Gatling

Test. Succeed. Iterate.

Live Webinar & Office Hours #3 - Tuesday, April 30, 2024

Today's Presenters

Peter Dutka

Team Lead Customer Success

Agenda

- What's new?
- Introduction to Gatling Enterprise Reporting
- The Report Summary
- Requests
- Groups
- Users
- Connections
- DNS
- Load Generators
- Run History and Trends
- Live questions and answers

Gatling

What's New

What's new for Gatling

What's new this month

Package Descriptor

- Now easier to switch from Enterprise Self-Hosted to Cloud and upload tests to the Cloud

Run Title and Description at Launch

- Use your build tool or the API to add titles and descriptions to your runs on Enterprise Cloud

Gatling 3.11

- Check the documentation for a full list of changes and features.

Gatling

Enterprise Reporting

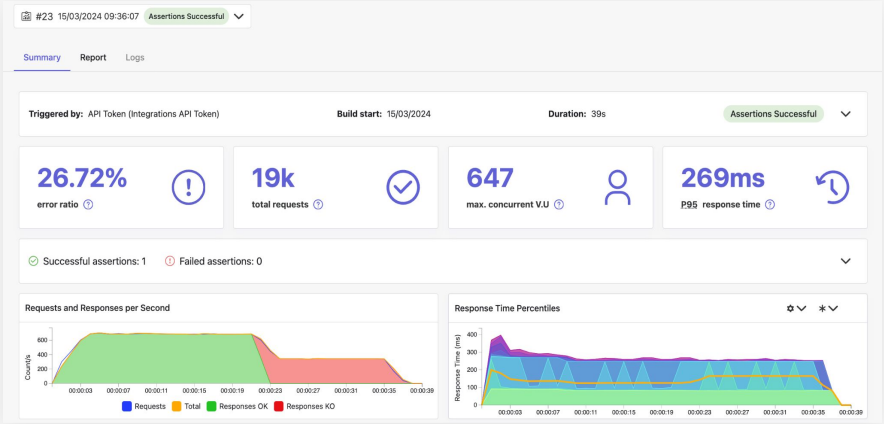
Advanced insights on your application

The Run Summary

A quick look at your test

Report Summary gives a high-level view of your run success including:

- Assertion success/failure
- Error ratio
- P95 response time



Requests

Dive into your response times

The Request Tab gives deep detail on your response times and errors by request.

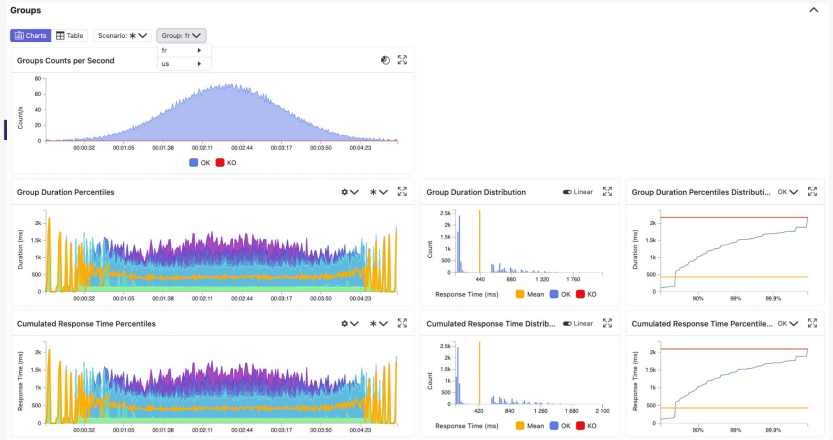
- Filter by Scenario, Group, and Request
- Filter graphs by status
- Switch from graph to table view
- Create markers that will transfer to all graphs throughout the report

Groups

Filter your data by groups and subgroups

The Groups tab allows you to view specific segments of your traffic.

- **Organization**: Groups help organize scenarios for better management.
- **Reporting**: They enable detailed performance insights for specific functionalities.
- **Debugging**: Groups aid in pinpointing and resolving issues faster.

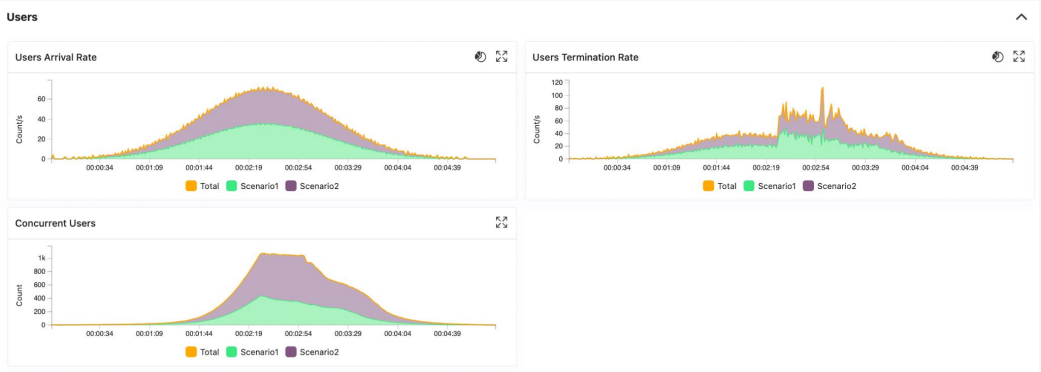


Users

View how users are arriving and leaving your app

The Users tab allow you to see how users:

- **Arrival**
- **Leave**
- **Concurrent users**

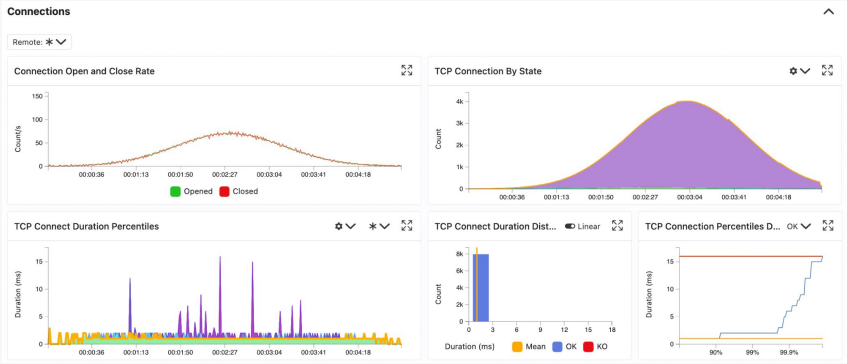


Connections Part 1

Advanced Metrics to pinpoint issues

The Connections tab contains the heart of the advanced statistics that drive Gatling Enterprise

- **Open and Close Rates**: Get insights on your scalability and resource utilization
- **TCP Durations**: Ensure that your users experience acceptable connection times.
- **Benchmarking**: Compare performance across different test scenarios or releases.

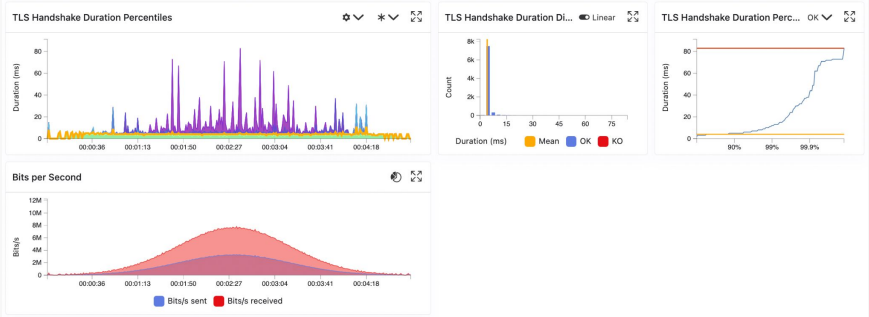


Connections Part 2

Advanced Metrics to pinpoint issues

The Connections tab contains the heart of the advanced statistics that drive Gatling Enterprise

- **TLS Duration:** Get insight on how TLS is effecting load times and help pinpoint bottlenecks
- **Bits per second (throughput):** Ensure that your system meets performance requirements, maintains reliability under load, and delivers a satisfactory user experience.

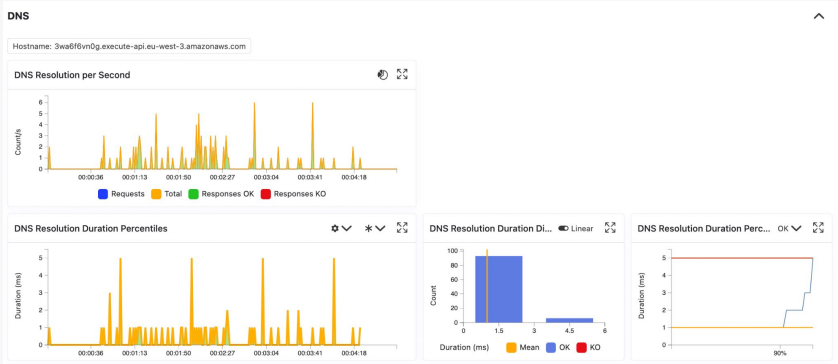


DNS

Get info on domain name systems

The DNS tab contains metrics specific to the domain naming system

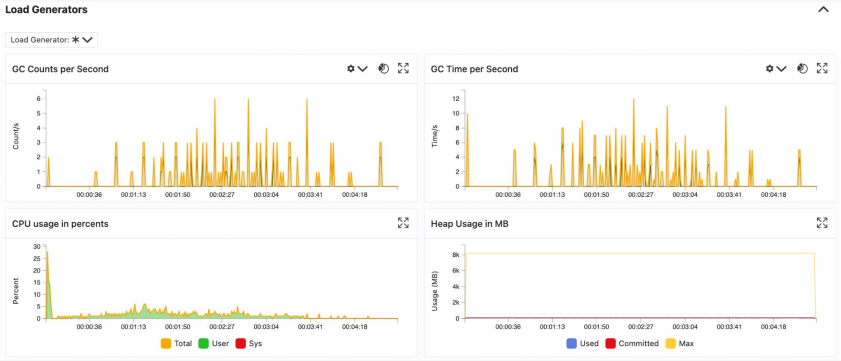
- **Get specific:** Find potential bottlenecks that are often overlooked.
- **Assess scalability:** Make sure that when you're ready to scale DNS resolution load times don't hold you back.



Optimize your load generation

The Load Generator tab lets you see how much CPU you're using and whether you need to add more or less

- **Check GC Counts:** Find memory leaks in your application
- **CPU usage:** Determine the correct number of load generators to use per test and ensure results aren't affected.

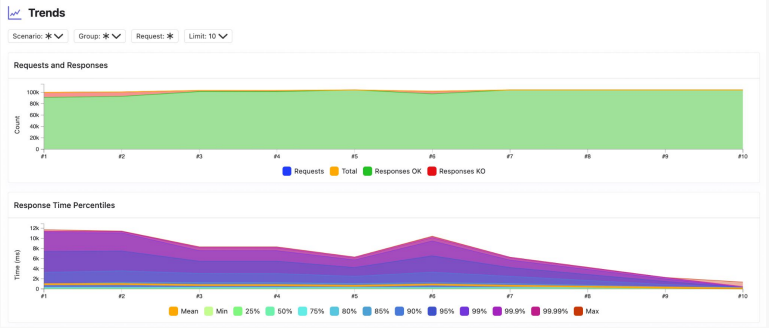


Run History and Trends

See your progress and and compare runs

Run History lets you view the progress of a simulation over multiple runs

- **Compare Runs:** Compare response times and error rates in requests between any two runs
- **Filter Trends:** Choose a scenario, group or request to see their progression over time.



Gatling

Coming Soon

What's up next?

GATLING SDK

Script in JavaScript and TypeScript

JavaScript SDK for JS developers including:

- npm and Yarn build tools
- External JS library support
- API and logic similar to existing Java SDK

```
import { core, http, runSimulation } from "@gatling/js";

const mySimulation = runSimulation((setUp) => {

  const baseHttpProtocol =
    http.baseUrl("https://computer-database.gatling.io");

  const scn = core.scenario("My scenario")
    .exec((session) => session.set("page", 1))
    .exec(http("Browse page 1").get("/computers?p=#{page}"));

  setUp(
    scn.injectOpen(core.constantUsersPerSec(2).during(30))
  ).protocols(baseHttpProtocol);
});

export default mySimulation;
```

JS

Live Q & A

You ask, we answer.

Now is your time to **submit your questions** and get answers!

All topics are open.

If we can't get to your question today, **we'll follow up by email** after the session.

Gatling

Thank you!

gatling.io