Performance testing with JMeter

Ilídio Oliveira | ico@ua.pt

TQS | 2022/05/31

universidade de aveiro
departamento de eletrónica, telecomunicações e informática
deti

ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — **System and software quality models**

https://blog.codacy.com/iso-25010-software-quality-model/

Functional quality characteristics/factors are just a part of the story...

Holistic view: need to consider quality characteristics in the product conception and development

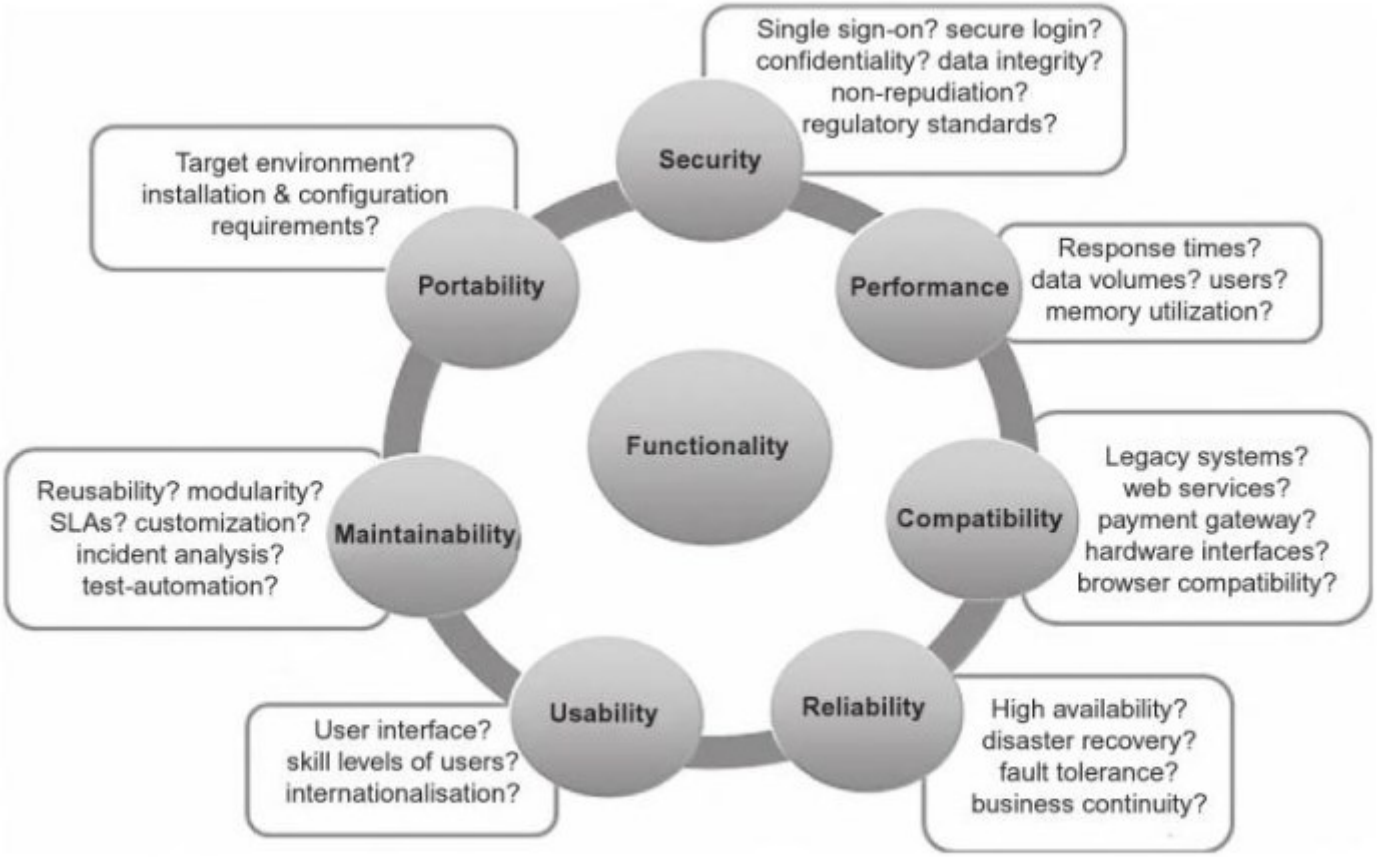


FIGURE 6.2

Holistic View of Product Quality.

ISO/IEC 25010:2011 - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models

Performance testing

Tools and metrics

Why performance testing?

Non function requirements

- Performance, latency
- How far can you load the system ensuring error-free behavior?

How?

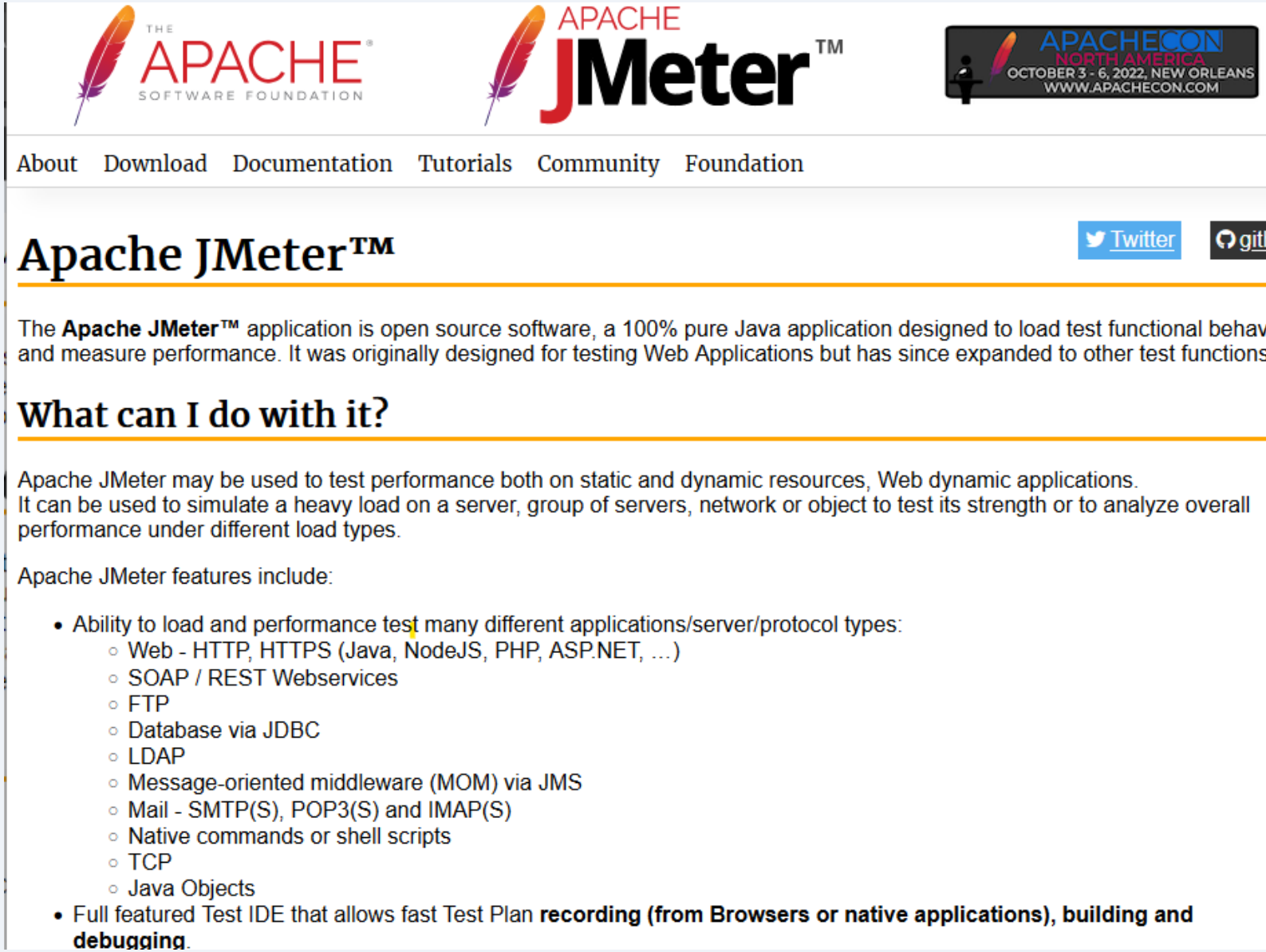
- Synthetic load generation
    - simulate user actions (functional)
- Measurement & reporting instrumentations

Mind some differences

Performance testing	Stress testing	Load testing
Testing to determine the performance of a software product.	A type of performance testing to evaluate a system or component **at or beyond the limits of its anticipated or specified workloads**, or with reduced availability of resources such as access to memory or servers.	A type of performance testing to evaluate the **behavior of a component or system with increasing load**, e.g., numbers of parallel users and/or numbers of transactions, to determine what load can be handled by the component or system.

https://istqb-glossary.page/


APACHE
JMeter™


THE APACHE SOFTWARE FOUNDATION
APACHE JMeter™
APACHECON NORTH AMERICA
OCTOBER 3 - 6, 2022, NEW ORLEANS
WWW.APACHECON.COM
About Download Documentation Tutorials Community Foundation
Apache JMeter™
Twitter
github
The Apache JMeter™ application is open source software, a 100% pure Java application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.
What can I do with it?
Apache JMeter may be used to test performance both on static and dynamic resources, Web dynamic applications.
It can be used to simulate a heavy load on a server, group of servers, network or object to test its strength or to analyze overall performance under different load types.
Apache JMeter features include:
Ability to load and performance test many different applications/server/protocol types:
Web - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, ...)
SOAP / REST Webservices
FTP
Database via JDBC
LDAP
Message-oriented middleware (MOM) via JMS
Mail - SMTP(S), POP3(S) and IMAP(S)
Native commands or shell scripts
TCP
Java Objects
Full featured Test IDE that allows fast Test Plan recording (from Browsers or native applications), building and debugging.

Browser vs JMeter as HTTP clients

Browser lifecycle

User performs an action

Browser sends an HTTP request

Server processes the request and responses

Browser parses the response and executes scripts

JMeter behaviour

~~User performs an action~~

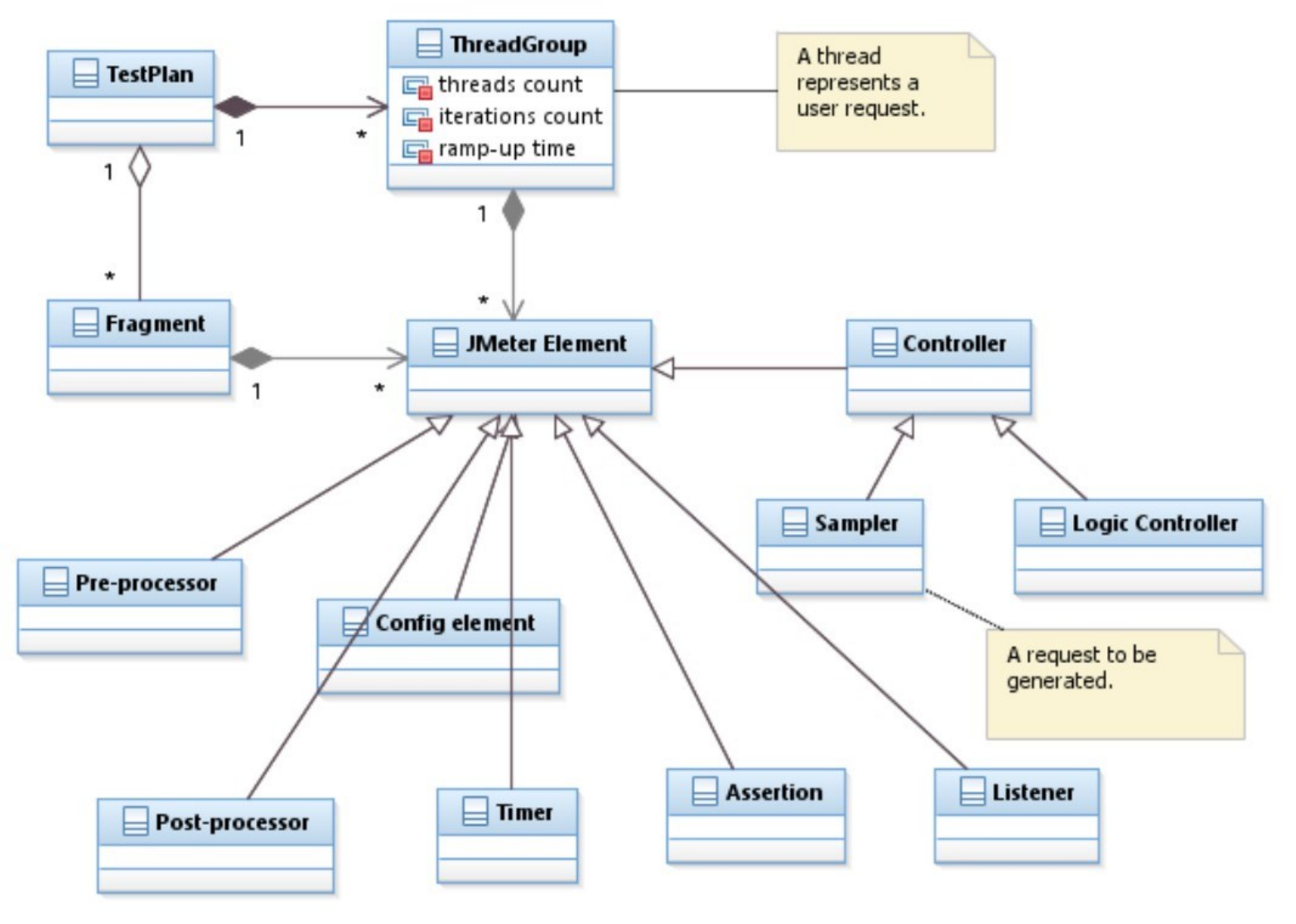
JMeter sends an HTTP request

Server processes the request and responses

JMeter parses the response ~~and executes scripts~~

Repeat

JMeter elements



JMeter elements

Element	Semantics
Test Plan	A JMeter script.
Thread Group	Simulates a group of users (request ~ user)
Sampler	An action that causes a request.
Config	Additional configuration.
Timer	Add a predefined delay.
Assertions	Error checking to evaluate responses
Pre-processor	Modify the request before it is issued
Post-processor	Modify the response
Logic controller	Control node (alternatives, looping,…)

Some useful Samplers

HTTP Request

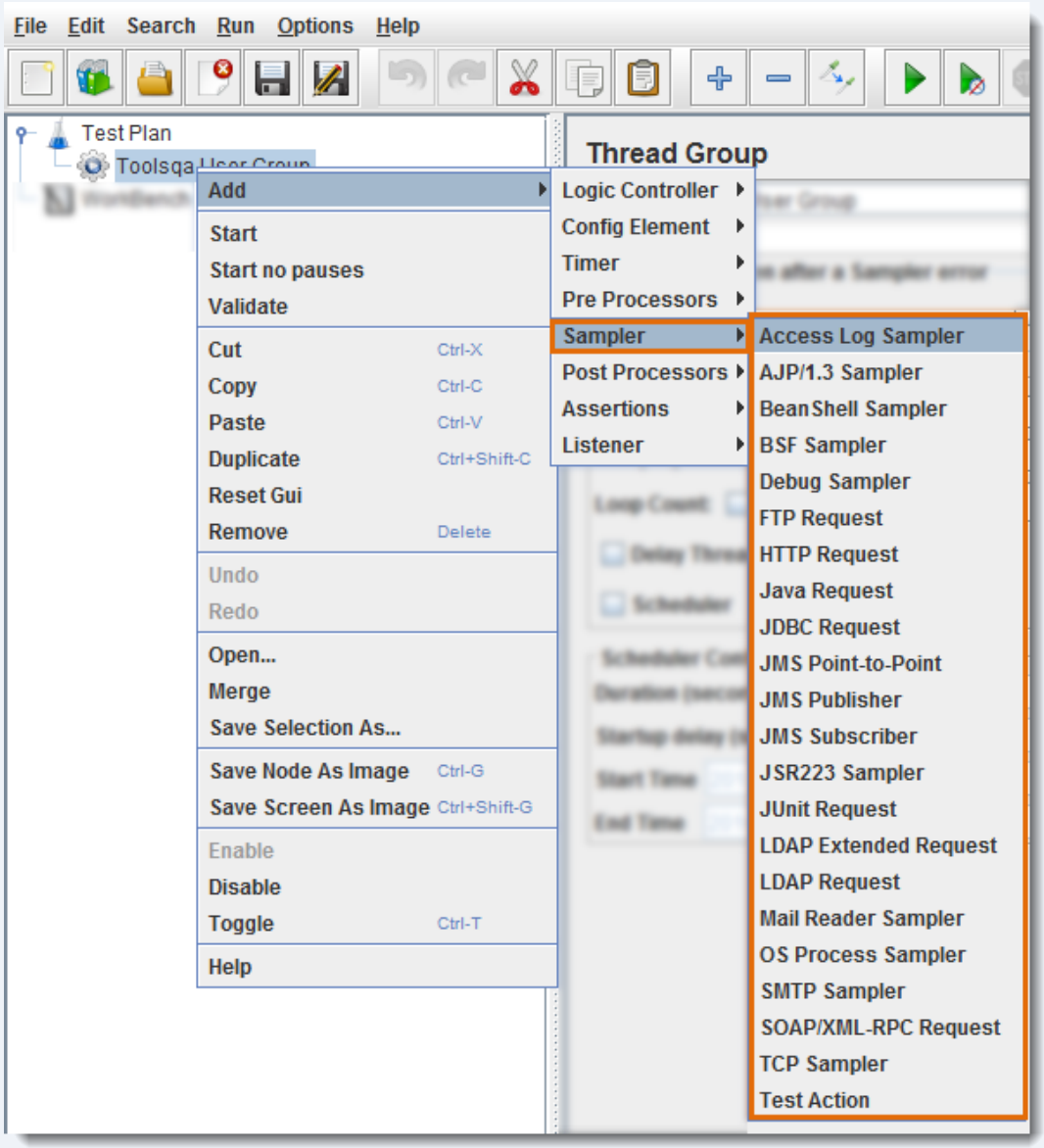
FTP Request

JDBC Request

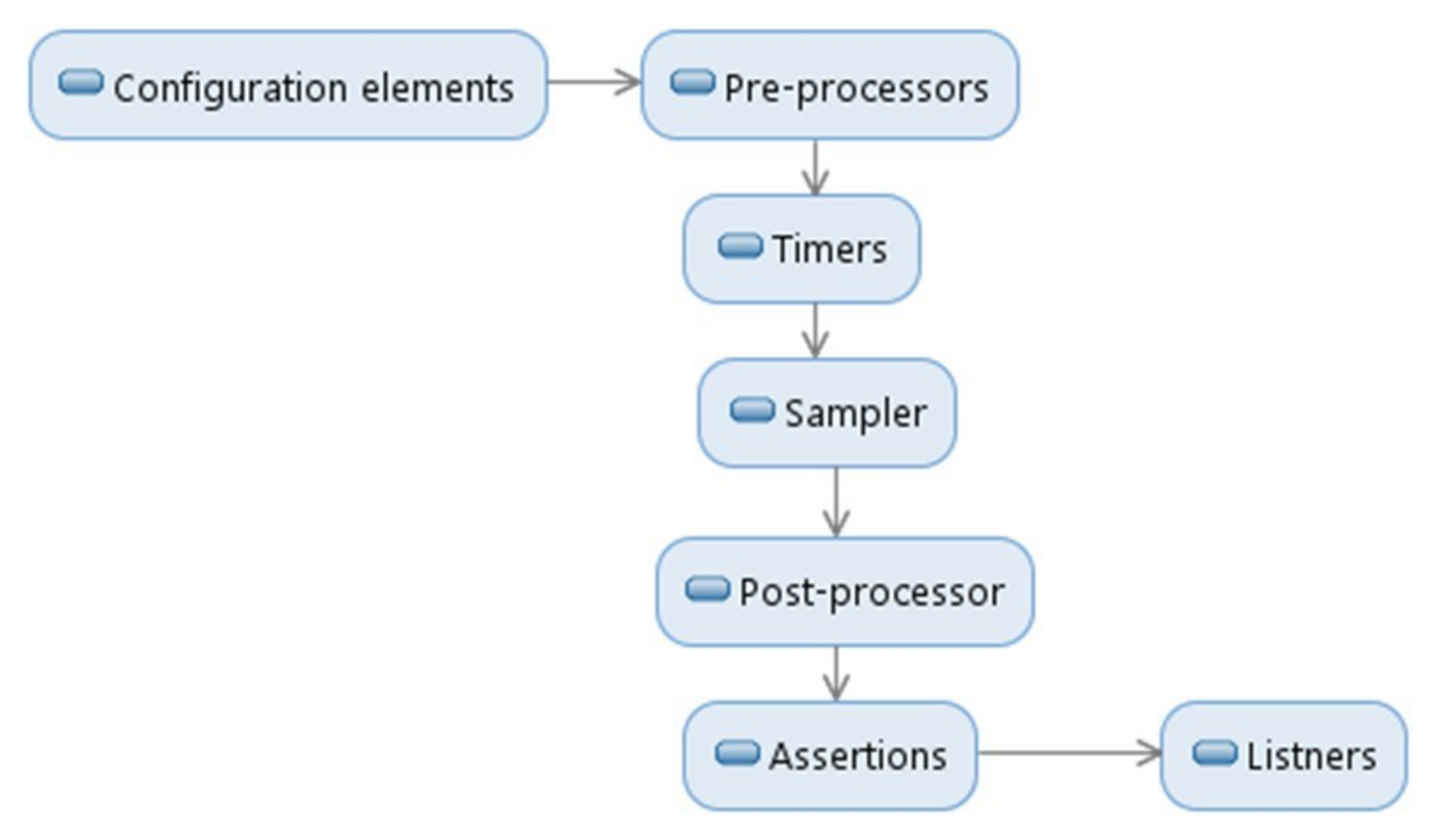
Java Request

SOAP/XML Request

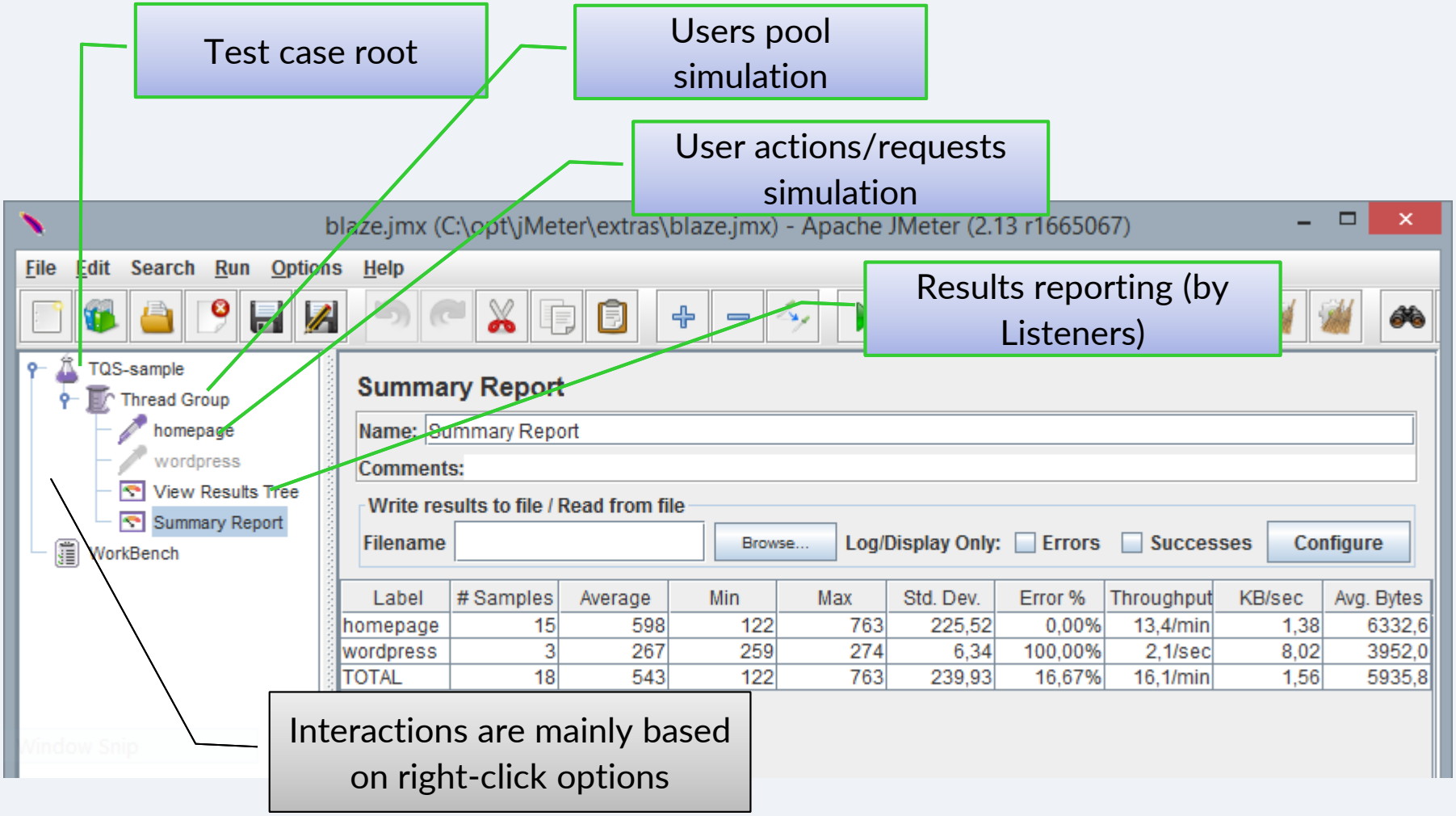
RPC Requests



Execution order of test elements



JMeter IDE


Test case root
Users pool simulation
User actions/requests simulation
Results reporting (by Listeners)
blaze.jmx (C:\opt\jMeter\extras\blaze.jmx) - Apache JMeter (2.13 r1665067)
File Edit Search Run Options Help
TQS-sample
Thread Group
homepage
wordpress
View Results Tree
Summary Report
WorkBench
Summary Report
Name: Summary Report
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Label # Samples Average Min Max Std. Dev. Error % Throughput KB/sec Avg. Bytes
homepage 15 598 122 763 225,52 0,00% 13,4/min 1,38 6332,6
wordpress 3 267 259 274 6,34 100,00% 2,1/sec 8,02 3952,0
TOTAL 18 543 122 763 239,93 16,67% 16,1/min 1,56 5935,8
Interactions are mainly based on right-click options

Basic http request

New test plan, with descriptive name

Simulate

- 4 users
- 1 visit
- to UA's home page (http request)

Display the results

- in Results Tree view
- in Summary report

http://www.tutorialspoint.com/jmeter/jmeter_web_test_plan.htm

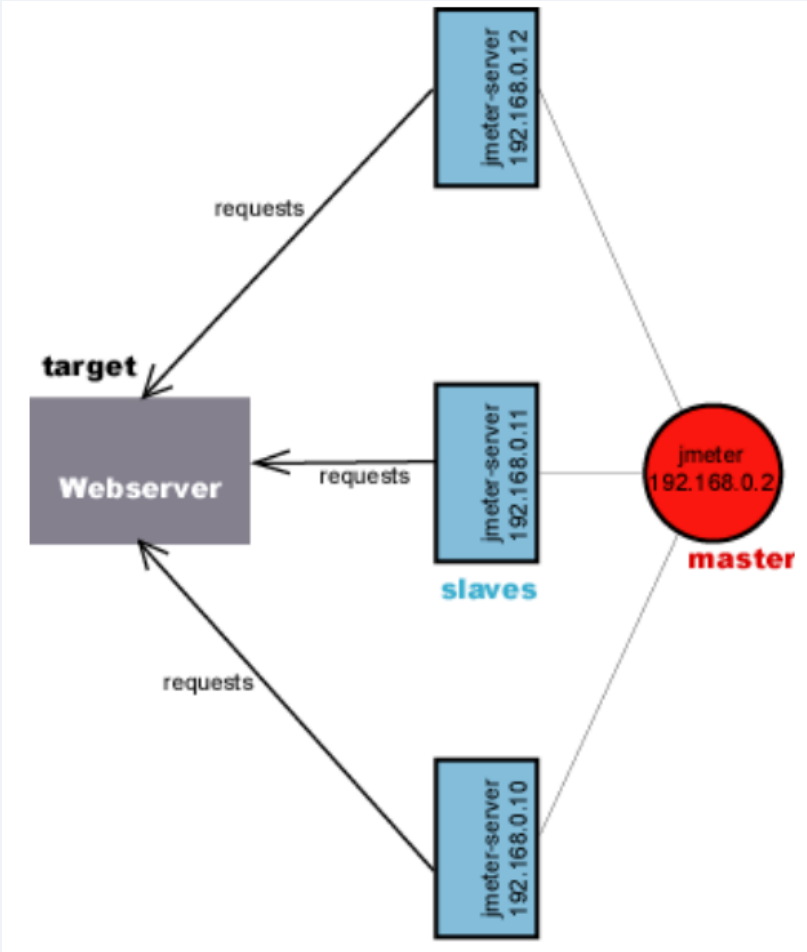
Test a REST endpoint

- Add http request
- Provide details for GET
- Change the http-header to send json

http://www.testautomationguru.com/how-to-test-rest-api-using-jmeter/

Distributed performance testing

Master/slave architecture



JMeter practices

Use multiple instances of JMeter if using many threads

JMeter can be run without GUI

- jmeter -n -t test.jmx -l test.jtl

Use as few Listeners as possible

- deactivate/activate

Prefer CSV over XML to save results

If the machine running the test case is resource-exhausted (e.g.: CPU 100%), the results will not be reliable!

See also:

http://www.testautomationguru.com/jmeter-tips-tricks-for-beginners/

Remember GUI mode is for Script creation and debugging, not for load testing

Resources and references

Apache JMeter site

- includes demo tutorials

The JMeter Manual

JMeter and WebDriver integration

Selected best practices