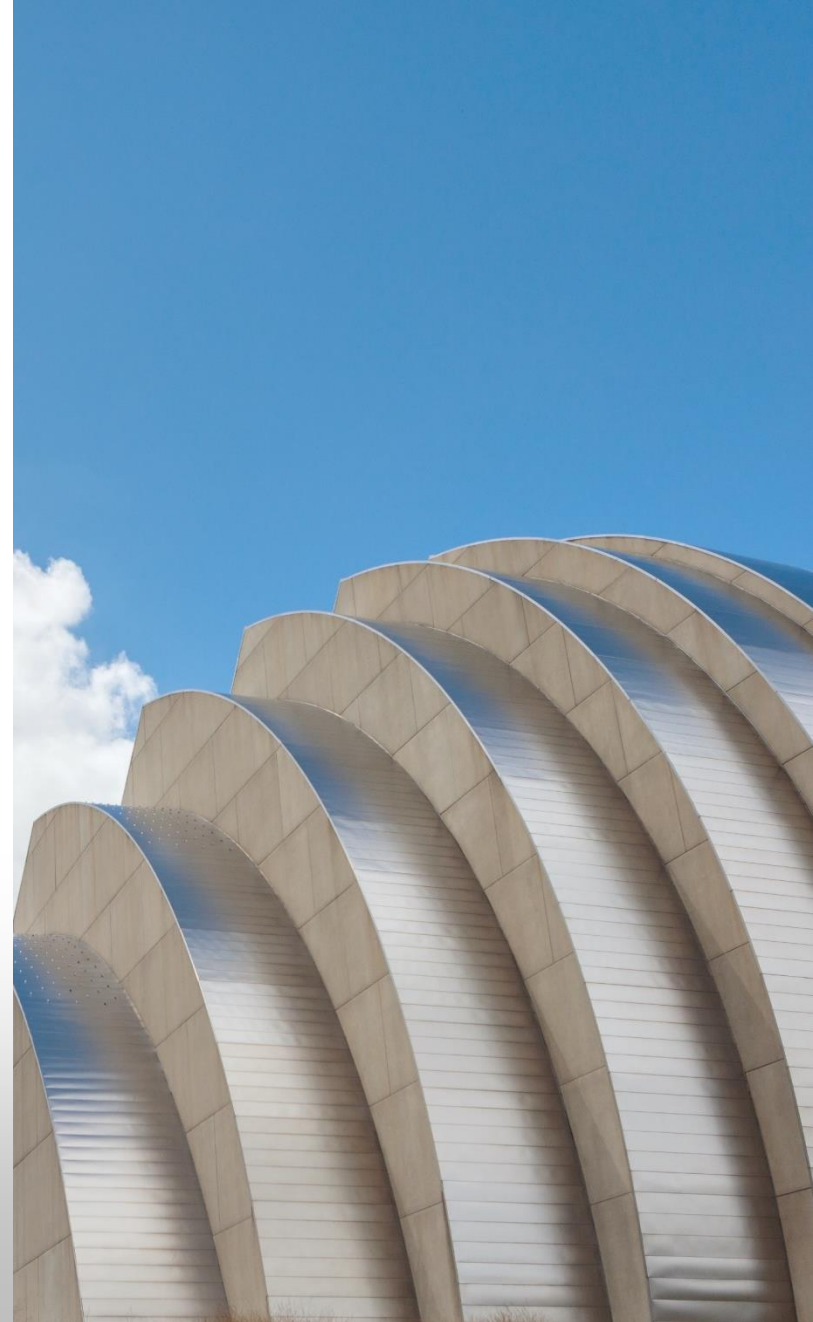


Open Source Performance

BlazeMeter Taurus

Nick Gover
Sr. Software Quality Engineer

10-23-2019





Why Performance Testing?

Why Performance Testing is Necessary

- To identify performance bottlenecks
- To monitor application's performance under load

Apache JMeter Recap

- Open source performance testing tool
- 100% pure Java application
- Most used tool, extremely popular, easy to use



What's the Problem?

JMeter Shortcomings

- Test result viewing conundrum
 - To analyze results of your test, you need the UI.
 - To run your test without exhausting your machine's memory, you shouldn't use the UI

Introducing Taurus

aka BZT Taurus

About Taurus

- Open Source Project
- Text-based configuration files
- Can run existing tests or defined in a DSL config
- Executed from the Command Line
- Can run locally or in the cloud (BlazeMeter)
- Can generate reports in BlazeMeter

Getting Started

- Prerequisites
 - Python (if installing manually)
 - Homebrew*
- A text editor of your choice
 - One that can format/lint YAML and/or JSON
- Install
 - Windows, Mac OS and Linux installation [instructions](#)
 - This is a second-level bullet placeholder; tab for indent

Tests as Configuration

- Taurus defines execution or new tests in it's own DSL
 - JSON
 - YAML
- Configuration as Python Dictionary
- Can create modular configuration scripts
 - Can include other Taurus configuration files

More about Taurus Configs

- Taurus defines execution or new tests in it's own DSL
 - JSON
 - YAML
- Can create modular configuration scripts
 - Can include other Taurus configuration files
- Taurus will recursively merge into resulting configuration any included-configs

bzt-rc

- ~/.bzt-rc
- Contained in user's home directory
 - Holds user preferences
 - Best practice to overwrite here first

JSON or YAML?

- Whichever you want, no one cares
- JSON is a subset of YAML
- Can use either JSON or YAML for your configs
- Mix and match

Command Line Tool

- Named ``bzt``, is a first rate citizen of CLI
- Consumes YAML/JSON configuration files
- Invoked by ``bzt <options> [configs]``
- URL shorthand capable
 - Simply provide a URL ``bzt http://blazedemo.com/``

Tools Taurus is Compatible With

- JMeter
- Gatling
- Selenium
- JUnit, TestNG
- PyTest, Locust
- Postman, SoapUI
- More...



Tests: JMeter vs Taurus

Writing Tests: JMeter vs Taurus

- JMX Can get very large quickly
 - Easily corrupted when multiple testers are editing
- Build tests programmatically
 - No need to open a different GUI
- Version Control friendly
 - See differences between commits

JMeter Test – XML vs YAML

```
Previous_Meter_Script.jmx
1 [J]ml version="1.0" encoding="UTF-8"
2 <jmeterTestPlan version="1.2" properties="5.0" jmeter="5.1.1 r1855137">
3   <hashTree>
4     <testPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test Plan" enabled="true">
5       <stringProp name="TestPlan.comments"></stringProp>
6       <boolProp name="TestPlan.functional_mode">false</boolProp>
7       <boolProp name="TestPlan.serialize_threadgroups">false</boolProp>
8       <elementProp name="TestPlan.user_defined_variables" elementtype="Arguments" guiclass="ArgumentsPanel" testclass="Arguments" testname="User Defined Variables" enabled="true">
9         <collectionProp name="Arguments.arguments">
10           <elementProp>
11             <stringProp name="TestPlan.user_define_classpath"></stringProp>
12           </elementProp>
13         </collectionProp>
14       </testPlan>
15     <hashTree>
16       <arguments guiclass="ArgumentsPanel" testclass="Arguments" testname="User Defined Variables" enabled="true">
17         <collectionProp name="Arguments.arguments">
18           <elementProp name="host" elementtype="Argument">
19             <stringProp name="Argument.name">host</stringProp>
20             <stringProp name="Argument.value">www.example.com</stringProp>
21             <stringProp name="Argument.metadata"></stringProp>
22           </elementProp>
23           <elementProp name="scheme" elementtype="Argument">
24             <stringProp name="Argument.name">scheme</stringProp>
25             <stringProp name="Argument.value">https</stringProp>
26             <stringProp name="Argument.metadata"></stringProp>
27           </elementProp>
28           <elementProp name="ccNumber" elementtype="Argument">
29             <stringProp name="Argument.name">ccNumber</stringProp>
30             <stringProp name="Argument.value">378563784760272</stringProp>
31             <stringProp name="Argument.metadata"></stringProp>
32             <stringProp name="Argument.desc">The 16 digit credit card number</stringProp>
33           </elementProp>
34           <elementProp name="ccExpYear" elementtype="Argument">
35             <stringProp name="Argument.name">ccExpYear</stringProp>
36             <stringProp name="Argument.value">${_time[${_time(yyyy)},ccExpYear]}</stringProp>
37             <stringProp name="Argument.metadata"></stringProp>
38             <stringProp name="Argument.desc">Current expiration year</stringProp>
39           </elementProp>
40         </collectionProp>
41       </arguments>
42     </hashTree>
43   </jmeterTestPlan>
44   <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement" testname="HTTP Request Defaults" enabled="true">
45     <elementProp name="HTTPSampler.Arguments" elementtype="Arguments" guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables" enabled="true">
46       <collectionProp name="Arguments.arguments">
47         <elementProp>
48           <stringProp name="HTTPSampler.domain">www.blazdemo.com</stringProp>
49           <stringProp name="HTTPSampler.port"></stringProp>
50           <stringProp name="HTTPSampler.protocol"></stringProp>
51           <stringProp name="HTTPSampler.contentEncoding"></stringProp>
52           <stringProp name="HTTPSampler.path"></stringProp>
53         </elementProp>
54       </collectionProp>
55     </elementProp>
56   </ConfigTestElement>
57 </hashTree>
58 </jmeterTestPlan>
```

Not as human readable

```
1 combined.yaml
2 modules:
3   nase:
4     ignore-unknown-actions: true
5
6 execution:
7   - executor: selenium
8     scenario: SIMPLE_TRAVEL_AGENCY-Selenium
9     blazegrid: true
10
11 capabilities:
12   browsername: chrome
13   hold-for: 20m
14
15 - executor: jmeter
16   scenario: SIMPLE_TRAVEL_AGENCY
17   concurrency: 20
18   ramp-up: 1m
19   hold-for: 3m
20
21 scenarios:
22   SIMPLE_TRAVEL_AGENCY:
23     headers:
24       User-Agent: 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.120 Safari/537.36'
25     requests:
26       - transaction: test
27         force-parent-sample: true
28         do:
29           - url: 'http://www.blazdemo.com/index.php'
30             method: GET
31             label: 'http://www.blazdemo.com/index.php'
32             think-time: 0ms
33             headers:
34               Upgrade-Insecure-Requests: '1'
35             Accept: 'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3'
36           - url: 'http://ajax.googleapis.com/ajax/libs/jquery/2.0.0/jquery.min.js'
37             method: GET
38             label: 'http://ajax.googleapis.com/ajax/libs/jquery/2.0.0/jquery.min.js'
39             headers:
40               Accept: '*/*'
41           - url: 'http://www.blazdemo.com/assets/bootstrap-table.js'
42             method: GET
43             label: 'http://www.blazdemo.com/assets/bootstrap-table.js'
44             headers:
45               Accept: '*/*'
46           - url: 'http://www.blazdemo.com/assets/bootstrap-table.js'
47             method: GET
48             label: 'http://www.blazdemo.com/assets/bootstrap-table.js'
49             headers:
50               Accept: '*/*'
51           - url: 'https://js.stripe.com/v2/stripe.min.js'
52             method: GET
53             label: 'https://js.stripe.com/v2/stripe.min.js'
54             headers:
55               Accept: '*/*'
```

Human Readable Configuration

Apache JMeter with Taurus

- Taurus comes bundled with JMeter
 - If you already have it installed you can ignore or delete
 - Need to let Taurus know where JMeter is installed

Convert Existing JMX scripts to YAML

- Simply run the command below
 - > `jmx2yaml <filename>.jmx`
- Will convert it to a Taurus formatted YAML config



Taurus for CI/CD

Performance Testing Challenges Continuous Testing

- Requirement of more complete environments
- Runs a lot longer
 - Extending the cycle time to complete
- Difficult to automate reporting analysis
- Debugging
 - Need to quickly retry

Fitting Performance Testing into Continuous Testing

- Develop performance tests earlier
- Make them modular
- Run automatically
 - As part of automated pipeline
 - Nightly

JMeter Demo

Run your existing JMeter Scripts

Advanced Config Demo

Run tests defined in pure configuration

Combine Test Executors Demo

Combine and run tests from multiple tools together



Scaling and Reporting

Cloud Scaling

- Taurus can also execute tests remotely
- Using JMeter distributed mode
- Using BlazeMeter cloud
 - This is a second-level bullet placeholder; tab for indent

What is BlazeMeter?

- Continuous Testing platform is
- 100% Open Source Compatible & Enterprise Ready
- 100% JMeter-compatible, SaaS web load testing cloud.
- Full REST API

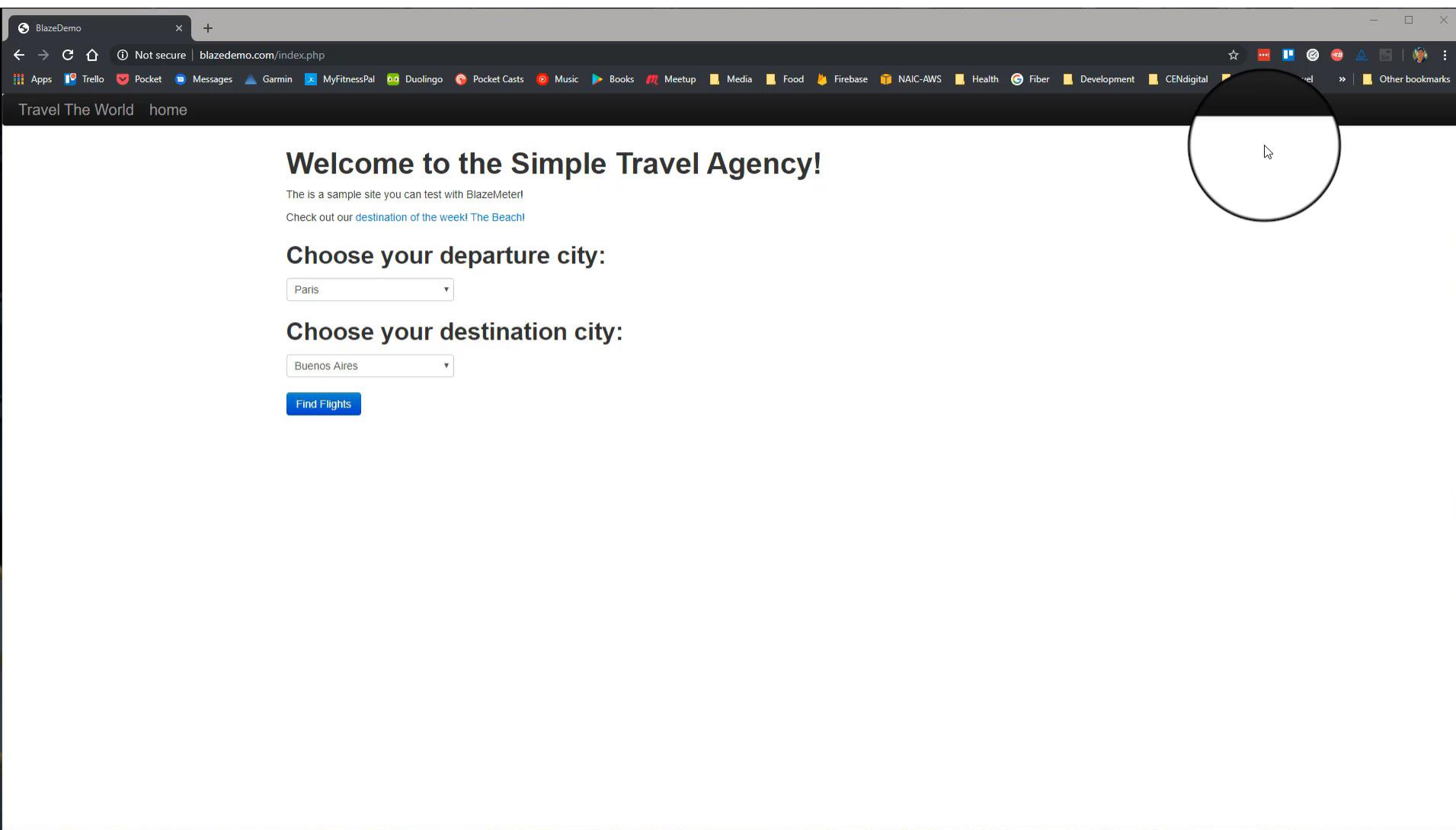


Bootstrapping Tests

BlazeMeter Recorder

- Chrome Extension
 - This is a second-level bullet placeholder; tab for indent
- Record JMeter Scripts
 - Export as .jmx, .yaml
 - Export as Selenium

Recording with BlazeMeter



Using Recordings with Taurus

- Chrome Extension
 - This is a second-level bullet placeholder; tab for indent
- Record JMeter Scripts
 - Export as .jmx, .yaml
 - Export as Selenium

Cloud Execution Demo

Run and scale your tests “instantly”



Questions?