



Quick answers to common problems

JMeter Cookbook

70 insightful and practical recipes to help you successfully use
Apache JMeter

Bayo Erinle

[PACKT] open source 
PUBLISHING community experience distilled

JMeter Cookbook

70 insightful and practical recipes to help you successfully use Apache JMeter

Bayo Erinle

[PACKT] open source 
PUBLISHING community experience distilled

BIRMINGHAM - MUMBAI

JMeter Cookbook

Copyright © 2014 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: October 2014

Production reference: 1221014

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-78398-828-0

www.packtpub.com

Cover image by Artie Ng (artherng@yahoo.com.au)

Credits

Author

Bayo Erinle

Project Coordinator

Harshal Ved

Reviewers

Yoann Ciabaud

Andrey Pokhilko

Shantonu Sarker

Federico Diaz Valpuesta

Žmicier Žaležničenka

Proofreaders

Simran Bhogal

Ameesha Green

Indexers

Hemangini Bari

Tejal Soni

Commissioning Editor

Usha Iyer

Production Coordinator

Shantanu N. Zagade

Acquisition Editor

Owen Roberts

Cover Work

Shantanu N. Zagade

Content Development Editor

Dayan Hyames

Technical Editor

Akash Rajiv Sharma

Copy Editors

Maria Gould

Paul Hindle

Deepa Nambiar

About the Author

Bayo Erinle is an author and senior software engineer with over 11 years of experience in designing, developing, testing, and architecting software. He has worked in various spectrums of the IT field, including government, commercial, finance, and health care. As a result, he has been involved in the planning, development, implementation, integration, and testing of numerous applications, including multi-tier, standalone, distributed, and Cloud-based applications. He is passionate about programming, performance, scalability, and all technology-related things. He is always intrigued by new technology and enjoys learning new things.

He currently resides in Maryland, USA, and when he is not hacking away at some new technology, he enjoys spending time with his wife, Nimota, and their three children, Mayowa, Durotimi, and Fisayo.

He has also authored *Performance Testing with JMeter 2.9*, Packt Publishing.

About the Reviewers

Yoann Ciabaud is an architect/developer with strong application architecture skills in Java/JEE, integration, and database environments. While mainly focused on application architecture, design, and development, Yoann also has extensive skills in infrastructure (networks, server, and operating systems) and the design, build, and configuration of development, test, and production environments.

He is currently working for an electronic financial transaction company called Monext in France. Including other project and line roles, Yoann has worked in large (more than 50) and small (sole team member) teams on complex distributed architectures based on a wide range of technologies.

I would like to thank my lovely wife, who lets me spend time working on reviewing a book instead of having more family time together. I want to assure her, I will catch up soon.

Andrey Pokhilko is well known as the founder of JMeter-Plugins.org, an open source project that has been enhancing the JMeter user experience since 2009. While working on this project, he also led Load Testing Group at search giant Yandex (NASDAQ: YNDX), a position that provided him with invaluable experience in high-load performance testing. At Yandex, he developed the updated version of Yandex.Tank, an open source load testing tool.

Keen to apply his ideas to a wider range of load testing problems, Andrey left Yandex in 2014 to focus on building Loadosophia.org, a start-up that provides load test results analysis as a service.

Shantonu Sarker is a freelance software test automation engineer, currently working for oDesk/E lance/Staff.com. As a consultant, he works as a trainer in BITM (an institute of BASIS) and as an automation tester for several reputed companies in Bangladesh. He has his own start-up company named QualitySofts.

He has attended three professional training courses provided by BASIS (Bangladesh Association for Software and Information Services) on OOP, OOAD, and software design and architecture. He also has training in Agile development and testing (Kanban and Scrum) and security testing. He is also learning to speak Japanese and has passed the JLPT-L4 exam.

He has previously reviewed *Performance Testing with JMeter 2.9*, Packt Publishing.

I would like to thank my Guru, Mahajatok. Without his guidance, I would not be what I am today.

Federico Diaz Valpuesta is a Linux systems administrator currently living in Seville, Spain. He has extensive experience in the administration of local area networks and Linux servers.

He has been using Linux since 1999, starting with RedHat 7.2, and currently uses Debian. He has learned to use all the various "flavors" of Linux when he needs to solve a problem, so he usually installs Debian for web servers, CentOS for clustering, and Zentyal for small LANs, user management, and file/printer sharing.

Right now, he is working as a freelance web developer, deploying web apps in AWS, DigitalOcean, and Hetzner. He lends a hand in virtualization and helps everyone who wants to get started with Linux.

In addition to trying new things on his computer for long hours, he also likes mountain biking and watching TV. You can follow his activities on his blog or Twitter.

Žmici er Žaležničenka holds a Master of Science degree in Computer Science from Delft University of Technology, the Netherlands. He is currently working as a software developer at TomTom in Amsterdam. His areas of interest include search engines, application monitoring software, and performance optimization of large-scale web applications. He has worked closely with JMeter, both during his studies and day-to-day development activities.

I'd like to thank Packt Publishing for giving me the opportunity to edit this publication.

www.PacktPub.com

Support files, eBooks, discount offers, and more

You might want to visit www.PacktPub.com for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at service@packtpub.com for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- ▶ Fully searchable across every book published by Packt
- ▶ Copy and paste, print and bookmark content
- ▶ On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at www.PacktPub.com, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Table of Contents

Preface	1
Chapter 1: JMeter Fundamentals	7
Introduction	7
Executing a test script	8
Recording a script via HTTP(S) Test Script Recorder	12
Recording scripts via the Chrome browser extension	17
Converting HTTP web archives (HAR) to JMeter test plans	19
Viewing and analyzing test results	22
Feeding data into a script	24
Using timers	26
Managing HTTP user sessions	27
Testing Single Page Applications (SPAs)	28
Testing AJAX-centric applications	30
Chapter 2: Handling Responses	33
Introduction	33
Using Regular Expression Extractor	34
Using Regular Expression Tester	36
Using CSS/jQuery Extractor	38
Using XPath Extractor	40
Dealing with file downloads	42
Handling XML responses	44
Handling JSON responses	47
Handling HTML responses	50
Using Response Assertion	54
Using Duration Assertion	56
Uploading files with your scripts	58

Chapter 3: Building Robust Test Plans with Controllers	61
Introduction	61
Using Transaction Controller in test plans	62
Using Loop Controller in test plans	65
Leveraging ForEach Controller in test plans	69
Using Interleave and Random Controller in test plans	72
Using Runtime Controller in test plans	77
Chapter 4: Testing Services	81
Introduction	81
Testing REST web services	82
Testing SOAP web services	85
Testing FTP services	90
Testing relational databases	94
Testing NoSQL databases	98
Testing JMS services	101
Chapter 5: Diving into Distributed Testing	105
Introduction	105
Testing applications with JMeter's master-slave setup	106
Testing internal applications using JMeter and Vagrant	109
Testing external facing applications using JMeter, Vagrant, and AWS	113
Testing external facing applications using Flood.IO	118
Testing external facing applications using BlazeMeter	122
Chapter 6: Extending JMeter	125
Introduction	125
Using REST Sampler	126
Using Ultimate Thread Group	130
Using Throughput Shaping Timer	132
Using Console Status Logger	136
Using Dummy Sampler	138
Developing custom JMeter plugins	140
Testing WebSocket-enabled applications	144
Chapter 7: Building, Debugging, and Analyzing the Results of Test Plans	149
Introduction	149
Using the View Results Tree listener	150
Using the Aggregate Report listener	152
Debugging with Debug Sampler	153
Using Constant Throughput Timer	155
Using the JSR223 postprocessor	156

Analyzing Response Times Over Time	159
Analyzing transactions per second	161
Using User Defined Variables (UDV)	163
Chapter 8: Beyond the Basics	165
Introduction	165
Continuous Integration with JMeter	166
Testing with different bandwidths	173
Using the HTTP Cache Manager component	176
Using script languages within test plans	178
Writing Test scripts through Ruby DSL	181
Understanding JMeter properties	183
Monitoring servers while executing tests (using VisualVM)	186
Monitoring servers while executing tests (using YourKit Profiler)	189
Monitoring servers while executing tests (using New Relic)	192
Performance tips to scale JMeter	197
Appendix: Installing the Supporting Software Needed for This Book	201
Introduction	201
Installing JMeter	202
Installing Java Development Kit (JDK)	202
Installing JMeter plugins	203
Installing Vagrant	203
Installing VirtualBox	204
Installing Maven	204
Installing Git	205
Obtaining AWS keys for EC2	205
Index	207

Preface

In today's ever-growing IT sector, users are growing increasingly impatient when faced with slow and unresponsive applications. Slow page load times and sluggish services could mean unsatisfactory customer experiences leading to fewer customer visits, which in the end translates to smaller profit margins for businesses. As such, it becomes increasingly important to a business's success to have fast, reliable, and responsive systems that give users a superb experience, foster growth, and increase revenue.

Performance testing is a type of testing intended to determine the responsiveness, reliability, throughput, interoperability, and scalability of a system and/or application under a given workload. It is critical and essential to the success of any software product launch and its maintenance. It also plays an integral part in scaling an application out to support a wider user base.

Apache JMeter is a free, open source, cross-platform performance testing tool that has been around since the late 90s. It is mature, robust, portable, and highly extensible. It has a large user base and offers lots of plugins to aid testing.

This book is a practical, hands-on guide that focuses on how to leverage Apache JMeter to meet your testing needs. With over 50 practical and carefully selected recipes, it will guide you through building robust and maintainable test scripts, Cloud testing, developing custom JMeter plugins, integrating JMeter into continuous delivery workflows, and much more. You will find a lot of useful knowledge here to apply to your current or future testing engagements. Whether you are a developer or tester, this book is sure to have an impact on you and provide you with valuable knowledge to help you achieve success in your future testing endeavors.

What this book covers

Chapter 1, JMeter Fundamentals, covers fundamental and intermediate skills to help you use JMeter efficiently.

Chapter 2, Handling Responses, details how to handle various server and application responses.

Chapter 3, Building Robust Test Plans with Controllers, covers five useful and often encountered JMeter controllers and how to apply them to your use cases.

Chapter 4, Testing Services, details how to test web services and supporting application resources with JMeter.

Chapter 5, Diving into Distributed Testing, takes an in-depth look at leveraging the Cloud for performance testing. We cover three Cloud providers and see how to roll our own when the need arises.

Chapter 6, Extending JMeter, covers how to extend JMeter with plugins. We also detail how to write your own JMeter plugin.

Chapter 7, Building, Debugging, and Analyzing the Results of Test Plans, discusses some useful components in JMeter and how to leverage them, as well as how to build realistic, robust, and maintainable scripts.

Chapter 8, Beyond the Basics, covers integrating JMeter into continuous delivery workflows, scaling JMeter, and many other advanced tips and pointers.

Appendix, Installing the Supporting Software Needed for This Book, covers how to install the supporting software and how to proceed with the recipes in the chapters.

What you need for this book

To follow along with the examples in this book, you will need the following:

- ▶ A computer with an Internet connection
- ▶ Apache JMeter: <http://jmeter.apache.org/>
- ▶ Java Runtime Environment (JRE) or Java Development Kit (JDK): <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

In addition, for some recipes in *Chapter 1, JMeter Fundamentals*, you will need the following:

- ▶ Google Chrome Browser: <https://www.google.com/chrome/browser/>
- ▶ Apache Maven: <http://maven.apache.org/>
- ▶ YourKit Java Profiler: <http://www.yourkit.com/> (this is commercial ware, but the free trial will do to follow along with the recipe)
- ▶ A Heroku account: <http://www.heroku.com>
- ▶ Apache Tomcat: <http://tomcat.apache.org/download-70.cgi>

For Chapter 5, *Diving into Distributed Testing*, you will need the following:

- ▶ Vagrant: <http://www.vagrantup.com/>
- ▶ An AWS account: <http://aws.amazon.com/>
- ▶ A Flood.IO account: <https://flood.io/>
- ▶ A BlazeMeter account: <http://blazemeter.com>

For Chapter 8, *Beyond the Basics*, any free Java IDE will do. This includes the following:

- ▶ IntelliJ Community Edition: <http://www.jetbrains.com/idea/download/>
- ▶ Eclipse: <https://www.eclipse.org/downloads/>

The book contains pointers and additional helpful links to set all these up.

Who this book is for

This book is targeted primarily at developers and testers. Developers who have always been intrigued by performance testing and want to get in on the action will find it extremely useful, and will gain essential skills as they walk through the practical recipes in the book.

Testers will also benefit from this book since it will guide them through solving practical, real-world challenges when testing modern web applications, giving them ample knowledge to aid them in becoming better testers. Additionally, they will be exposed to certain helpful testing tools that will come in handy at some point in their testing careers.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows:
"Open the `add_cookie_manager.jmx` test script."

A block of code is set as follows:

```
JMeterVariables:
JMeterThread.last_sample_ok=true
JMeterThread.pack=org.apache.jmeter.threads.SamplePackage@2ae97e14
START.HMS=053854
START.MS=1396517934834
START.YMD=20140403
TESTSTART.MS=1396702498383
speed=7.7
```

```
speed_g=1
speed_g0="speed":7.7
speed_g1=7.7
sunrise=1396694648
sunrise_g=1
sunrise_g0="sunrise":1396694648
sunrise_g1=1396694648
sunset=1396740883
sunset_g=1
sunset_g0="sunset":1396740883
sunset_g1=1396740883
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
<body>
Posted by
<a href='/login'>login</a>
<a href='/register'>register</a>
```

Any command-line input or output is written as follows:

```
Adding newrelic:stark on frozen-headland-2987... done, v7 (free)
Use `heroku addons:docs newrelic` to view documentation.
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Press the **Record** button."



Warnings or important notes appear in a box like this.



Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to feedback@packtpub.com, and mention the book title via the subject of your message. If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on www.packtpub.com/authors.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.packtpub.com>. If you purchased this book elsewhere, you can visit <http://www.packtpub.com/support> and register to have the files e-mailed directly to you.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting <http://www.packtpub.com/submit-errata>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from <http://www.packtpub.com/support>.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at copyright@packtpub.com with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at questions@packtpub.com if you are having a problem with any aspect of the book, and we will do our best to address it.

1

JMeter Fundamentals

In this chapter, we will cover the following recipes:

- ▶ Executing a test script
- ▶ Recording a script via HTTP(S) Test Script Recorder
- ▶ Recording scripts via the Chrome browser extension
- ▶ Converting HTTP web archives (HAR) to JMeter test plans
- ▶ Viewing and analyzing test results
- ▶ Feeding data into a script
- ▶ Using timers
- ▶ Managing HTTP user sessions
- ▶ Testing Single Page Applications (SPAs)
- ▶ Testing AJAX-centric applications

Introduction

In this chapter, you will learn the fundamentals and intermediate skills that will help you work better with JMeter. Tasks such as executing or recording scripts are routine and you can be almost certain that you will need to perform them, sometimes on a daily basis. Also, we will learn how to view and analyze the results of test executions, feed data into test scripts, and make test scripts mimic user behavior with the aid of timers. In later recipes, we dive into dealing with handling authentication, authorization, and testing today's new breed of applications with JMeter.

Executing a test script

Sometimes, the test scripts to execute have already been recorded and handed over to you to run. In this recipe, we will show you just how to go about executing a script that was prerecorded.

How to do it...

To execute a test script, perform the following steps:

1. Open the command line prompt.
2. Change to the directory of your JMeter install. We'll refer to this as `JMETER_HOME`.



Refer to *Appendix* for JMeter installation instructions.

3. Change `JMETER_HOME` to the `bin` directory.
4. To launch the JMeter GUI on Windows, type in `jmeter.bat`. Alternatively, for Unix/Mac OS, type in `./jmeter`.
5. Navigate to **File | Open**.



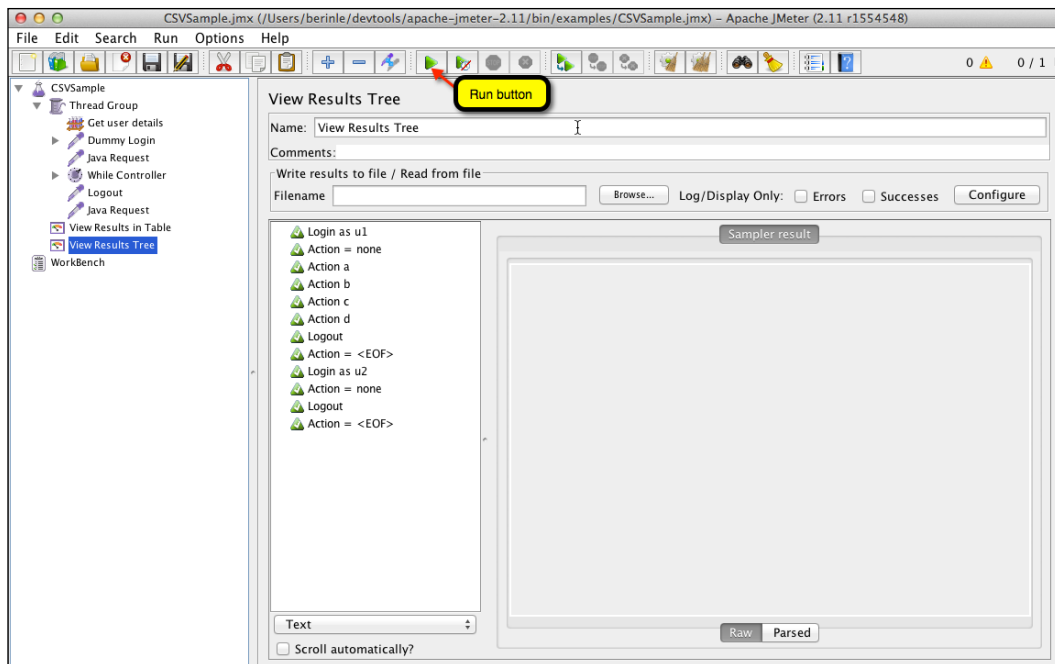
Alternative key binding: Mac OS (*Command + O*),
Windows (*Ctrl + O*).

6. Navigate to the script you want to execute. For example, `ch1/getting_started.jmx` provided with the sample code.
7. Press the green start icon at the top.



Alternative key binding: Mac OS (*Command + R*),
Windows (*Ctrl + R*).

8. View the results from one of the added listeners. If you are using `getting_started.jmx` from step 6, click on the **View Results Tree** listener as shown in the following screenshot:



Executing a test script

How it works...

Test scripts are a series of prerecorded requests issued against an application. They are captured interactions of a user's actions with the application. These include visiting a URL, navigating around several pages, logging in, and so on. JMeter, like most test tools, has the ability to record and replay back test scripts. JMeter test scripts are stored in the XML (extendable markup language) format with the `.jmx` extension. Curious users can open the test script in a text editor of their choice (for example, Sublime Text, Notepad, and so on) and view exactly what it is doing, though, it is much more clear to understand what a script does by looking at it in JMeter's GUI.

After executing the test, results are reported through any configured listeners. A script can have one or several listeners attached to it (see recipes in *Chapter 7, Building, Debugging, and Analyzing the Results of Test Plans*).

There's more...

Scripts can also be executed in what JMeter refers to as non-GUI mode, which is completely on the command line without launching the JMeter GUI. To do this, instead of executing `jmeter.bat` or `JMeter`, like we did in step 4, we'll use certain JMeter command-line arguments to specify the test script and the resulting output file, which we can then go back and view with the JMeter GUI after the tests are executed.

On Unix/Mac OS, type in the following command:

```
./jmeter -n -t [path to test script] -l [path to results files]
```

On Windows, type in the following command:

```
jmeter.bat -n -t [path to test script] -l [path to results files]
```

An example of running in non-GUI mode

Here is an example of running code in non-GUI mode: This is how we executed the `google_simulation.jmx` test script in our bundled sample in non-GUI mode. As you can see, we supplied parameters `-n` and `-t` and the absolute path to where our script is located (`/Users/berinle/workspace/jmeter-cookbook/ch1/google_simulation.jmx`) followed by the path to the results file (`/Users/berinle/workspace/jmeter-cookbook/ch1/google_simulation_result.csv`).

```
./jmeter -n -t /Users/berinle/workspace/jmeter-cookbook/ch1/google_simulation.jmx -l /Users/berinle/workspace/jmeter-cookbook/ch1/google_simulation_result.csv
```

Created the tree successfully using `/Users/berinle/workspace/jmeter-cookbook/ch1/google_simulation.jmx`.

Starting the test @ Tue Feb 25 05:07:45 EST 2014 (1393322865378)

Waiting for possible shutdown message on port 4445.

Tidying up ... @ Tue Feb 25 05:08:46 EST 2014 (1393322926156)

... end of run



Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.packtpub.com>. If you purchased this book elsewhere, you can visit <http://www.packtpub.com/support> and register to have the files e-mailed directly to you.