



Learning JMeter in 60 Minutes



785 Castro Street, Mountain View, CA 94041 | 1.855.445.2285 | info@blazemeter.com



@alongir

About Me

Alon Girmionsky

- Hands-On Developer for the Past 20 Years
- Expertise
 - Large-Scale Web and Mobile Applications
 - IP & Internet
- Founder & CEO of **BlazeMeter**
 - A Load Testing Platform as a Service
 - 100% Compatible with Apache JMeter

TESTS
500,000

USERS
50,000

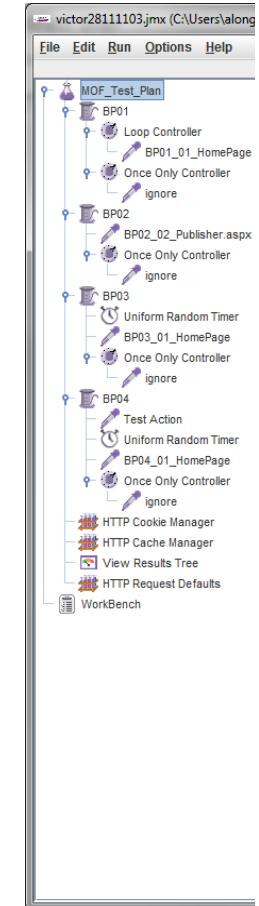


Any Script, Any Scenario

Comprehensive scripting at your fingertips

- HTTP / HTTPS
- Login / Signup
- Form POSTing / Credit card transactions
- Web services
- Parameters extraction
- Mobile
- MSSQL, Oracle SQL, MySQL
- Multiple user profiles

- » Any script
- » Any scenario
- » Any protocol



100% Apache JMeter Compatible



785 Castro Street, Mountain View, CA 94041 | 1.855.445.2285 | info@blazemeter.com



@alongir

How to Install JMeter?

Download the Latest Release of JMeter

- <http://jmeter.apache.org/>

Download

- [Download Releases](#)
- [Developer \(Nightly\) Builds](#)

Documentation

- [User Manual](#)
- [Best Practices](#)
- [Component Reference](#)
- [Functions Reference](#)
- [Javadocs](#)
- [Localisation \(Translator's Guide\)](#)
- [Building JMeter and Add-Ons](#)
- [JMeter Wiki](#)
- [FAQ \(Wiki\)](#)

Tutorials (PDF format)

Other mirrors: <http://apache.spd.co.il/>

The [KEYS](#) link links to the code signing
[MD5](#) link downloads the checksum from

For more information concerning Apache
[KEYS](#)

Apache JMeter 2.9 (Requires Java 6 or

Binaries

[apache-jmeter-2.9.tgz](#) [md5](#) [pgp](#)
[apache-jmeter-2.9.zip](#) [md5](#) [pgp](#)



The Thread Group

- What is this? - A group of virtual users where each distinct user will follow a certain business process

The Thread Group Properties:

Property	Description
No of Threads	The maximum number of threads that will run concurrently at the end of the ramp up period
Ramp up	The period is will take the load to ramp up from “idle” to maximum load
Iterations	The number of iteration each thread will execute
Duration	The duration of the test from start to end
The Thread	The sequence of events that comprise the simulated business process



The Thread Group

The screenshot shows the JMeter Test Plan editor interface. At the top, there is a toolbar with various icons for file operations like Open, Save, and Print, as well as for modifying and running tests. Below the toolbar is a menu bar with "Test Plan" selected. A sub-menu "Add" is open, showing options: "Threads (Users)" (which is currently selected), "Thread Group", "Test Fragment", and "Config Element". To the right of the sub-menu, two examples are shown: "jp@gc - Stepping Thread Group" and "jp@gc - Ultimate Thread Group".

The main workspace shows a tree view on the left under "Test Plan": "Thread Group" is selected and highlighted in green, while "WorkBench" is also listed. On the right, a detailed configuration dialog for the "Thread Group" is open. The "Name:" field contains "Thread Group". Under "Comments:", there is a section for "Action to be taken after a Sampler error" with two radio button options: "Continue" (selected) and "Start Next Thread Loop".

The "Thread Properties" section includes:

- "Number of Threads (users):" set to 1
- "Ramp-Up Period (in seconds):" set to 1
- "Loop Count:" with a "Forever" checkbox checked and a value of 1
- Two unchecked checkboxes at the bottom: "Delay Thread creation until needed" and "Scheduler"



The Sampler

- What is this? - The actual request object with its properties

Supported Protocols:

- HTTP
- FTP
- LDAP
- MSSQL, MySQL, Oracle SQL
- Mongo DB
- SOAP/XML-RPC
- Generic
- SMTP
- oAuth
- Rest
- TCP
- UDP
- Many many more



The Sampler

The screenshot shows the BlazeMeter Test Plan interface. On the left, there's a tree view with 'Test Plan' at the root, followed by 'Thread Group' (which is selected and highlighted in green), 'HTTP Request 1', 'HTTP Request 2', and 'WorkBench'. A context menu is open over the 'Thread Group' node, with 'Add' selected. The 'Add' submenu lists various samplers: Access Log Sampler, AJP/1.3 Sampler, BeanShell Sampler, BSF Sampler, Debug Sampler, FTP Request, HTTP Request (which is also highlighted in green), Java Request, JDBC Request, JMS Point-to-Point, and IMQ Publisher. Below the menu is a toolbar with various icons. To the right, the main panel shows the 'HTTP Request' configuration dialog. It has fields for 'Name:' (set to 'HTTP Request'), 'Comments:', 'Web Server', 'Server Name or IP:', 'Implementation:' (dropdown), 'Protocol [http:]' (set to 'http'), 'Path:' (empty), and checkboxes for 'Redirect Automatically', 'Follow Redirects', and 'Use KeepAlive'. There's also a 'Parameters' section with a 'Send Parameters' button and a 'Detail' button.



The Listener

- What is this? - A fancy name for a report
- Type of Listeners
 - Aggregated
 - Detailed

The screenshot shows a JMeter interface with three main panels:

- Left Panel (Aggregated):** Displays three tree nodes: "First Page", "Second Page", and "Third Page".
- Middle Panel (Detailed):** Shows the following metrics:
 - Size in bytes: 37208
 - Headers size in bytes: 346
 - Body size in bytes: 36862
 - Sample Count: 1
 - Error Count: 0
 - Response code: 200
 - Response message: OK
- Right Panel (Summary Report):** Contains fields for "Name" (Summary Report), "Comments", and options for "Write results to file / Read from file" and "Filename". Below these are two tables:
 - Sampler result:** A table showing the count of samples for each page and their total.

Label	# Samples	Average	Min
First Page	1	1252	1252
Second Page	1	125	125
Third Page	1	380	380
TOTAL	3	585	125
 - Request | Response data:** A table showing the details of the single sample.

Label	Value
Size in bytes	37208
Headers size in bytes	346
Body size in bytes	36862
Sample Count	1
Error Count	0
Response code	200
Response message	OK

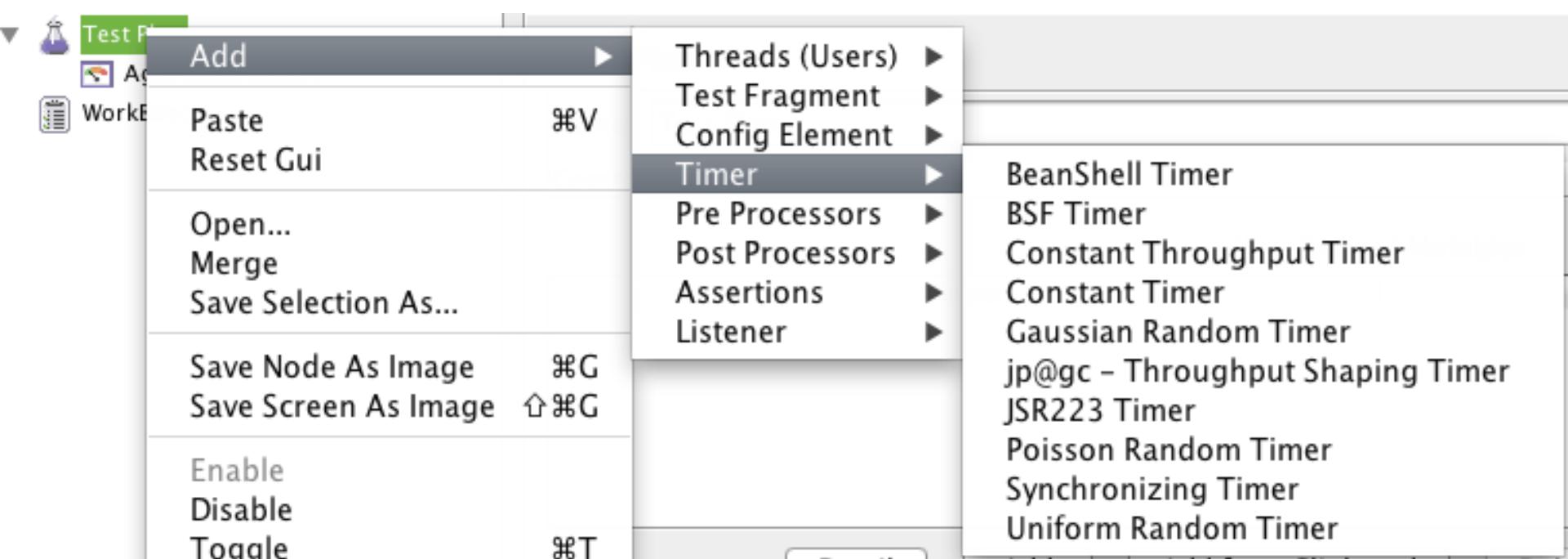
- Use for debugging only
- Consume a lot of memory
- Use when you want to build custom reports (e.g. for errors only)



The Timer

- What is this? - A fancy name for delay or think time

Type of Timers:



The Timer

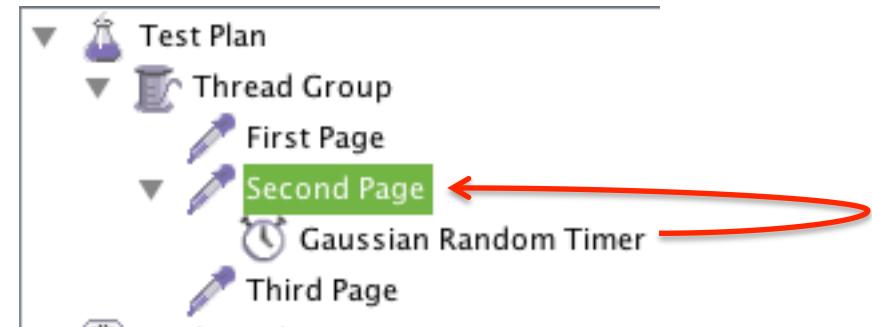
Timers belong to a scope- We need to understand this

- Timers (as other controllers/elements) work in scope
- Once put in scope, applies to every element in the scope
- If scope is of one element, applies only to that element

Upper Scope



Child Scope



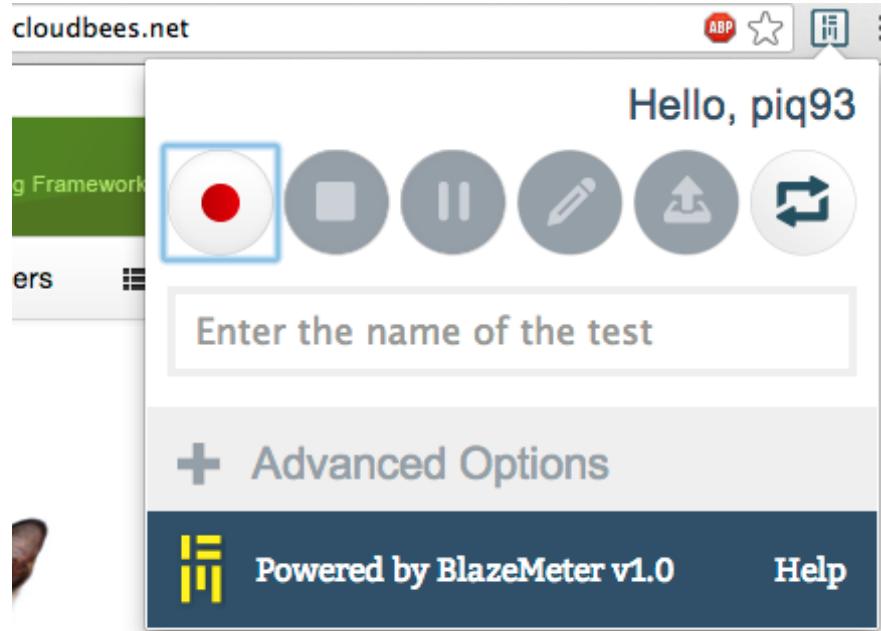
Record Using a Proxy

- A good way to start a script
- Don't forget to clean and parametrize

The screenshot shows the BlazeMeter Test Plan configuration interface. On the left, there's a sidebar with various tools: HTTP Cookie Manager, First Page, Response Assertion, Second Page, Third Page, Gaussian Random Timer, Summary Report, View Results Tree, Recording Controller (with several recorded URLs listed), WorkBench, and HTTP Proxy Server (which is selected and highlighted in green). The main area contains several sections: Global Settings (Port: 8088), Test plan content (Target Controller: Test Plan > Thread Group > Recording Controller, Grouping: Do not group), HTTP Sampler settings (Type dropdown, Redirect Automatically checkbox, Follow Redirects checkbox checked, Use KeepAlive checkbox checked), Content-type filter (Include and Exclude fields), URL Patterns to Include (empty list with Add, Delete, and Add from Clipboard buttons), and URL Patterns to Exclude (empty list with Add, Delete, and Add from Clipboard buttons).



Record Using a Chrome Extension



The screenshot shows the BlazeMeter test editor. On the left, a tree view displays a recorded object with ID 37844, which includes a label, URL, method (GET), and timestamp (1375373629599). The "headers" section is currently selected and highlighted in yellow. On the right, a context menu for the headers section is open, listing options: Type, Sort, Append, Insert, Duplicate, and Remove.



Creating a Script Using a List of URLs



JMeter in the Cloud



GET and POST Requests



From Google Analytics



Drupal Test

If you want to start load testing without writing test scripts - no problem!

Simply enter the URLs you want to test and desired max concurrent users for the test – that's it.

BlazeMeter will create a load of fully simulated browser users. Each virtual user will load each URL sequentially. For HTML pages all included resources on the page will also be loaded (images, css, js, etc). Each virtual user has their own browser session, cache and cookies.

The reports will provide you with all the KPIs you need and you can easily share all the results with your team.

URLs to Test

Label:	URL:	Method:
Home	<input type="text" value="http://yahoo.com"/>	<input checked="" type="radio"/> GET <input type="radio"/> POST
About	<input type="text" value="http://google.com"/>	<input checked="" type="radio"/> GET <input type="radio"/> POST
Post	<input type="text" value="http://post.co"/>	<input type="radio"/> GET <input checked="" type="radio"/> POST

POST Body:



Test Defaults

Necessary for Every Test

The screenshot shows the BlazeMeter Test Plan interface. On the left, there's a tree view under 'Test Plan' with 'Thread Group' expanded. Inside 'Thread Group', four items are listed: 'HTTP Request Defaults', 'HTTP Cache Manager', 'HTTP Header Manager' (which is highlighted with a red box), and 'HTTP Cookie Manager'. Below this tree view, there are icons for 'First Page', 'Second Page', 'Third Page', and 'Gaussian Rand'. To the right of the tree view is a detailed configuration panel for the 'HTTP Header Manager'. It has fields for 'Name' (set to 'HTTP Header Manager') and 'Comments'. A section titled 'Headers Stored in the Header Manager' lists several headers with their values: 'Accept-Language' (en-US,en;q=0.5), 'Accept' (text/css,*/*;q=0.1), 'User-Agent' (Mozilla/5.0 (Macintosh; Intel Mac OS X 1), 'Referer' (http://www.yahoo.com/), and 'Accept-Encoding' (gzip, deflate). At the bottom of the page, there are tabs for 'Header' and 'Parameters'.

HTTP Header Manager

Name: HTTP Header Manager

Comments:

Headers Stored in the Header Manager

Accept-Language
Accept
User-Agent
Referer
Accept-Encoding

en-US,en;q=0.5
text/css,*/*;q=0.1
Mozilla/5.0 (Macintosh; Intel Mac OS X 1
http://www.yahoo.com/
gzip, deflate

Request Defaults

HTTP Request Defaults

Server

Name or IP: s3-eu-west-1.amazonaws.com Port Number: Timeouts (milliseconds)

Connect: Response:

Request

Implementation: Content encoding:

Header

Parameters



Practice: Writing Your First Script

Getting your hands dirty with Apache JMeter

Goal

- Create a test simulating a gradual ramp up of up to 10 users against these two pages:
 - http:// ..
 - http:// ..

How to go about it:

- Use a proxy to record
- Clean and label
- Add the defaults
- Add listeners and delays
- Run
- Analyze the results



Advanced Scripting Concepts



Single Iteration Vs Continuous Load

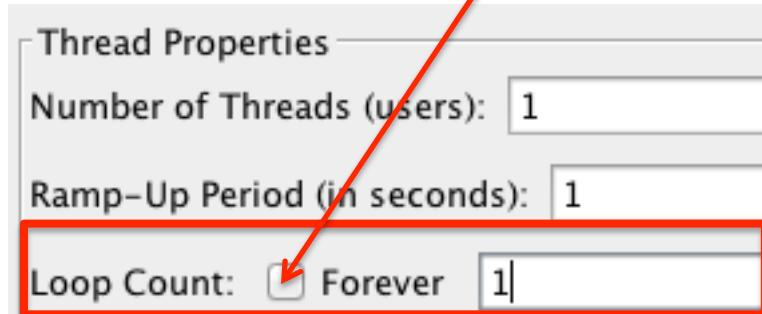
Single Iteration

KPI: Number of Unique Users

- Simulate a ramp up of unique users
- Low concurrency level

For Example:

- I want only 25K users visiting my website over 1 hour



Continuous

KPI: Load Level

- Keep the load steady
- High concurrency level

For Example:

- I want a constant and steady load of 25K users visiting my website over 1 hour (I'll probably end up with 200K users in that hour)

Assertions

- Define Proprietary Errors
- Communicate Events

The screenshot shows the 'Response Assertion' configuration dialog in JMeter. On the left, there's a sidebar with icons for 'First Page', 'Response Assertion' (which is selected and highlighted in green), 'Second Page', 'Third Page', 'Gaussian Random Timer', 'WorkBench', and 'HTTP Proxy Server'. The main panel has several sections:

- Response Field to Test:** Radio buttons for 'Main sample and sub-samples' (unchecked), 'Main sample only' (checked), 'Sub-samples only' (unchecked), and 'JMeter Variable' (unchecked).
- Pattern Matching Rules:** Radio buttons for 'Text Response' (checked), 'Document (text)' (unchecked), 'URL Sampled' (unchecked), 'Response Code' (unchecked), and 'Response Message' (unchecked).
- Pattern to Test:** A dropdown menu showing 'Contains' (checked), 'Matches', 'Equals', 'Substring', and 'None'.
- Patterns to Test:** A text input field containing the value 'Success'.



Using CSV Files

The best way to individualize each thread

The screenshot shows the BlazeMeter Test Plan interface. On the left, the Test Plan tree view is open, showing a Thread Group node with several child elements: HTTP Request Defaults, HTTP Cache Manager, HTTP Header Manager, HTTP Cookie Manager, First Page, Response Assertion, Second Page, Third Page, Gaussian Random Timer, Summary Report, View Results Tree, Recording Controller, and an HTTP Proxy Server. A CSV Data Set Config element is selected and highlighted in green.

A context menu is open over the Thread Group node, listing options: Logic Controller, Config Element, Timer, Pre Processors, Sampler, Counter, CSV Data Set Config, FTP Request Defaults, and HTTP Authorization Manager. The 'CSV Data Set Config' option is highlighted in the menu.

The right panel displays the configuration dialog for the CSV Data Set Config element. The configuration fields are as follows:

- Name: CSV Data Set Config
- Comments: (empty)
- Configure the CSV Data Source
 - Filename: myfile.csv
 - File encoding: (empty)
 - Variable Names (comma-delimited): param1,param2
 - Delimiter (use '\t' for tab): ,
 - Allow quoted data?: False
 - Recycle on EOF ?: True
 - Stop thread on EOF ?: False
 - Sharing mode: All threads



External Properties

- Pass external properties to the script
- Customize same script to support various scenarios

Methods

- Command line arguments (-J)
- Properties file: user.properties

This section will allow you to set JMeter properties [?]

Key:

Value:



Parameter Extraction

Extract from a page and use for the next page

The screenshot shows the JMeter Test Plan tree on the left and the configuration dialog for a 'Regular Expression Extractor' on the right.

Test Plan Structure:

- Test Plan
 - Thread Group
 - HTTP Request Defaults
 - HTTP Cache Manager
 - HTTP Header Manager
 - HTTP Cookie Manager
 - First Page
 - Response Assertion
 - View Results Tree
 - Regular Expression Extractor
 - Debug Sampler
 - View Results Tree
 - Second Page
 - View Results Tree
 - Third Page
 - Gaussian Random Timer
 - Summary Report
 - View Results Tree

Regular Expression Extractor

Name: Regular Expression Extractor

Comments:

Apply to:

Main sample and sub-samples Main sample only

Response Field to check

Body Body (unescape) Body as

Reference Name:

example

Regular Expression:

\[_setAccount', '(.*)'\],

Template:

\$1\$

Match No. (0 for Random):

1

Default Value:

NOTFOUND



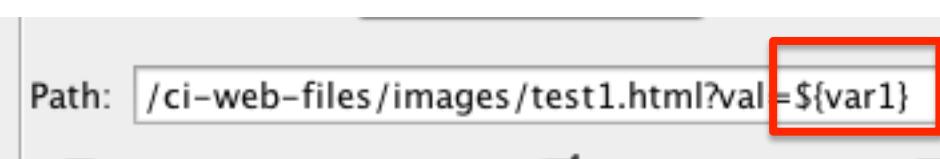
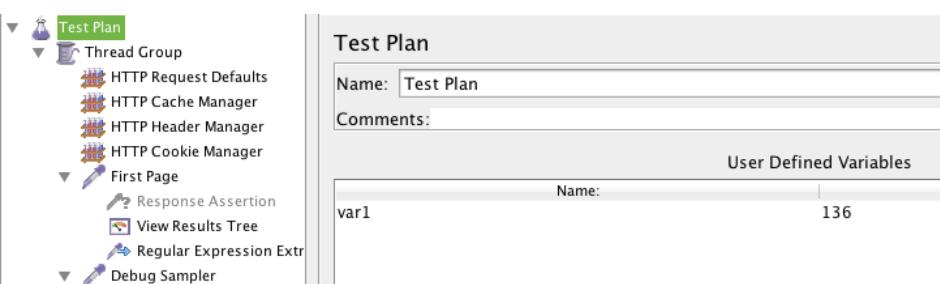
Variables and Properties

Variables

- Usage: \${var}
- Scope: unique per thread

Properties

- Usage: \${_P(prop)}
- Scope: Test
- Pass external properties to the script
- Customize same script to support various scenarios



Methods

- Command line arguments (-J)
- Properties file: user.properties



Debugging

The Debug Sampler

- The developer's best friend

The screenshot shows the JMeter interface with the following components:

- Left Sidebar:** A tree view of test elements:
 - Response Assertion
 - View Results Tree
 - Regular Expression Extr
 - Debug Sampler (selected)
 - View Results Tree (highlighted)
 - Second Page
 - View Results Tree
 - Third Page
 - Gaussian Random Timer
 - Summary Report
 - View Results Tree
 - Recording Controller
 - HTTP Proxy Server
 - CSV Data Set Config
- Middle Panel:** A list of three "Debug Sampler" elements.
- Right Panel:** A detailed view of the selected "Debug Sampler". It includes tabs for "Sampler result", "Request", and "Res". The "Sampler result" tab displays the following output:

```
JMeterVariables:  
JMeterThread.last_sample_ok=true  
JMeterThread.pack=org.apache.jmeter.threads.Sar  
START.HMS=091548  
START.MS=1368252948918  
START.YMD=20130511  
TESTSTART.MS=1368255470901  
example=UA-3225456-9  
example_g=1  
example_g0=['_setAccount', 'UA-3225456-9'],  
example_g1=UA-3225456-9  
param1=<EOF>  
param2=<EOF>
```



Practice: Writing Your Second Script

Getting your hands dirty with Apache JMeter

Goal

- Create a test simulating a gradual ramp up of up to 10 users logging in to:
 - http:// ..

How to go about it:

- Use a proxy to record
- Use RexEx to parameterize
- Clean, label, add defaults, listeners and delays
- Add a CSV file with user names and passwords
- Use assertions to validate responses
- Run
- Analyze the results



Performance

1-on-1



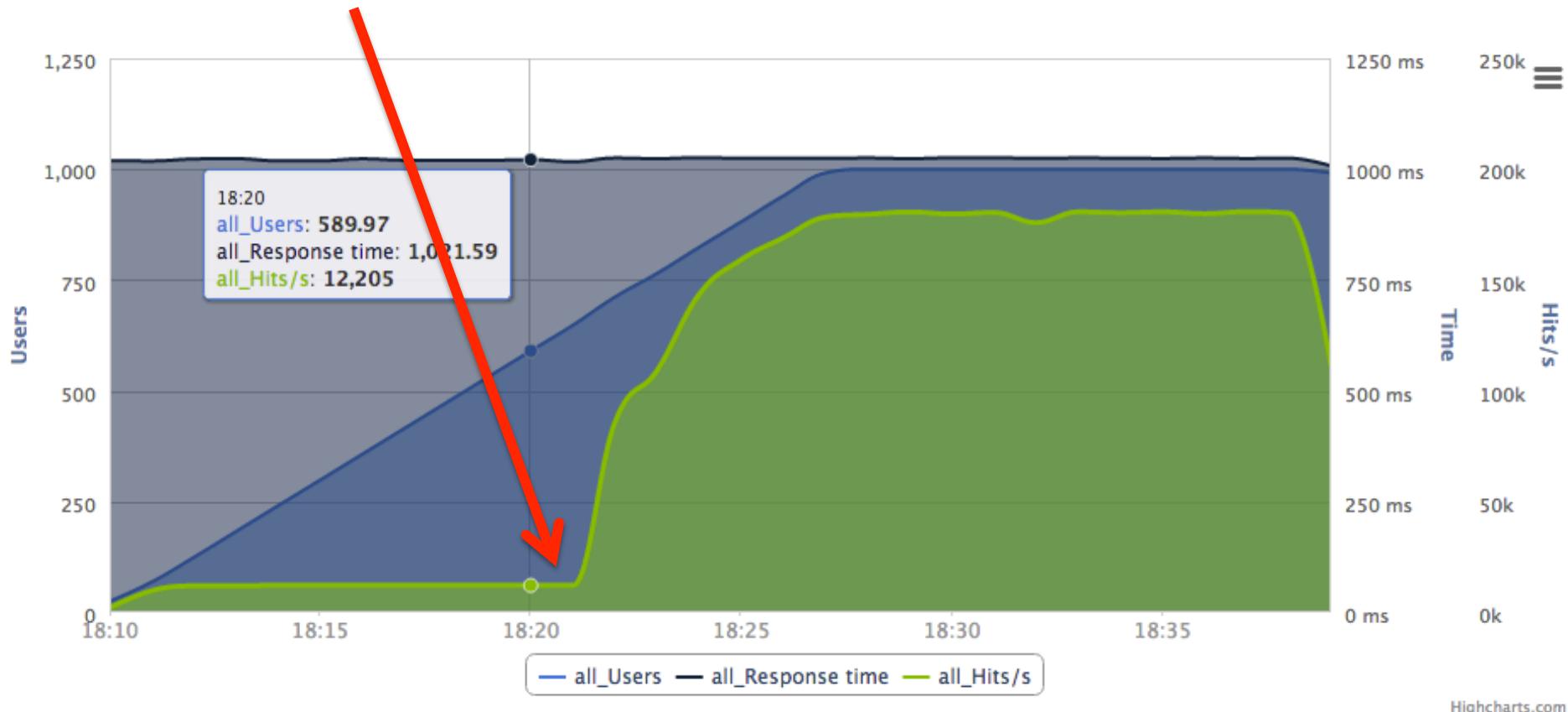
785 Castro Street, Mountain View, CA 94041 | 1.855.445.2285 | info@blazemeter.com



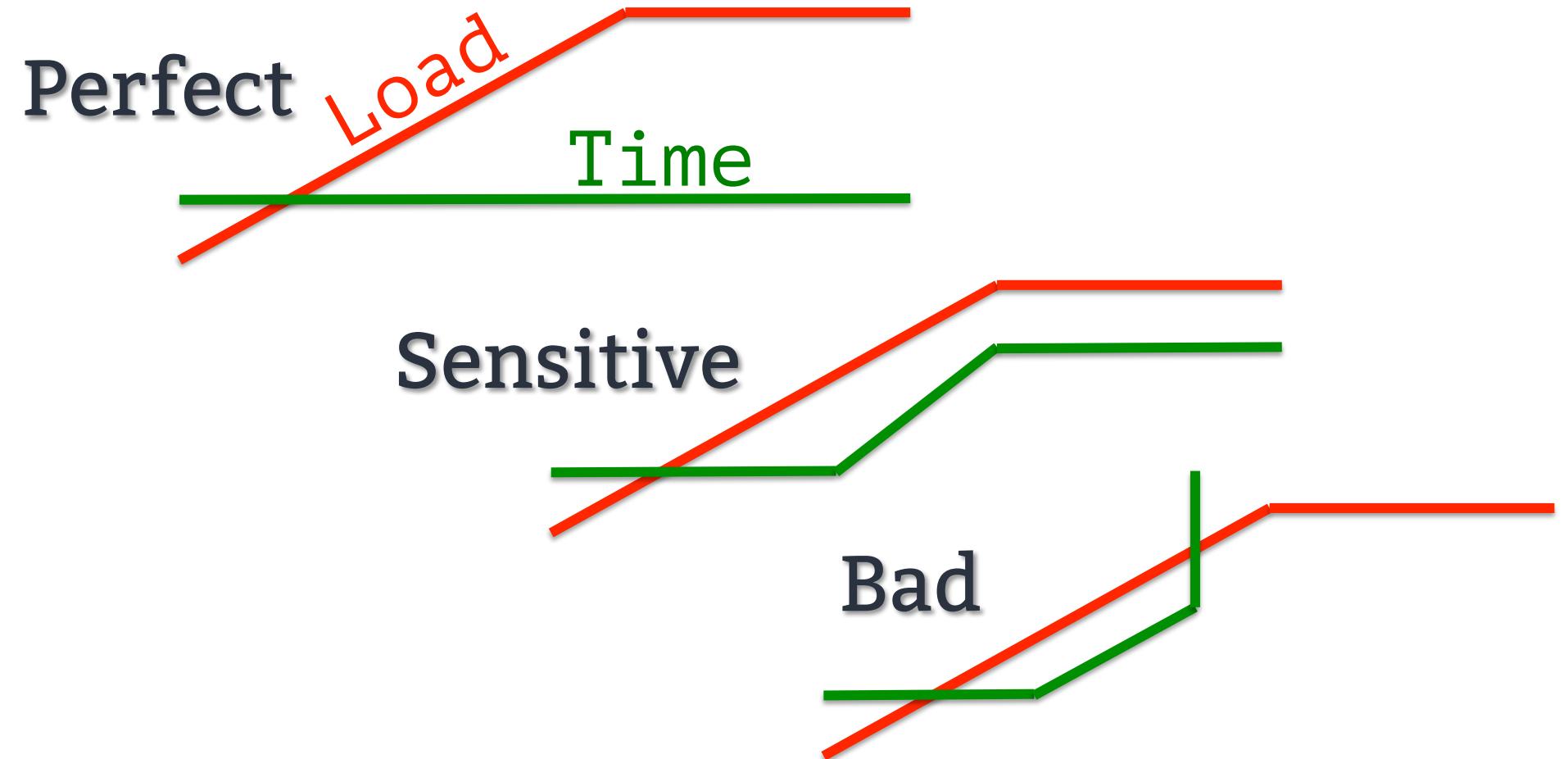
@alongir

What to look for?

The Load Sensitivity Point

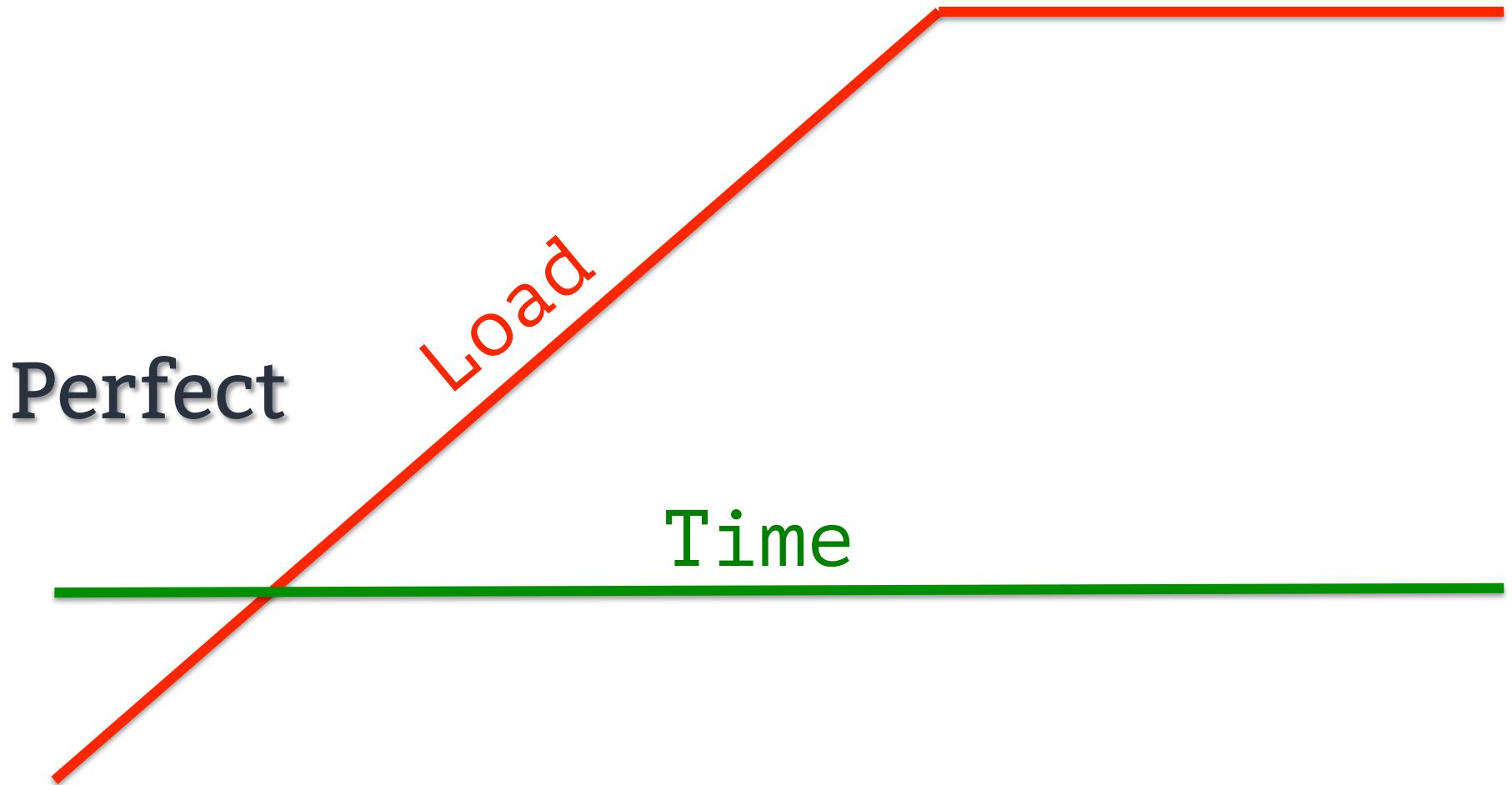


Load Performance Types



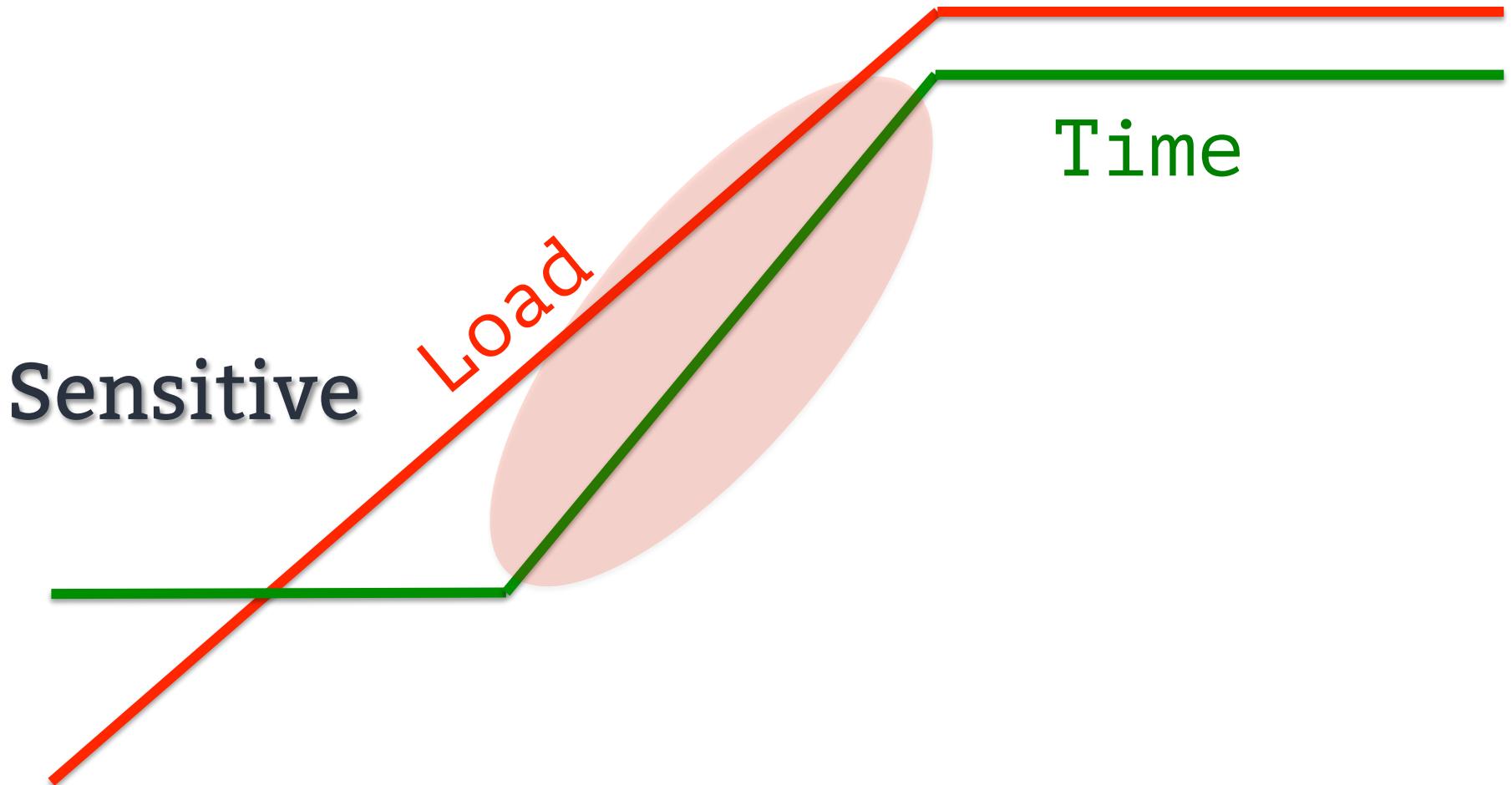
Load Performance Types

The Perfect Load



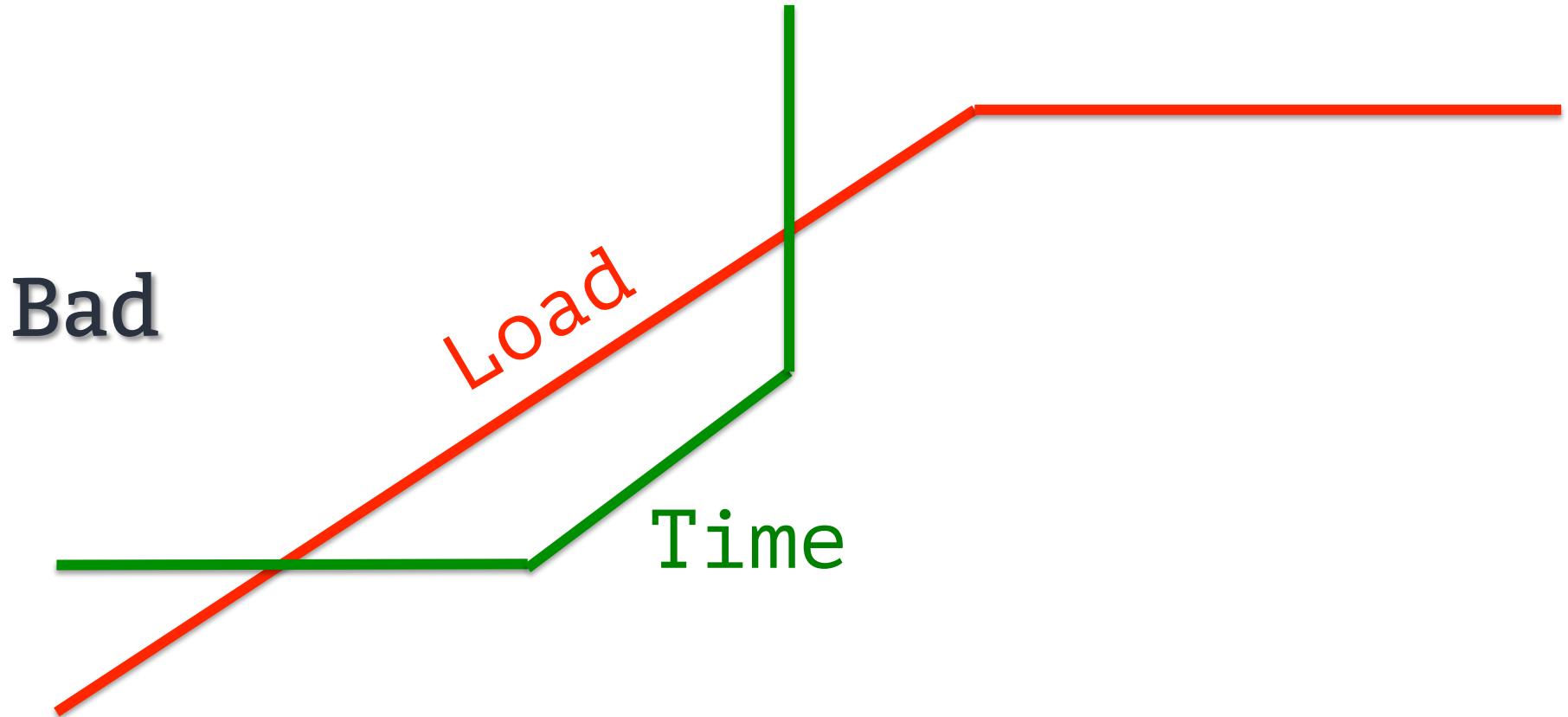
Load Performance Types

Sensitive to Load



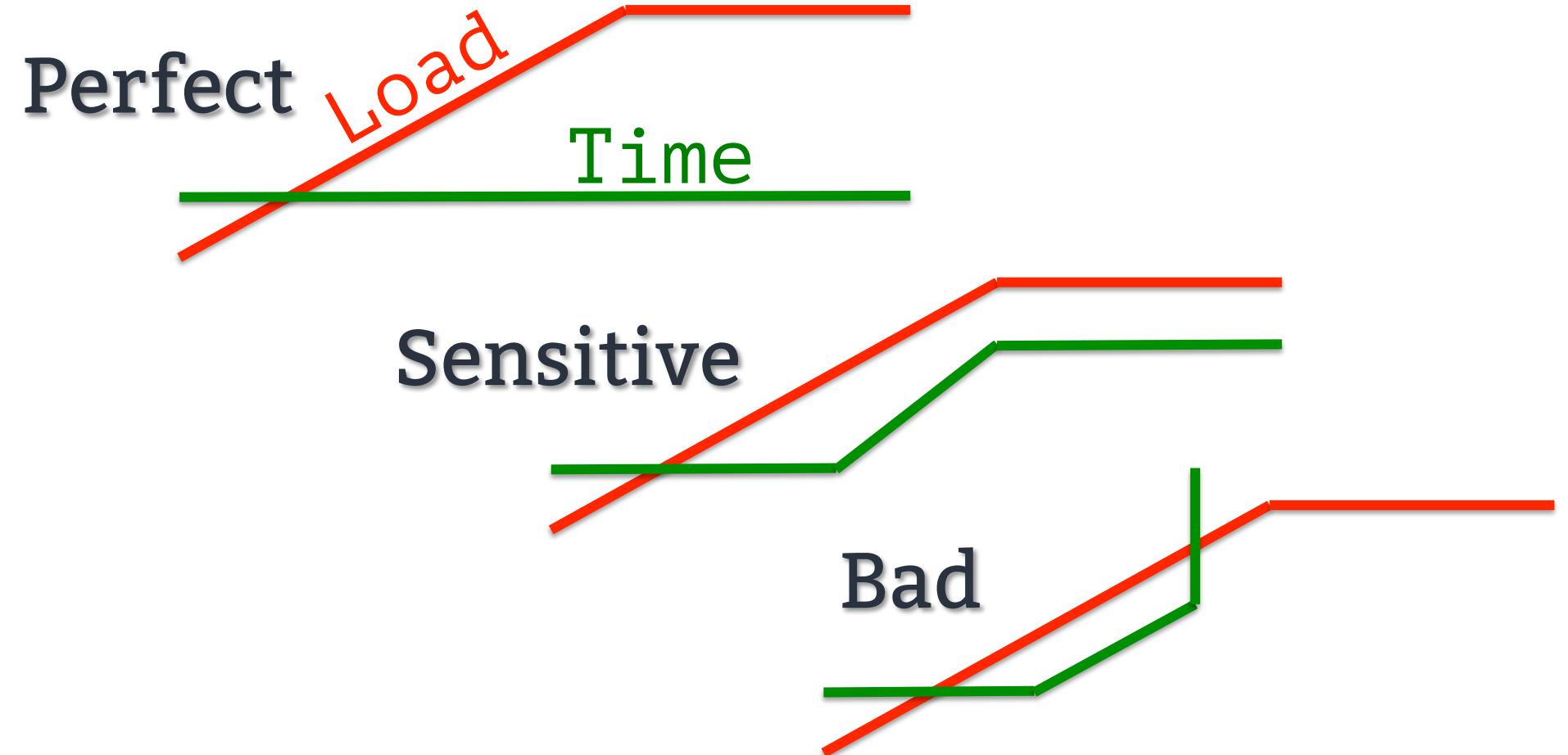
Load Performance Types

Bad Performance



Load Performance Types

Here They Are Again



Thank you for your time!

Thank you for attending.

We'll send out an email with the recording and the deck.

Q & A

For more information about Performance Testing and JMeter:

<http://BlazeMeter.com/blog/>

support@blazemeter.com

@BlazeMeter



Jolt Awards 2013:
The Best Testing Tools



CRN
EMERGING VENDORS
2012



785 Castro Street, Mountain View, CA 94041 | 1.855.445.2285 | info@blazemeter.com



@alongir