



Авторы: Сарычев
Артём
Воробьёв Егор

Инструмент для НТ
Gatling

Что и откуда?

- Страна разработчик: Франция
- Компания: Gatling Corp
- Продукты: Gatling и Gatling Frontline
- Язык разработки: Scala
- Поддержка выполнения
Java кода



Первый взгляд

Запись скрипта

Gatling
STRESS TOOL

Recorder mode: HTTP Proxy

Network
Listening port*: localhost HTTP/HTTPS 8000 HTTPS mode: Self-signed Certificate
Outgoing proxy: host: HTTP HTTPS Username Password

Simulation Information
Package: Class Name*: RecordedSimulation
 Follow Redirects? Infer html resources?
 Remove cache headers? Save & check response bodies? Automatic Referers?

Output
Output folder*: /home/egor/Зарпункты/gatling-charts-highcharts-bundle-2.3.1/user-files/simulations **Browse**
Encoding: Unicode (UTF-8)

Filters
java regular expressions that matches the entire URI Strategy Disabled

Whitelist Blacklist

+ - Clear + - Clear No static resources

Save preferences **Start !**

Gatling Recorder - Running...

Controls
Tag: Add Clear Cancel Stop & Save

Executed Events
PAUSE 149ms
GET | http://detectportal.firefox.com/success.txt
GET | http://detectportal.firefox.com/success.txt
PAUSE 108ms
GET | http://detectportal.firefox.com/success.txt
PAUSE 242ms

Summary
Body

Response Information
Summary
Body

quiring accepting a certificate:

Сценарий

```
1  import io.gatling.http.Predef._
2  import io.gatling.core.Predef._
3
4
5  object Requests {
6    val search = exec(http("Search")
7      .get("search?string=core+i7&rootCategory=&sort=6")
8      .formParam("text", "ралли дакар")
9      .check(status is 200))
10
11
12   val clothes = exec(http("Clothes")
13     .get("catalog/verhnaa_odezda_7")
14     .check(status is 200))
15
16   val mobile = exec(http("Mobile")
17     .get("catalog/mobiles_gps_mp3")
18     .check(status is 200))
19
20   val office = exec(http("Office")
21     .get("catalog/office networks")
22     .check(status is 200))
23
24 }
25
```

Симуляция

```
1 import Requests._
2 import io.gatling.http.Predef._
3 import io.gatling.core.Predef._
4 import io.gatling.core.structure.ScenarioBuilder
5 import io.gatling.http.Predef.http
6 import io.gatling.http.protocol.HttpProtocolBuilder
7
8 import scala.concurrent.duration._
9 import scala.language.postfixOps
10
11
12 class LoadTest extends Simulation{
13   val httpConf = http
14     .baseUrl("https://www.ulmart.ru/")
15     .acceptHeader("text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8")
16     .acceptEncodingHeader("gzip, deflate, br")
17     .acceptLanguageHeader("ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7")
18     .userAgentHeader("Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.186 Safari/537.36")
19     .check(status is 200)
20
21   val scn = scenario("BaseLoad") randomSwitch(
22     (25, search),
23     (25, clothes),
24     (25, mobile),
25     (25, office)
26   )
27
28   setUp(scn.inject(constantUsersPerSec(100) during(10 seconds)).protocols(httpConf).assertions(Seq(
29     global.responseTime.percentile3.between(5,10)
30   )))
31 }
32
```

Подробнее о Gatling Frontline

| Версия | Gatling FrontLine Developer | Gatling FrontLine Professional | Gatling FrontLine Enterprise | Gatling-highcharts |
|-----------------------|-----------------------------------|-----------------------------------|-----------------------------------|--------------------------------|
| Локальная установка | 1 локальная установка | 1 локальная установка | 1 локальная установка | 1 локальная установка |
| Доступность Функций | Доступ ко всем функциям FrontLine | Доступ ко всем функциям FrontLine | Доступ ко всем функциям FrontLine | --- |
| Обучение | Обучение не включено | Обучение не включено | 12 часов обучения | Обучение не включено |
| Установка | Установка на месте не включена | Установка на месте не включена | Включена установка на месте` | Установка на месте не включена |
| Хранение симуляций | 2 | 8 | Неограниченно | Неограниченно |
| Распределенный запуск | да | да | да | нет |
| Лицензирование | proprietary | proprietary | proprietary | OpenSource |

Что нового и интересного?

- **Отличная поддержка протоколов HTTP (S), а также может использоваться для тестирования нагрузки JDBC и JMS**
- Автоматический HTML-отчет
- Асинхронный движок на базе Akka
- **Scala-** основанные скрипты
- Кроссплатформенность (скрипты выполняются на JVM)

Пример сценария

```
1 import Requests._
2 import io.gatling.http.Predef._
3 import io.gatling.core.Predef._
4 import io.gatling.core.structure.ScenarioBuilder
5 import io.gatling.http.Predef.http
6 import io.gatling.http.protocol.HttpProtocolBuilder
7
8 import scala.concurrent.duration._
9 import scala.language.postfixOps
10
11
12 class LoadTest extends Simulation{
13   val httpConf = http
14     .baseUrl("https://www.ulmart.ru/")
15     .acceptHeader("text/html,application/xhtml+xml,application/xml")
16     .acceptEncodingHeader("gzip, deflate, br")
17     .acceptLanguageHeader("ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7")
18     .userAgentHeader("Mozilla/5.0 (Windows NT 6.1; Win64; x64) App")
19     .check(status is 200)
20
21   val scn = scenario("BaseLoad") randomSwitch(
22     (25, search),
23     (25, clothes),
24     (25, mobile),
25     (25, office)
26   )
27
28   setUp(scn.inject(constantUsersPerSec(100) during(10 seconds)).protocols(httpConf).assertions(Seq(
29     global.responseTime.percentile3.between(5,10)
30   )))
31 }
32
```

```
1 import io.gatling.http.Predef._
2 import io.gatling.core.Predef._
3
4
5 object Requests {
6   val search = exec(http("Search")
7     .get("search?string=core+i7&rootCategory=&sort=6")
8     .formParam("text", "ралли дакар")
9     .check(status is 200))
10
11
12   val clothes = exec(http("Clothes")
13     .get("catalog/verhnaa_odezda_7")
14     .check(status is 200))
15
16   val mobile = exec(http("Mobile")
17     .get("catalog/mobiles_gps_mp3")
18     .check(status is 200))
19
20   val office = exec(http("Office")
21     .get("catalog/office_networks")
22     .check(status is 200))
23
24 }
25
```

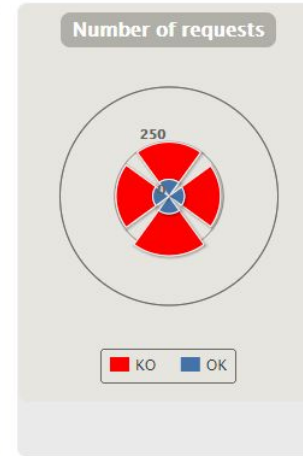

Результат

```
=====
2018-03-27 11:07:17                                     14s elapsed
---- BaseLoad ----
[#####]100%
      waiting: 0      / active: 0      / done:1000
---- Requests ----
> Global                                               (OK=302   KO=698   )
                                                    (OK=80    KO=193   )
                                                    (OK=71    KO=165   )
                                                    (OK=77    KO=171   )
                                                    (OK=74    KO=169   )
=====
---- Global Information ----
> request count          1000 (OK=302   KO=698   )
> min response time      53   (OK=679   KO=53    )
> max response time      6495 (OK=6495  KO=4173  )
> mean response time     2707 (OK=3404  KO=2406  )
> std deviation          926   (OK=1017  KO=693   )
> response time 50th percentile 2705 (OK=3341  KO=2497  )
> response time 75th percentile 3199 (OK=4084  KO=2898  )
> response time 95th percentile 4304 (OK=5128  KO=3362  )
> response time 99th percentile 5310 (OK=5845  KO=3812  )
> mean requests/sec      71.429 (OK=21.571 KO=49.857)
----- Response Time Distribution -----
> t < 800 ms             2   ( 0%)
> 800 ms < t < 1200 ms  6   ( 1%)
> t > 1200 ms            294 (29%)
> failed                 698 (70%)
----- Errors -----
> status.find.is(200), but actually found 503          698 (100,0%)
=====
```

Автоматический HTML-отчет

- Active Users
- Requests / sec
- Responses / sec

> Global Information



STATISTICS Expand all groups | Collapse all groups

| Requests ^ | Executions | | | | Response Time (ms) | | | | | | | | |
|--------------------|------------|------|------|--------|--------------------|-------|------------|------------|------------|------------|-------|--------|-----------|
| | Total ↓ | OK ↓ | KO ↓ | % KO ↓ | Req/s ↓ | Min ↓ | 50th pct ↓ | 75th pct ↓ | 95th pct ↓ | 99th pct ↓ | Max ↓ | Mean ↓ | Std Dev ↓ |
| Global Information | 1000 | 302 | 698 | 70% | 71.429 | 53 | 2705 | 3199 | 4304 | 5310 | 6495 | 2707 | 926 |
| Mobile | 248 | 77 | 171 | 69% | 17.714 | 196 | 2623 | 3010 | 3820 | 4706 | 5846 | 2547 | 842 |
| Office | 236 | 71 | 165 | 70% | 16.857 | 163 | 2651 | 3095 | 4045 | 4882 | 6396 | 2604 | 863 |
| Clothes | 273 | 80 | 193 | 71% | 19.5 | 53 | 2814 | 3317 | 4467 | 5437 | 6495 | 2815 | 988 |
| Search | 243 | 74 | 169 | 70% | 17.357 | 594 | 2805 | 3325 | 4452 | 5368 | 5907 | 2849 | 957 |

ERRORS

| Error ↓ | Count ↓ | Percentage ↓ |
|---|---------|--------------|
| status.find.is(200), but actually found 503 | 698 | 100 % |

Сравнение с Jmeter

| Что сравниваем | Gatling | Jmeter |
|--------------------|---------------------------------|----------------------|
| Производительность | Асинхронный движок на базе Akka | Java - движок |
| Построение отчетов | html-report | есть |
| Тех.поддержка | есть | интернет |
| Цена | 5,9K – 15,9K EUR | open source software |

Сравнение с HP LR

| Что сравниваем | Gatling | HP LR |
|--------------------------|--|-------------------------|
| Производительность | На одинаковой задаче в 2 раза быстрее. | |
| Количество пользователей | Не ограничено | 50 |
| Построение отчетов | html-report | Html-report, PDF, |
| Тех.поддержка | Есть, в платной версии | Платная |
| Цена | Free version | Free version |

Конец

Спасибо за внимание!

