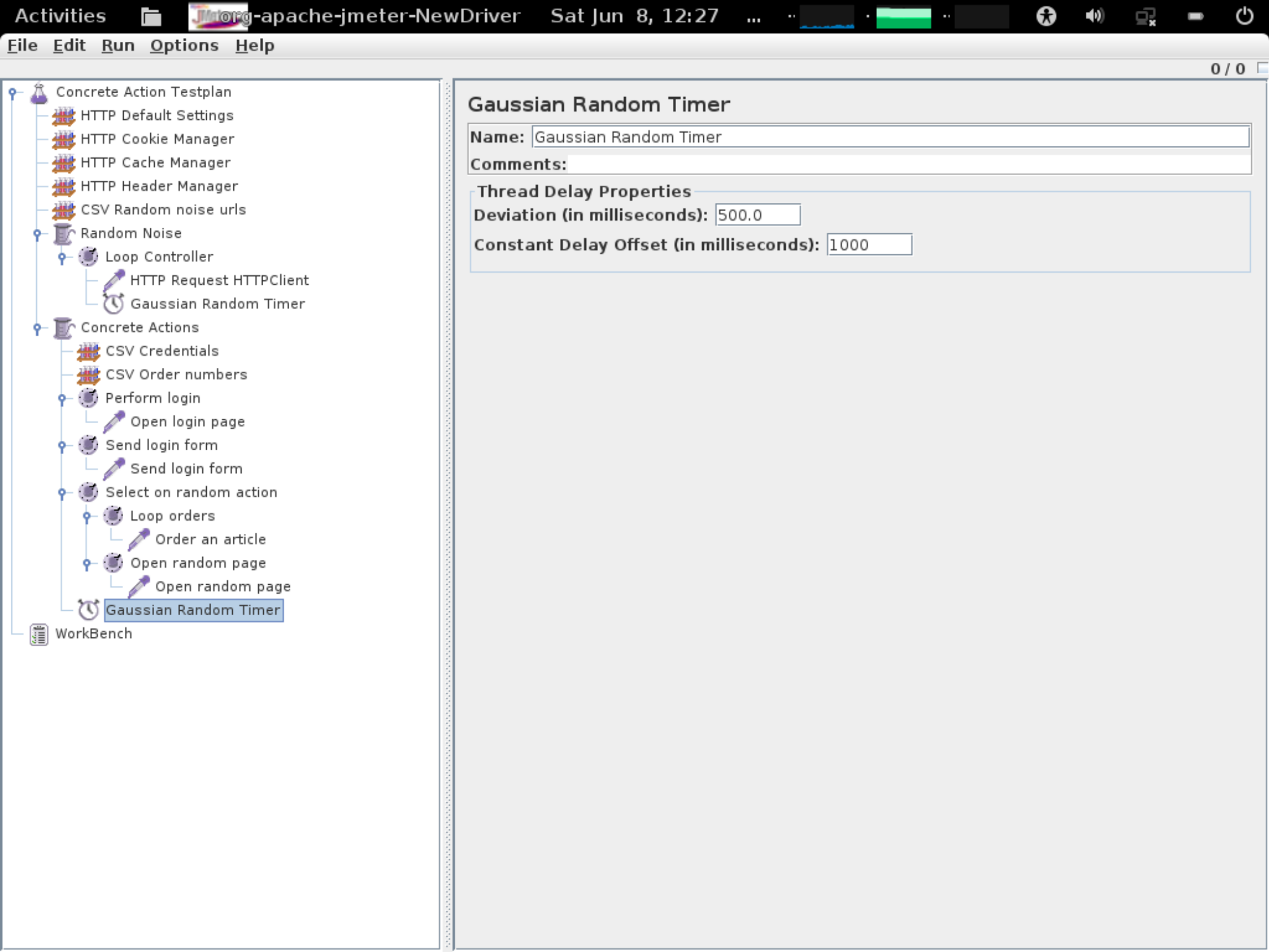

Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Random Controller
Name: Select on random action
Comments:
Ignore sub-controller blocks


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Loop Controller
Name: Loop orders
Comments:
Loop Count: Forever 3


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
HTTP Request
Name: Order an article
Comments:
Web Server
Server Name or IP:
Port Numb
HTTP Request
Implementation: HttpClient3.1
Protocol [http]:
Method: POST
Path: /order/${data.order.number}
Redirect Automatically
Follow Redirects
Use KeepAlive
Use multipart/form-d
Send Parameters With the Request:
Name:
Value
add
add
Add
Delete
Send Files With the Request:
File Path:
Add
Browse...
Delete
Proxy Server
Server Name or IP:
Port Number
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size:
4
Embedded URLs must match:


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Simple Controller
Name: Open random page
Comments:


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
HTTP Request
Name: Open random page
Comments:
Web Server
Server Name or IP:
Port Numb
HTTP Request
Implementation: HttpClient3.1
Protocol [http]:
Method: GET
Path: ${noise.path}
Redirect Automatically
Follow Redirects
Use KeepAlive
Use multipart/form-d
Send Parameters With the Request:
Name:
Value
Add
Delete
Send Files With the Request:
File Path:
Add
Browse...
Delete
Proxy Server
Server Name or IP:
Port Number
Optional Tasks
Retrieve All Embedded Resources from HTML Files
Use concurrent pool. Size:
4
Embedded URLs must match:


Activities
org-apache-jmeter-NewDriver
Sat Jun 8, 12:27
File Edit Run Options Help
0 / 0
Concrete Action Testplan
HTTP Default Settings
HTTP Cookie Manager
HTTP Cache Manager
HTTP Header Manager
CSV Random noise urls
Random Noise
Loop Controller
HTTP Request HTTPClient
Gaussian Random Timer
Concrete Actions
CSV Credentials
CSV Order numbers
Perform login
Open login page
Send login form
Send login form
Select on random action
Loop orders
Order an article
Open random page
Open random page
Gaussian Random Timer
WorkBench
Gaussian Random Timer
Name: Gaussian Random Timer
Comments:
Thread Delay Properties
Deviation (in milliseconds): 500.0
Constant Delay Offset (in milliseconds): 1000

Automation

- Ant JMeter integration
- Automation of your environment
  - Setting up different software versions
  - Testing with different extensions
  - Running different database setups
  - Handling multi-node environments
  - Plain Ant or Puppet, Chef, Vagrant

Apache Ant example

```
26      <target name="build-apc-bytecode-cache-user-cache-file"
27              depends="setup-apc-bytecode-file-cache, -build"
28              description="-> Run with APC opcode cache and file based cache." />
29
30      <target name="build-apc-bytecode-cache-user-cache-apc"
31              depends="setup-apc-bytecode-apc-cache, -build"
32              description="-> Run with APC opcode cache and APC based cache." />


211     <target name="-remote-exec-parallel">
212         <subant target="${target}" inheritall="true">
213             <fileset dir="${project.dir}" includes="server*.xml" />
214         </subant>
215     </target>
216
217     <target name="-remote-exec">
218         <sshexec command="${command}"
219                  username="${ssh.username}"
220                  password="${ssh.password}"
221                  host="${hostname}"
222                  trust="true" />
223     </target>
```

Apache Ant example

```
454     <target name="-restart-host">
455         <antcall target="-remote-exec">
456             <param name="command" value="shutdown -r now" />
457         </antcall>
458
459         <echo taskname="waitfor" message="Wait for ${hostname} to stop ..." />
460         <waitfor maxwait="5" maxwaitunit="minute" checkevery="100">
461             <not>
462                 <http url="http://${hostname}"/>
463             </not>
464         </waitfor>
465
466         <echo taskname="waitfor" message="Wait for ${hostname} is up again ..." />
467         <waitfor maxwait="5" maxwaitunit="minute" checkevery="100">
468             <http url="http://${hostname}"/>
469         </waitfor>
470     </target>
```

Apache Ant example

```
421     <target name="jmeter" depends="-settings-init,-start-jmeter" />
422
423     <target name="-start-jmeter">
424         <antcall target="-start-jmeter-before-hook" />
425
426         <jmeter jmeterhome="${local.jmeter.home.dir}"
427                 resultlog="${local.jmeter.log.file}"
428                 testplan="${local.jmeter.test.dir}/${jmeter.file}">
429
430             <property name="jmeter.data.dir" value="${local.project.data.dir}" />
431             <property name="jmeter.rampup.time" value="${jmeter.rampup.time}" />
432             <property name="jmeter.execution.time" value="${jmeter.execution.time}" />
433         </jmeter>
434
435         <antcall target="-start-jmeter-after-hook" />
436     </target>
```

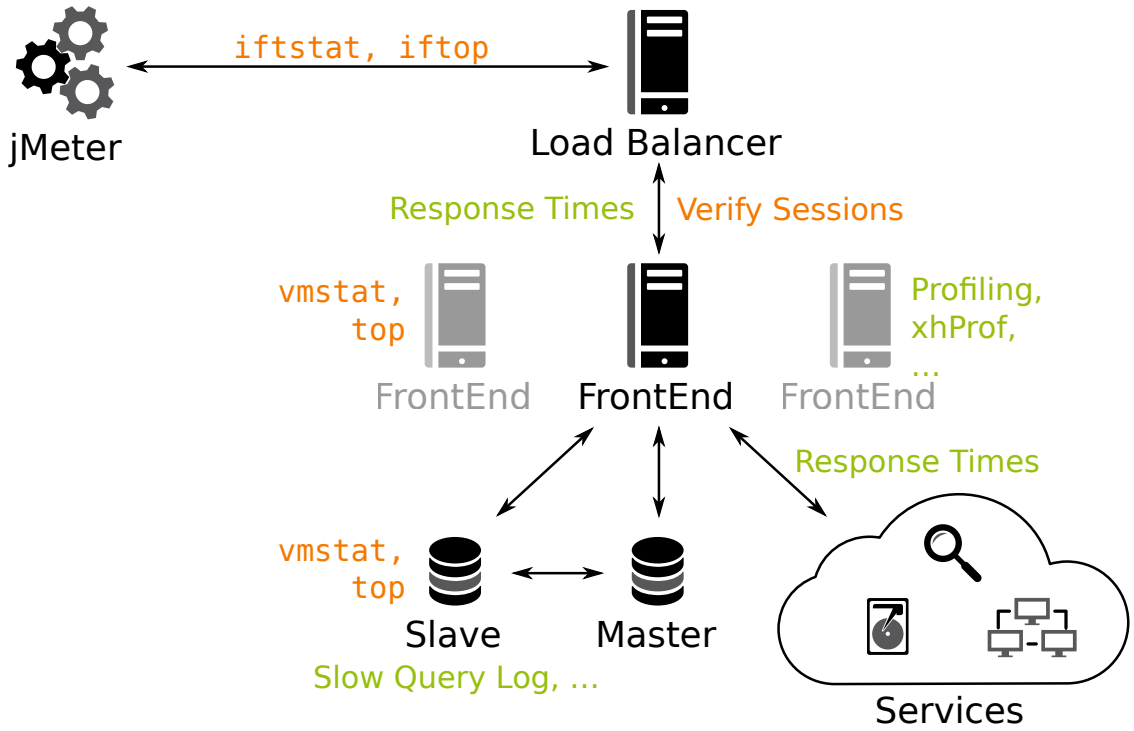
Apache Ant example

```
351     <target name="-backup-data-from-host">
352         <scp remotefile="${ssh.username}:${ssh.password}@${hostname}:'${remote.webserver.
                error.log}'"
353              localtofile="${local.builddir}/${hostname}-error.log"
354              trust="true" />
355
356         <scp remotefile="${ssh.username}:${ssh.password}@${hostname}:'${remote.webserver.
                access.log}'"
357              localtofile="${local.builddir}/${hostname}-access.log"
358              trust="true" />
359
360         <scp remotefile="${ssh.username}:${ssh.password}@${hostname}:'${remote.php.error.
                log}'"
361              localtofile="${local.builddir}/${hostname}-php_errors.log"
362              trust="true" />
363
364         <scp remotefile="${ssh.username}:${ssh.password}@${hostname}:'${remote.basedir}/
                scripts/load.log'"
365              localtofile="${local.builddir}/${hostname}-load.log"
366              trust="true" />
367
368     </target>
```

Test Setup

What do we actually test?

Test Setup



Hardware

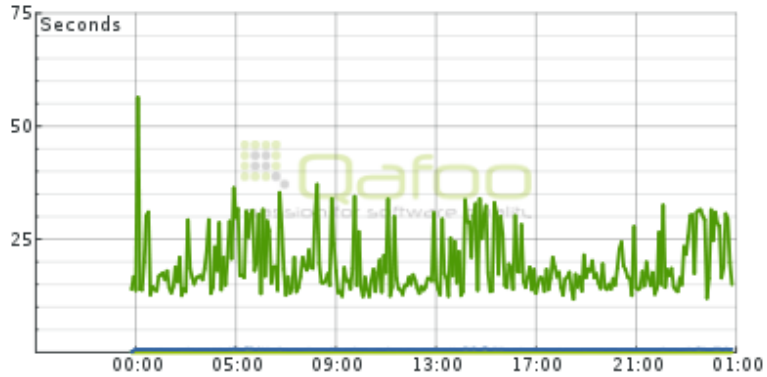
- Test in a realistic environment
  - If your software runs in the cloud test against virtual environment
  - If you use real hardware, also test against real hardware
- JMeter might have serious hardware requirements
  - Use real hardware
  - Use the biggest VM available
  - Ensure that not the JMeter hardware is the bottleneck
- Be sure that the network is not the bottleneck
  - See `ifstat`, `iftop`
- Measure several system metrics
  - See `vmstat`, `top`

Frontend (Request & Response)

Requests per second

Average

Response time



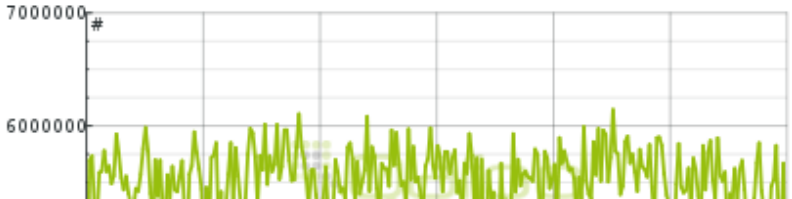
Minimum Average Maximum

Request failures

Failures

Database

Context switches

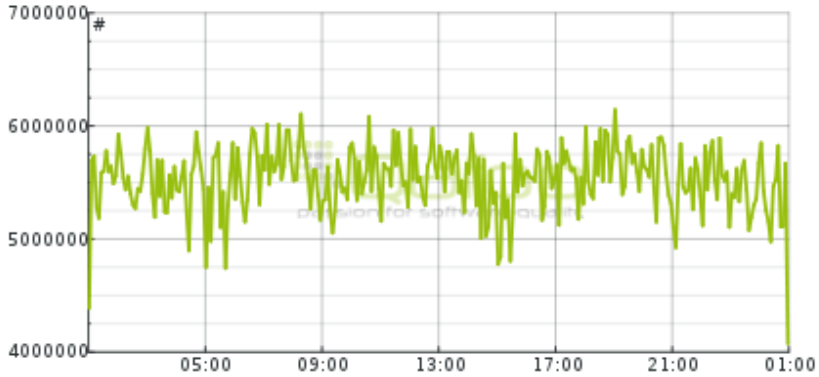


Frontend

Context switches

Database

Context switches



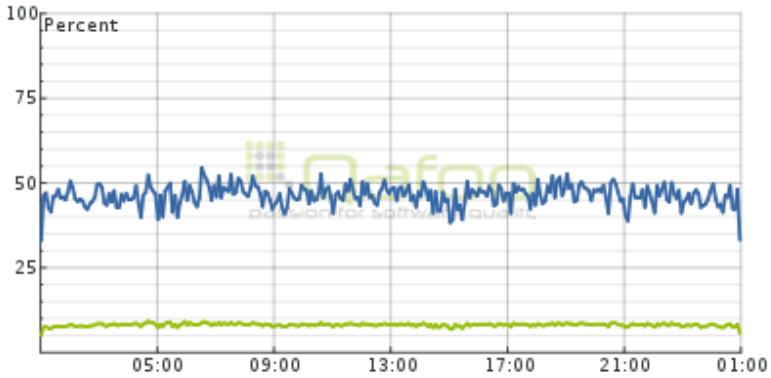
CPU context switches

Frontend

Context switches

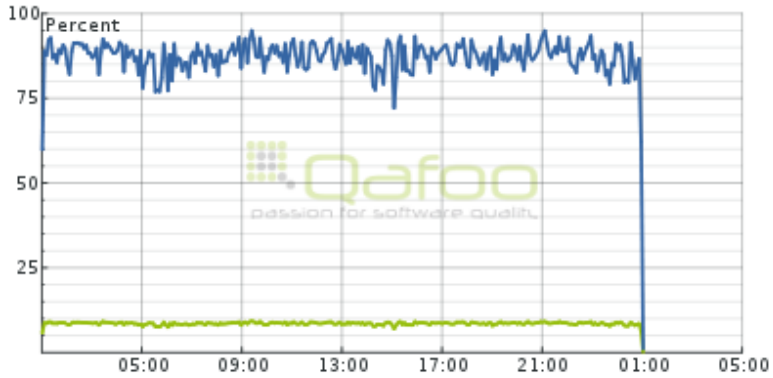
CPU context switches

CPU usage



System User

CPU usage



System User

Database

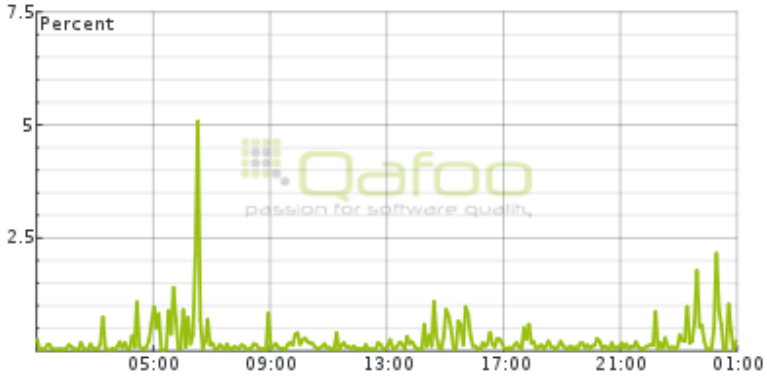
IO Wait

Frontend

IO Wait

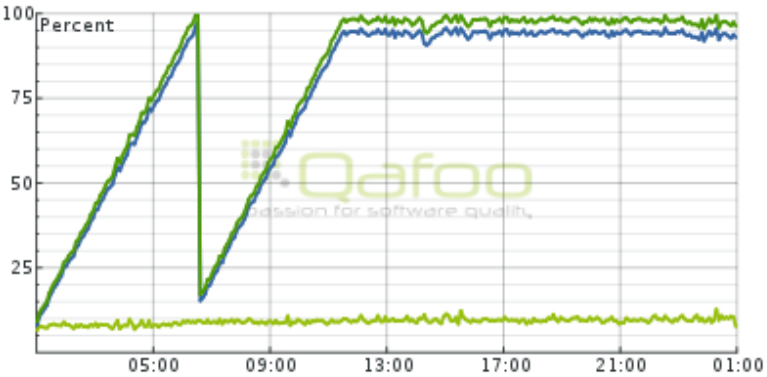
Database

IO Wait



IO Wait

Memory usage



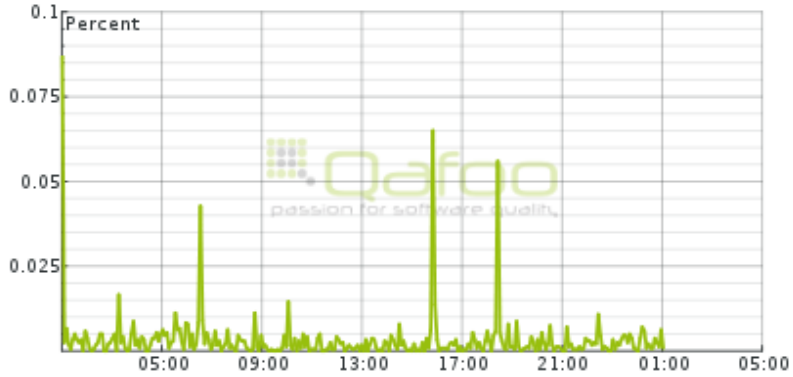
Used Memory Shared Cached

Database

Pages paged

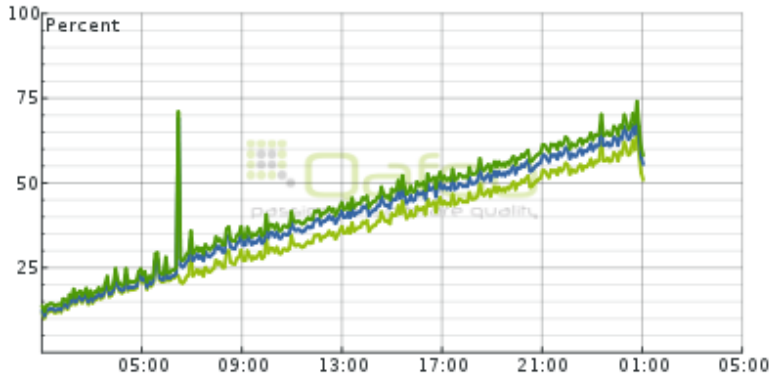
Frontend

IO Wait



IO Wait

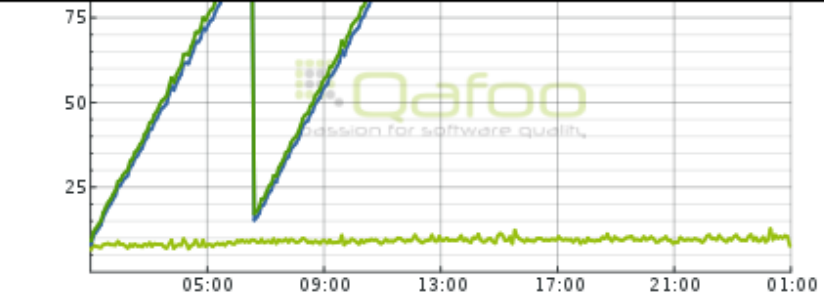
Memory usage



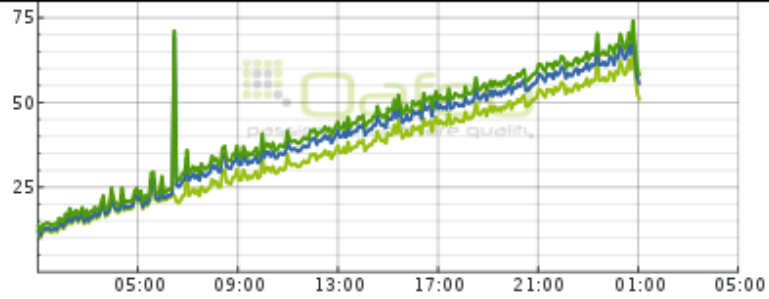
Used Memory Shared Cached

Frontend

Pages paged



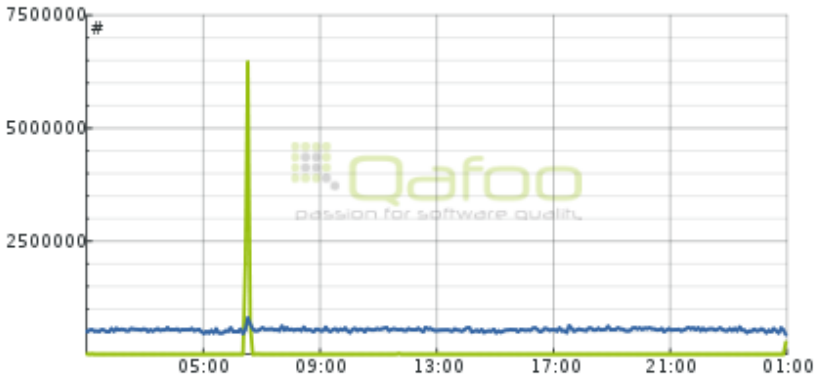
Used Memory Shared Cached



Used Memory Shared Cached

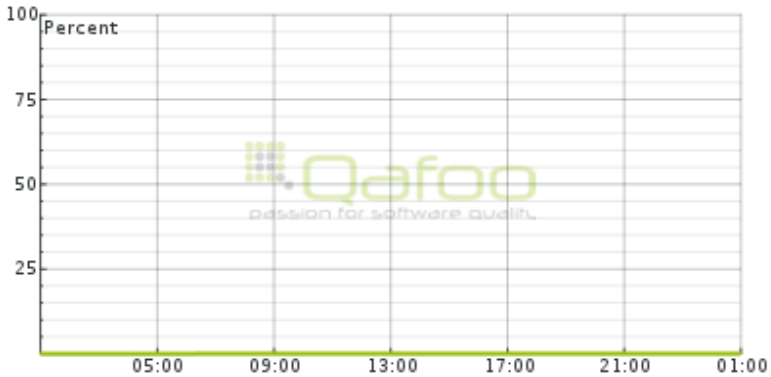
Database

Pages paged



Into memory Out of memory

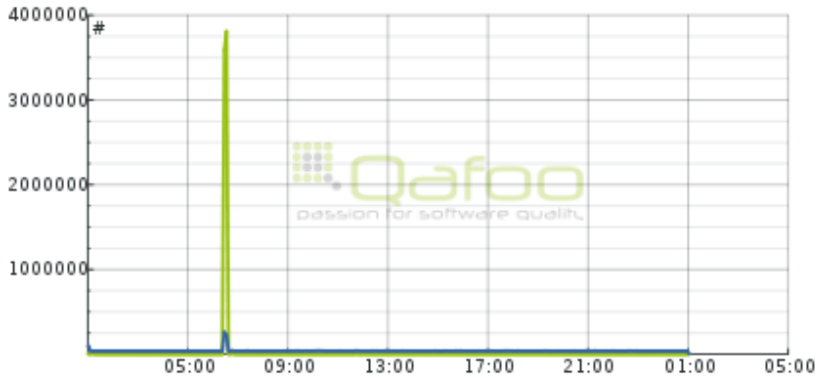
Swap usage



Used Swap

Frontend

Pages paged



Into memory Out of memory

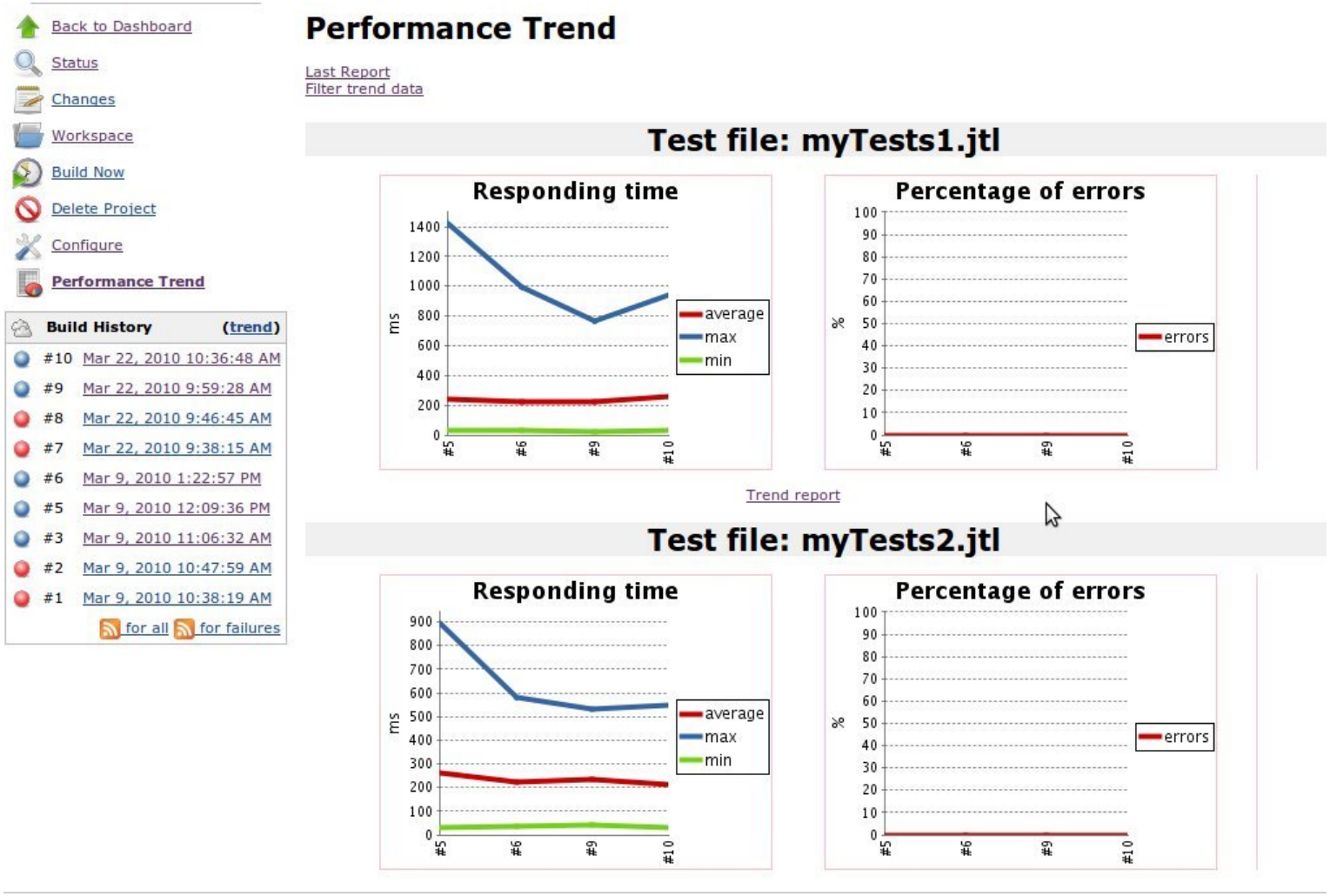
Swap usage

Used Swap

Continuous Performance

- Plugins available for:
  - Jenkins
  - Sonar
- Maintaining all those servers can be expensive

Continuous testing with Jenkins



Questions

- Common questions:
  - Can we survive christmas?
  - What is the maximum we can accomplish with the current setup?
  - Does (APC|XCache|MemCache) really help us?

Getting realistic settings

- Ask for real access logs:
  - Extract exact request model from those
- Your customer usually only knows very broad values, like:
  - 1.000.000 PIs per month
  - 30.000 sold articles per month
  - 45.000 registrations per month
- Very seldom, that you get more:
  - Ask for access statistics before Christmas and the ratio compared with regular months
  - Ask for hours with the highest conversion rates
  - Maybe get the aggregated access logs from existing similar shops

Example calculation

- Customer provided values, for a classic webshop:
  - 1.000.000 PIs per month
  - 30.000 sold articles per month
  - 45.000 registrations per month
- Per day: $1.000.000/26 = 38.500$ (non-business)
- Per hour: $38.500/12 = 3.200$ (national shop)
- Peak hour: $3.200 * 8 = 25.500$ (18:00 to 19:00)
- Per second: $25.500/3600 = 7 PI/s$
- Add Christmas / Easter bonus
- Add launch bonus
- So . . . **50 PI/s** should be safe?
  - Spare resources for scaling are always a business decision
  - Provide with trade-off: Costs vs. downtime / slowness
  - Fail gracefully

Verification

- Do your requests actually model customer requirements?
  - Compare generated accesss logs with real access logs
  - Compare user registrations / checkouts per hour with requested values

Outline

Motivation

Conclusion

Conclusion

- Plan your test scenario
- Use realisitic thresholds
- Choose the right tool

with care

THANK YOU

Rent a quality expert
qafoo.com