# Plone Performance Testing Documentation

Release 1.0

**Timo Stollenwerk** 

January 21, 2015

#### Contents

1 Getting Started with jMeter					
	1.1	Introduction			
	1.2	Step 1: Install jMeter			
	1.3	Step 2 : Create a Test Plan			
	1.4	Step 3: Create a Thread Group			
	1.5	Step 4: Create HTTP Request Defaults			
	1.6	Step 5: Create an HTTP Request			
	1.7	Step 6: Aggregate Graph			
	1.8	Step 7: Save Test Plan			
	1.9	Step 8: Run Performance Tests			
	1.10	Next Steps			
2	Indic	tes and tables			

# Getting Started with jMeter

### **1.1 Introduction**

This document explains how to create a minimal performance test plan with jMeter to run performance tests agains a Plone site.

## 1.2 Step 1: Install jMeter

The easiest way to install jMeter ist to install it from the package sources of your distribution (e.g. on Debian/Ubuntu):

\$ sudo apt-get install jmeter

Note: Further install options http://jmeter.apache.org/usermanual/get-started.html

## 1.3 Step 2 : Create a Test Plan

Once you've started jMeter (e.g. by typing "jmeter" into the command line or by clicking on the jMeter icon on your desktop) you will see an empty Testplan:

A minimal jMeter testplan consists of the following elements:

• Thread Group:

The root element of every test plan. It simulates the (concurrent) users than run all requests. Each thread simulates a single user.

• HTTP Request Default (Configuration Element):

The default values for all HTTP requests within this thread group.

• HTTP Request (Sampler):

An HTTP request that can be used to measure the time it takes to answer the request to a single URL.

• Aggregate Graph (Listener):

An aggregate graph listener can be used to generate statistics about the HTTP requests that are used within a thread group.

#### <u>File Edit Run Options Help</u>

Fue Fair Wan Obtions Web		0/0
Testplan	WorkBench	
	Name: WorkBench	
	Comments:	



# 1.4 Step 3: Create a Thread Group

In order to create a thread group, right-click on the Testplan in the left column of the jMeter GUI. Then choose "Add" and then "Threads (Users)" and "Thread-Group". The titles might vary dependend on the jMeter version you are using:

```
Testplan > Add > Threads (Users) > Thread-Group
```

Fill out the right column with the following values:

```
Name: Users
Number of Threads (users): 2
Ramp-Up Period (in seconds): 1
Loop Count: 10
```

"Number of Threads (users)" is the number of concurrent users that jMeter will create to run the HTTP requests. The "Ramp-Up Period (in seconds)" are the seconds that jMeter waits between running the threads. The "Loop Count" describes how many times jMeter will run the tests.

```
<u>File Edit Run Options Help</u>
```

Thread Group          Name:       Users         Comments:         Ation to be taken after a Sampler error	
WorkBench     Name:   Users   Comments:     Action to be taken after a Sampler error      © Continue      Stop Test     Stop Test Now        Thread Properties   Number of Threads (users):   Ramp-Up Period (in seconds):   Loop Count:   Forever   Scheduler	
Comments: Aggregate Graph WorkBench Comments: Action to be taken after a Sampler error © Continue © Stop Thread © Stop Test © Stop Test Now Thread Properties Number of Threads (users): 1 Ramp-Up Period (in seconds): 1 Loop Count: © Forever 10 © Scheduler	
Action to be taken after a Sampler error © Continue O Stop Thread O Stop Test O Stop Test Now Thread Properties Number of Threads (users): 1 Ramp-Up Period (in seconds): 1 Loop Count: Forever 10 Scheduler	
© Continue O Stop Thread O Stop Test O Stop Test Now Thread Properties Number of Threads (users): 1 Ramp-Up Period (in seconds): 1 Loop Count: Forever 10 O Scheduler	
Thread Properties Number of Threads (users): 1 Ramp-Up Period (in seconds): 1 Loop Count: Forever 10 Scheduler	
Number of Threads (users): 1 Ramp-Up Period (in seconds): 1 Loop Count: Forever 10 Scheduler	
Ramp-Up Period (in seconds): 1     Loop Count: Forever     IO     Scheduler	
Loop Count: Forever 10	
Scheduler	

Figure 1.2: Thread Group

# 1.5 Step 4: Create HTTP Request Defaults

In order to run performance tests on certain resources, jMeter fires up HTTP requests. jMeter allows to create default values for all HTTP requests within a test plan, so that we do not have to add those values for every single HTTP request in our plan.

To add a "HTTP Request Defaults" element, right-click on the "Testplan" in the left column of the jMeter GUI, choose "Users", "Add", "Config Element" and "HTTP Request Defaults":

Testplan > Users > Add > Config Element > HTTP Request Defaults

We set the default HTTP request values to use a standard Plone site that needs to be available under "http://localhost:8080/Plone":

Name: localhost:8080/Plone Server Name or IP: localhost Port Number: 8080 Path /Plone

Testplan	HTTP Request Defaults			0.				
	Name: leastbest: 8020/Plana							
Front-page	Name: localhost:8080/Plone							
☐ rront-page S Aggregate Graph WorkBench	Web Copier	Comments:						
	Web Server	Dest Number 2000	Timeouts (milliseconds)					
		Port Number: 8080	Connect:	Response:				
	Protocol (default http):	Content encoding:						
	Path: /Plone2							
	Ser	d Parameters With the Requ	est:					
	Name:	Value		Encode? Include E				

Figure 1.3: HTTP Request Defaults

# **1.6 Step 5: Create an HTTP Request**

In order to run a performance test on the front page of our Plone site, we create a "HTTP Request" element, by right-click on the "Testplan" in the left column of the jMeter GUI, choose "Users", "Add", "Sampler" and "HTTP Request":

Testplan > Users > Add > Sampler > HTTP Request

We give the HTTP request a name and provide a path. We don't have to fill out the server name or port number, since our "HTTP Request Defaults" element takes care of that:

Name: Front-page Path: /Plone/front-page

**Note:** The values are just defaults, not prefixes. This means if you want to add an HTTP request for "front-page" it is not enough to use "/front-page", you always have to provide the full path "/Plone/front-page".

- Â Testplan		0/1					
- Trusers	HTTP Request						
localhost: 8080/Plone	Name: Front-page						
— 🥕 Front-page	Comments:						
Aggregate Graph	Web Server	s (milliseconds)					
MorkBench	Server Name or IP: Port Number: Connect:	Response:					
	THTTP Request						
	Protocol (default http): Method: GET 💌 Content encoding:						
	Path: //front-page						
	☑ Redirect Automatically ☐ Follow Redirects ☑ Use KeepAlive ☐ Use multipart Send Parameters With the Request:	/form-data for HTTP I					
	Name: Value	Encode? Inclu					
	Add Delete						
	Send Files With the Request:						
	File Path:	Paramete MIME					
	Add Browse Delete						
	Optional lasks						
	Retrieve All Embedded Resources from HTML Files Use as Monitor Save re	sponse as MD5 has					
	Embedded UBLs must match:						

Figure 1.4: HTTP Request

# 1.7 Step 6: Aggregate Graph

In order to display the results of our HTTP requests, we have to add a listener in jMeter by right-clicking on the "Testplan" in the left column of the jMeter GUI, choose "Users", "Add", "Listener" and "Aggregate Graph":

Testplan > Users > Add > Listener > Aggregate Graph

- 8 Tastolan									0/1
- Tr Users	Aggrega	Aggregate Graph							
localhost:8080/Plone	Name: Agg	Name: Aggregate Graph							
- 🎤 Front-page	Comments	Comments:							
- 🔄 Aggregate Graph	Write res	Write results to file / Read from file							
WorkBench	Filename				Browse	Log/Display	y Only: 🔲 Err	ors 📃 Suc	cesses
	Label	# Samples	Average	Median	90% Line	Min	Max	Error %	Through
	Front-page	10	9	Prodiciti	6 43	5	43	100,00%	94,3/
	TOTAL	10	9		6 43	5	43	100,00%	94,3/
	Max length Width Height Column A	n of x-axis lab	el Display G	raph	Save Graph	Save Tab	le Data 🗌	Save Table	Header
	1.00	°	Graph						
	50 -8	0 		<u>م</u>					
	Millisecon			Front-page					
			Δ1	erade					
			A	craye					
			Response 7	ìme					

# 1.8 Step 7: Save Test Plan

Before we can run our jMeter test plan we should save it by choosing "Save Testplan as" from the "File" menu: File > Save Testplan as

## 1.9 Step 8: Run Performance Tests

To run our test plan choose "Start" from the "Run" menu:

Run > Start

# 1.10 Next Steps

Creating a test plan:

http://jmeter.apache.org/usermanual/build-web-test-plan.html

Debug requests with "View Results Tree":

Testplan > Users > Add > Listener > View Results Tree (optional)

CHAPTER 2

Indices and tables

- genindex
- modindex
- search