ajuna

From Zero to Monitored

Ajuna Kyaruzi

ajunaky

About Me

- Developer Relations at Datadog
- Mentor and Board Member for Africode
- Previous: SWE/SRE at Google
- I’m based in Atlanta, GA
- Originally from Dar es Salaam, Tanzania
- My favourite conference swag is socks!

ajunaky

ajuna

From Zero to ~~Monitored~~ Monitoring

ajunaky

Honestly Black Lives Matter
@honest_update

Our new monitoring product just watches Twitter and IRC for our name + "down".

Where to get started: Facets of Observability

- Logs
- Metrics
- Traces
- Application Performance Monitoring
- ....

Logging

Logging

↓ DATE	HOST	SERVICE	CONTENT
Mar 26 08:32:25.740	gke-demo-11287-us-prod-c…	web-store	Cart item 623f3209a8f9020014e73ce2 was successfully …
Mar 26 08:32:25.740	i-0540f67492f7bdc10	web-store	Mongoid::Errors::DocumentNotFound: message: Document…
Mar 26 08:32:25.739	i-009d015a9910ed94b	web-store	Cart item 623f3209cf77d7001583d0af was successfully …
Mar 26 08:32:25.739	gke-demo-11287-us-prod-w…	web-store	Shipping 2 items for checkout 623f3204d7dd7700147554…
Mar 26 08:32:25.737	gke-us-staging-default-p…	web-store	Request/Server info: SCRIPT_NAME='' QUERY_STRING='' …
Mar 26 08:32:25.737	vm-8b90b44e-a228-472b-54…	web-store	Getting ads from ads service with customer_id=623f30…
Mar 26 08:32:25.735	gke-demo-dpn-us-west-def…	web-store	Finding customer with session_id: FRCFLLOQSL
Mar 26 08:32:25.735	gke-demo-11287-us-prod-e…	web-store	Getting product recommendations for customer_id=623f…
Mar 26 08:32:25.735	i-0540f67492f7bdc10	web-store	Mongoid::Errors::DocumentNotFound: message: Document…
Mar 26 08:32:25.734	gke-demo-11287-us-prod-c…	web-store	Getting product recommendations for customer_id=623f…
Mar 26 08:32:25.734	gke-demo-dpn-us-west-def…	web-store	Getting product recommendations for customer_id=623f…
Mar 26 08:32:25.733	gke-dev-shopist-default-…	web-store	Finding cart for session_id: VWCGXXXTBN
Mar 26 08:32:25.733	gke-demo-11287-us-prod-w…	web-store	Request/Server info: SCRIPT_NAME='' QUERY_STRING='' …
Mar 26 08:32:25.733	gke-demo-11287-us-prod-w…	web-store	Finding cart for session_id: LNCCIZFVWV
Mar 26 08:32:25.733	gke-demo-11287-us-prod-c…	web-store	Finding cart for session_id: JBETAEDZTN
Mar 26 08:32:25.732	gke-demo-11287-us-prod-c…	web-store	Request/Server info: SCRIPT_NAME='' QUERY_STRING='' …
Mar 26 08:32:25.731	gke-dev-shopist-default-…	web-store	Creating new cart for customer: 623f31b9123dc6001378…

ajunaky

ERROR Mar 22, 2019 at 11:52:44.453 (a minute ago) View in Context

HOST
i-097d46b2533362651

SERVICE
java-app

SOURCE
java

TAGS

account:demo autoscaling_group:awseb-e-wbnjfa2afu-stack-awsebautoscalinggroup-1msseb7d5b2lh availability-zone:us-east-1a aws:cloudformation:logical-id:awsebautoscalinggroup aws:cloudformation:stack-id:arn:aws:cloudformation:us-east-1: :stack/awseb-e-wbnjfa2afu-stack/7de35e10-39d9-11e9-a179-1297d6a5b478 aws:cloudformation:stack-name:awseb-e-wbnjfa2afu-stack aws_account: cloud_provider:aws elasticbeanstalk:environment-id:e-wbnjfa2afu elasticbeanstalk:environment-name:javaloggingdemo-env-1 ...

```
Servlet.service() for servlet [dispatcherServlet] in context with path [] threw exception [Request processing failed; nested exception is
java.lang.StringIndexOutOfBoundsException: Index out of range] with root cause
```

LOGGER NAME
org.apache.catalina.core.ContainerBase.[Tomcat].[localhost].[/].[dispatcherServlet]

THREAD NAME
http-nio-8080-exec-8

ERROR KIND
java.lang.StringIndexOutOfBoundsException

ERROR MESSAGE

```
Index out of range
```

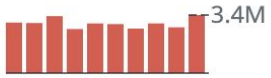
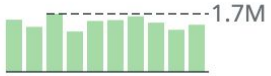
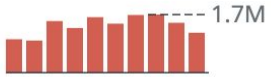
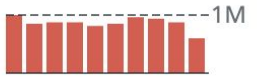
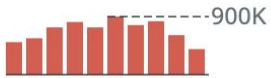
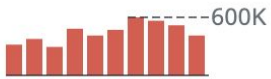
ERROR STACK

```
java.lang.StringIndexOutOfBoundsException: Index out of range
        at com.example.app.loggingApp.classOne.getResult(classOne.java:16)
        at com.example.app.loggingApp.AppController.defaultMessage(AppController.java:24)
        at sun.reflect.GeneratedMethodAccessor29.invoke(Unknown Source)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(InvocableHandlerMethod.java:209)
        at org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:136)
        at org.springframework.web.servlet.mvc.method.annotation.ServletInvocableHandlerMethod.invokeAndHandle(ServletInvocableHandlerMethod.java:102)
        at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.invokeHandlerMethod(RequestMappingHandlerAdapter.java:877)
        at org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter.handleInternal(RequestMappingHandlerAdapter.java:783)
        at org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter.handle(AbstractHandlerMethodAdapter.java:87)
        at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:991)
        at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:925)
        at org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:974)
        at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:866)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:635)
        at org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:851)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
        at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
        at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
        at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
```

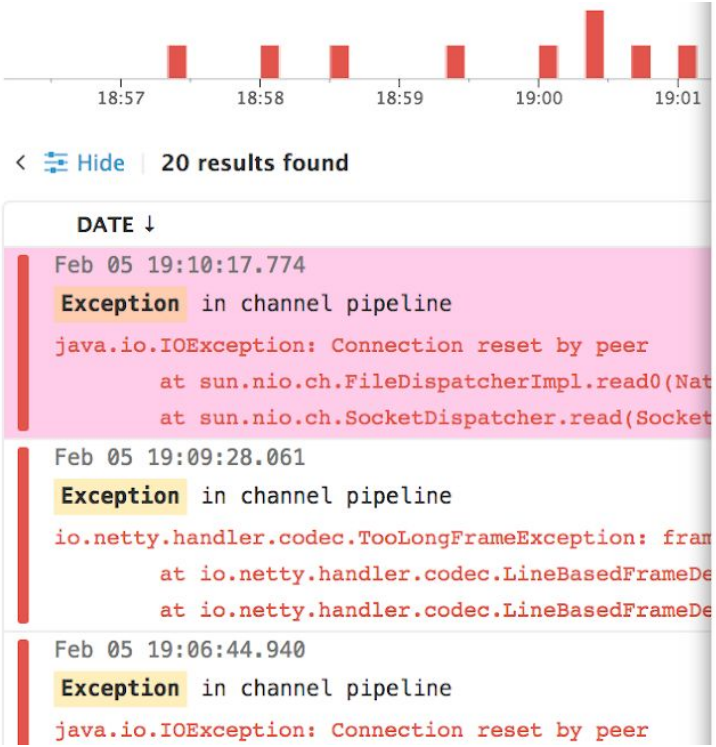
ajunaky

```
package com.foo;
import org.apache.logging.log4j.Logger;
import org.apache.logging.log4j.LogManager;

public class Bar {
  static final Logger logger = LogManager.getLogger(Bar.class.getName());

  public boolean doIt() {
    logger.entry();
    logger.error("Did it again!");
    return logger.exit(false);
  }
}
```

Example of adding logging with Log4J

COUNT	~VOLUME	STATUS	SERVICE	MESSAGE
≈29M		error	web-store	Error sending * notification
≈14.6M		ok	web-store	/*/-**-
≈13.6M		error	web-store	PaymentServiceUnavailableError: Payment service reported 503 Unavailable.
≈13.1M		error	web-store	Unable to connect to mongo, retrying in 0.001s
≈8.6M		error	web-store	Redis connection failed, requeueing shipping job.
≈7.4M		error	web-store	Error resolving location for the coordinates provided
≈7.1M		error	web-store	Payment rejected for transaction due to number of API calls per minute exceeding 1000
≈4.7M		error	web-store	Payment Service unavailable for <email redacted> for checkout *

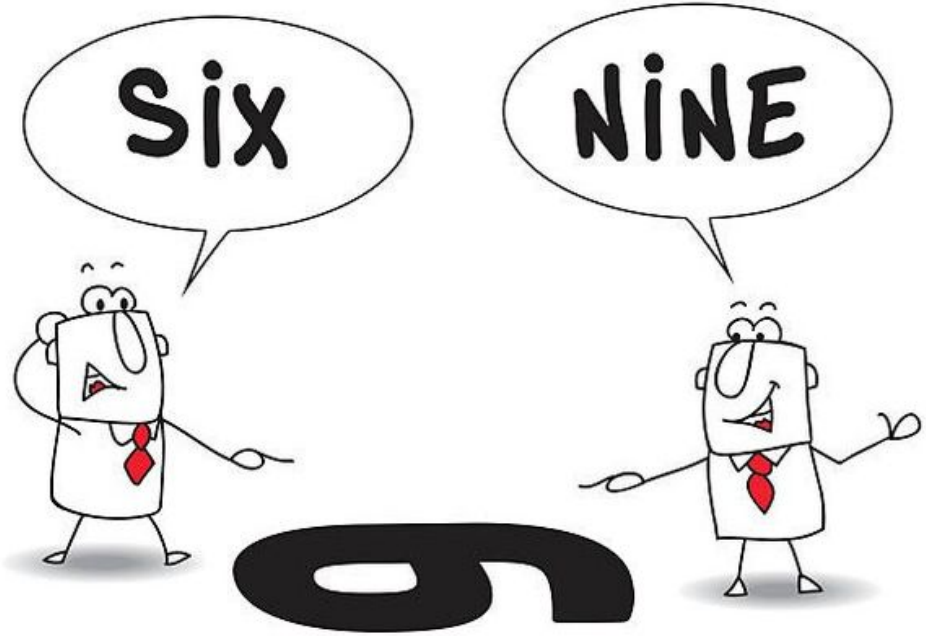
Alert on Logs



Metrics


METRICS
METRICS
GOAL
KEY
PLAN
DATA
GROWTH
IMPLEMENT
SALES
TARGET
IMPROVEMENT
BUSINESS STUDY VALUE DATA
DASHBOARD
INDICATOR
COMMUNICATION

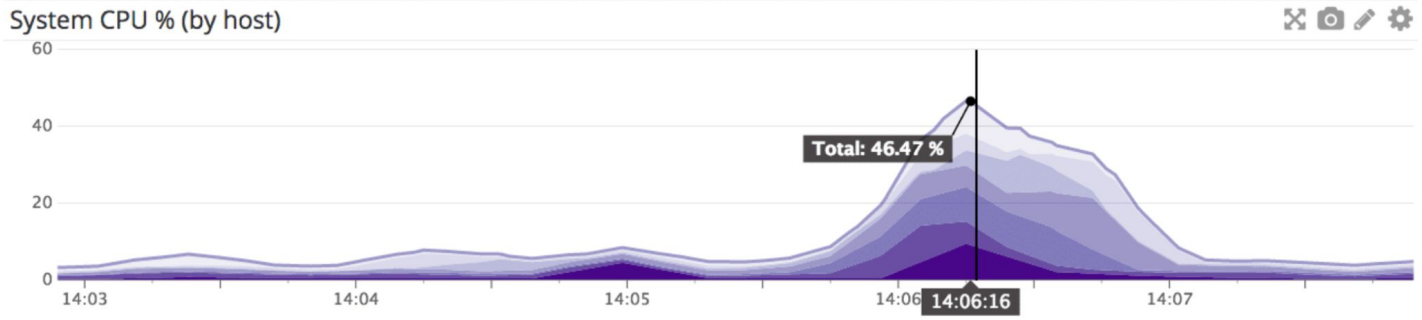
4 Qualities of Good Metrics



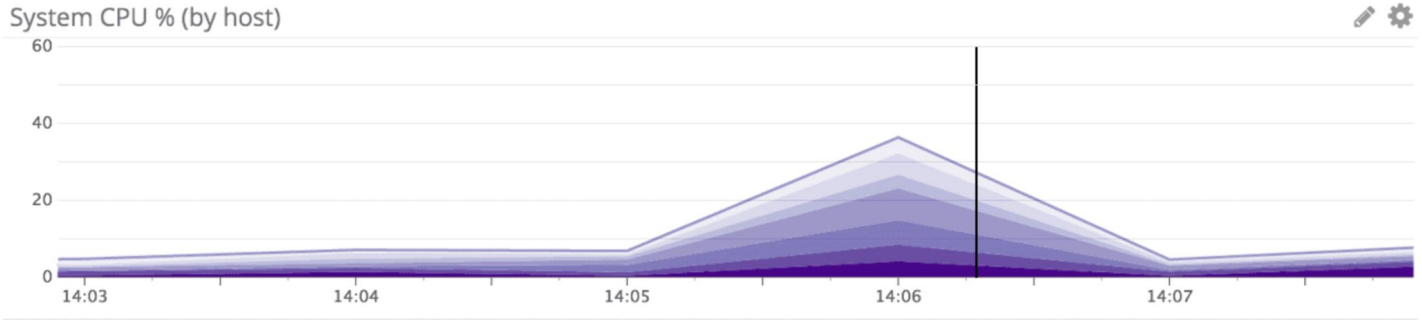
1. Well Understood

1	14	USA	JACKSON Erin	37.04	0.00
2	4	JPN	TAKAGI Miho	37.12	+0.08
3	13	ROC	GOLIKOVA Angelina	37.21	+0.17
4	4	AUT	HERZOG Vanessa	37.28	+0.24
5	7	NED	LEERDAM Jutta	37.34	+0.30
6	9	NED	KOK Femke	37.39	+0.35
7	10	KOR	KIM Minsun	37.60	+0.56
8	11	ROC	KACHANOVA Daria	37.65	+0.61

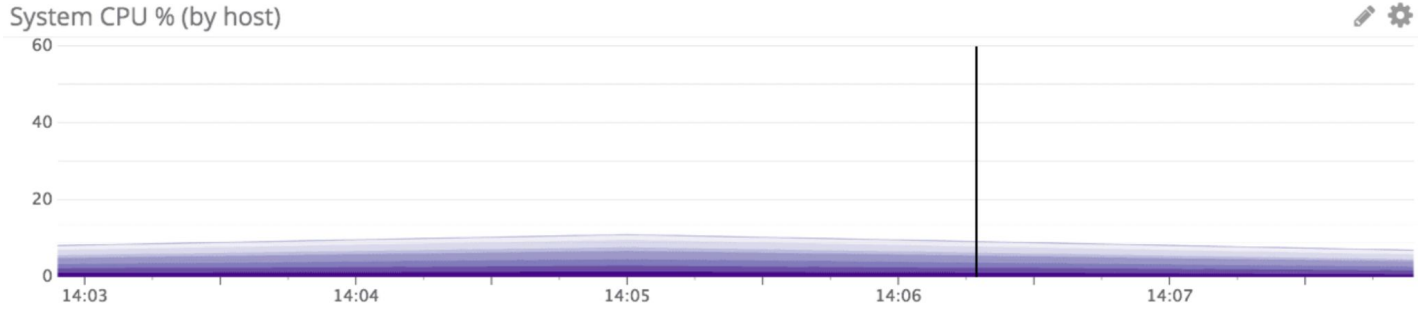
2. Sufficient Granularity



1 second
Peak 46%



1 minute
Peak 36%

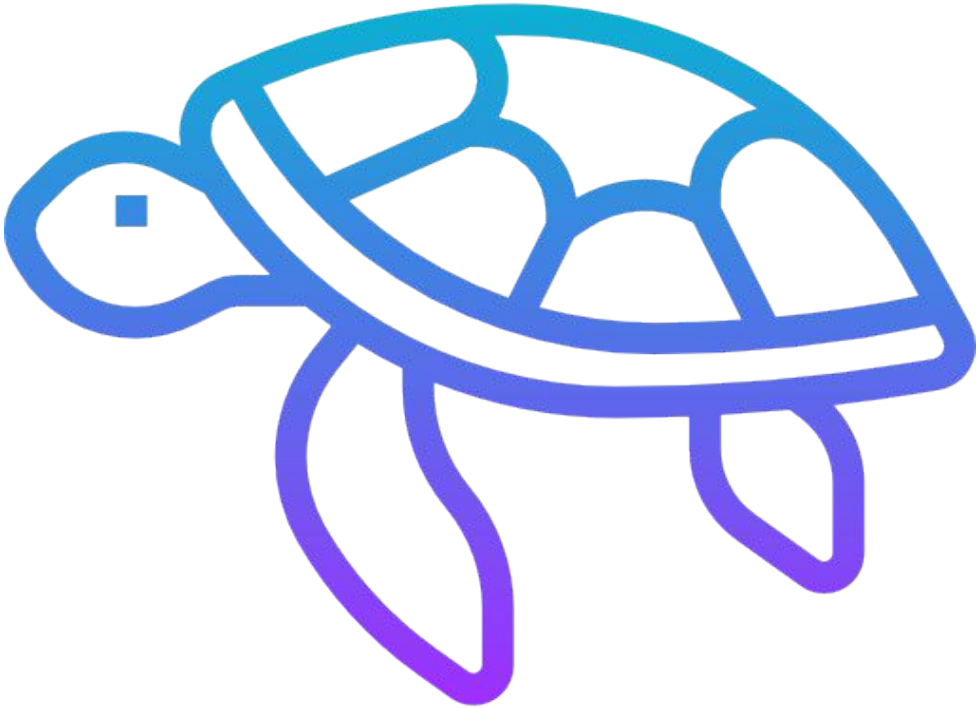


5 minutes
Peak 12%

ajunaky

3. Tagged and Filterable

Icons made by eucalyp from flaticon.com



4. Long Lived

METRIC TYPE	USE IT FOR...	EXAMPLES
Gauge	Measuring resource usage, capacity, etc. Values that can rise and fall, and that have fixed upper bounds	Size of a collection, number of running threads, number of messages on a queue, memory usage
Counter	Measuring a number of events or actions - a value that only increases, and never decreases.	Total number of orders processed, total tasks completed, etc.
Timer	Measuring short-duration events and their frequency	Method execution time, request duration, time taken to boil an egg.

```
import io.micrometer.core.instrument.MeterRegistry;
import io.micrometer.core.instrument.Tags;

import org.springframework.stereotype.Component;

@Component
public class MyBean {

    private final Dictionary dictionary;

    public MyBean(MeterRegistry registry) {
        this.dictionary = Dictionary.load();
        registry.gauge("dictionary.size", Tags.empty(),
                       this.dictionary.getWords().size());
    }

}
```

Example of adding metrics with Micrometer

Search APM | env:datad0g.com | Services | coffee-house

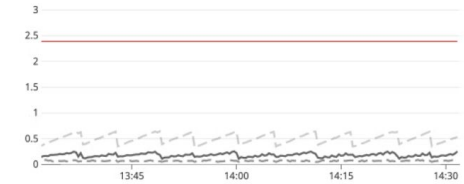
coffee-house

60 m | Jun 13, 1:32 pm - Jun 13, 2:32 pm

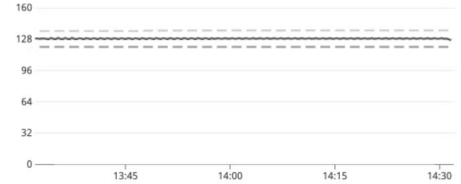
JVM Metrics

View integration dashboard

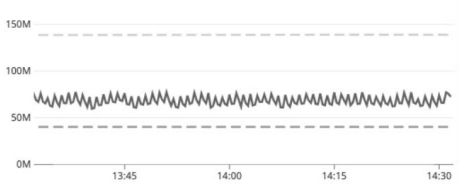
Heap Usage



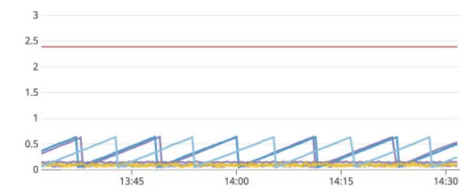
Non-Heap Usage



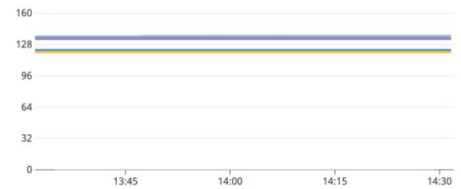
GC Old Gen Size



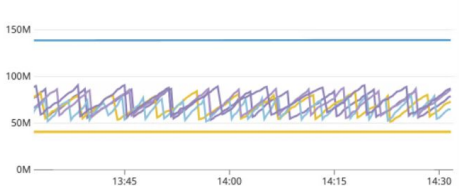
Heap Usage (top 10)



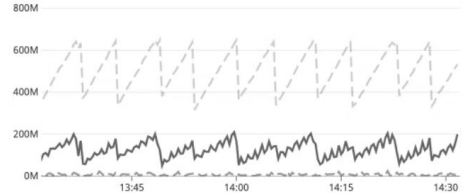
Non-Heap Usage (top 10)



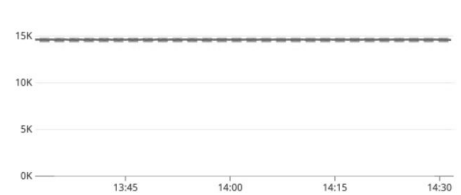
GC Old Gen Size (top 10)



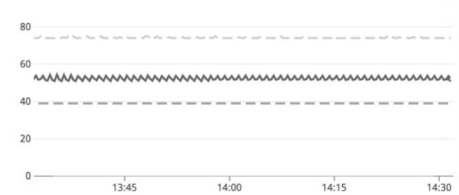
GC New Gen Size



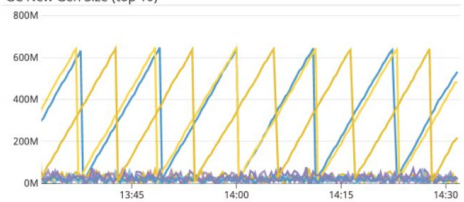
Number of Classes Loaded



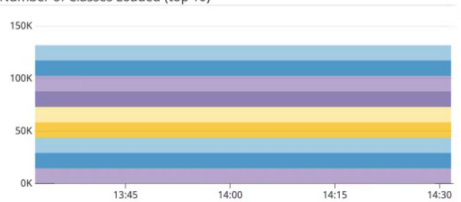
Thread Count



GC New Gen Size (top 10)



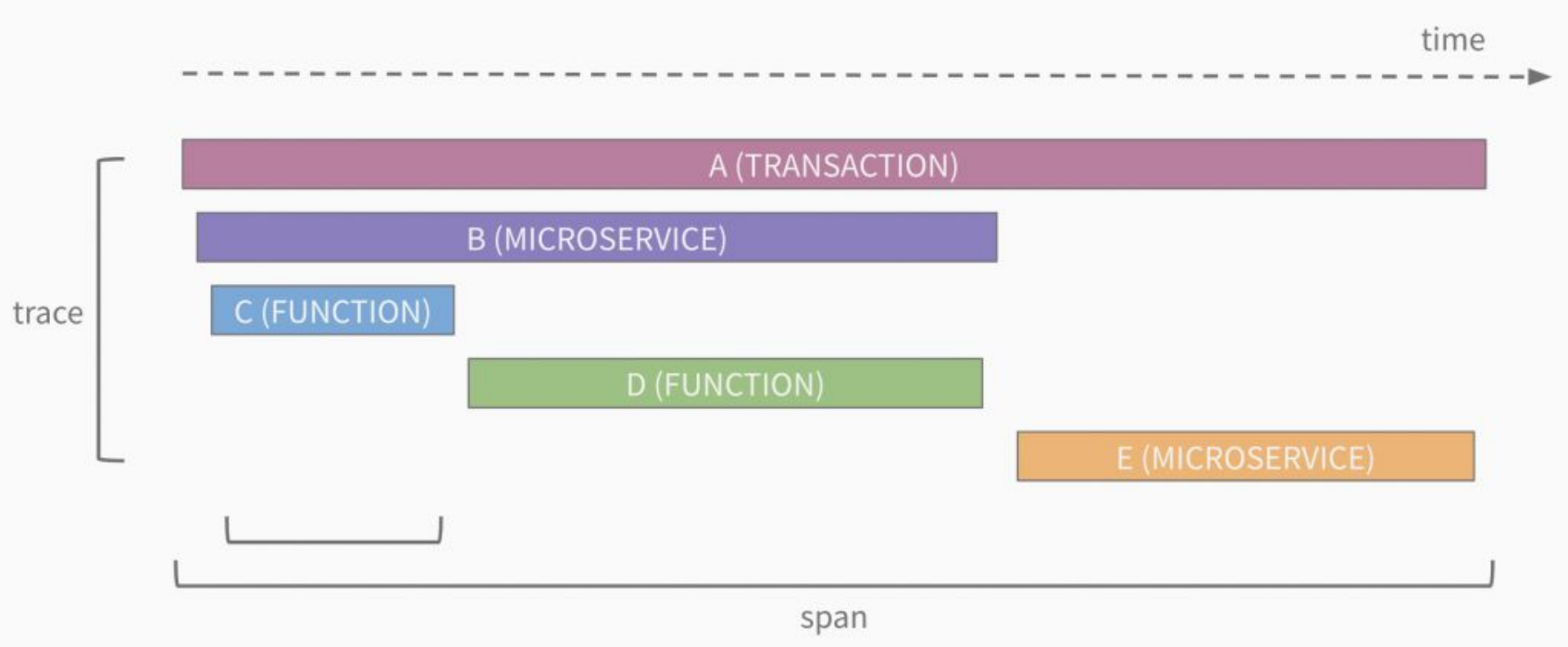
Number of Classes Loaded (top 10)



Thread Count (top 10)

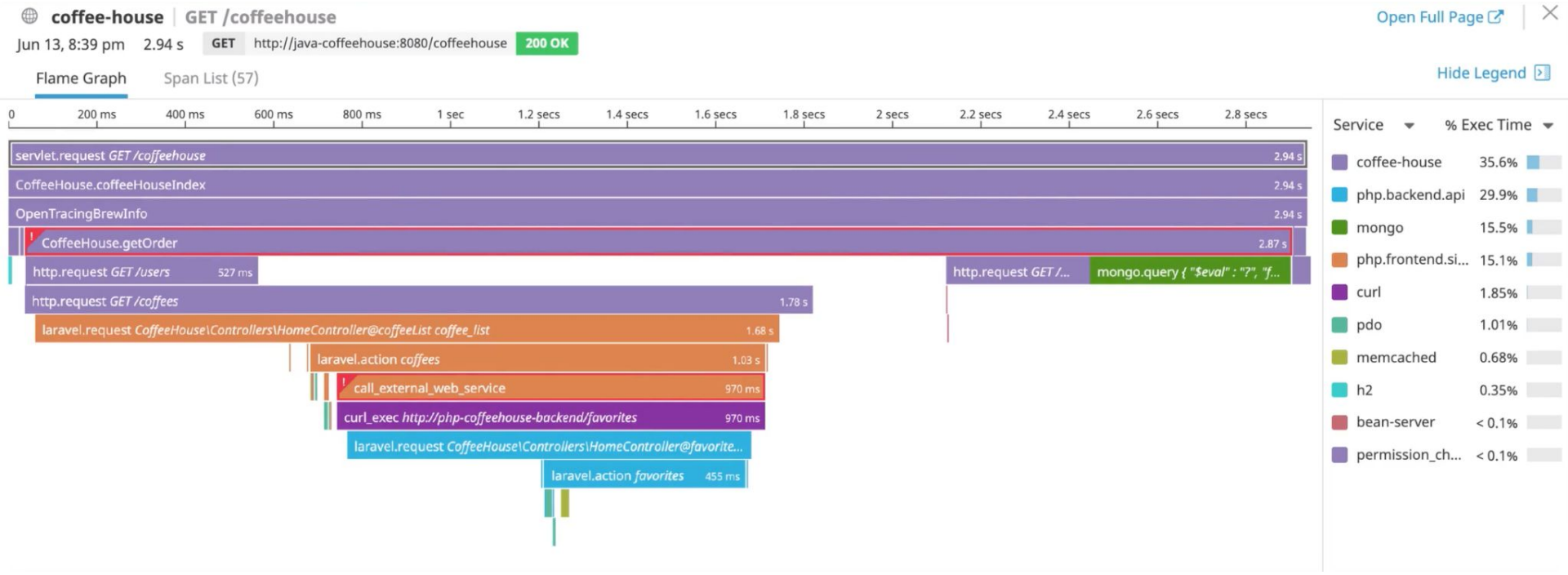
ajunaky

Traces



https://sflanders.net/2019/03/28/an-intro-to-distributed-tracing/

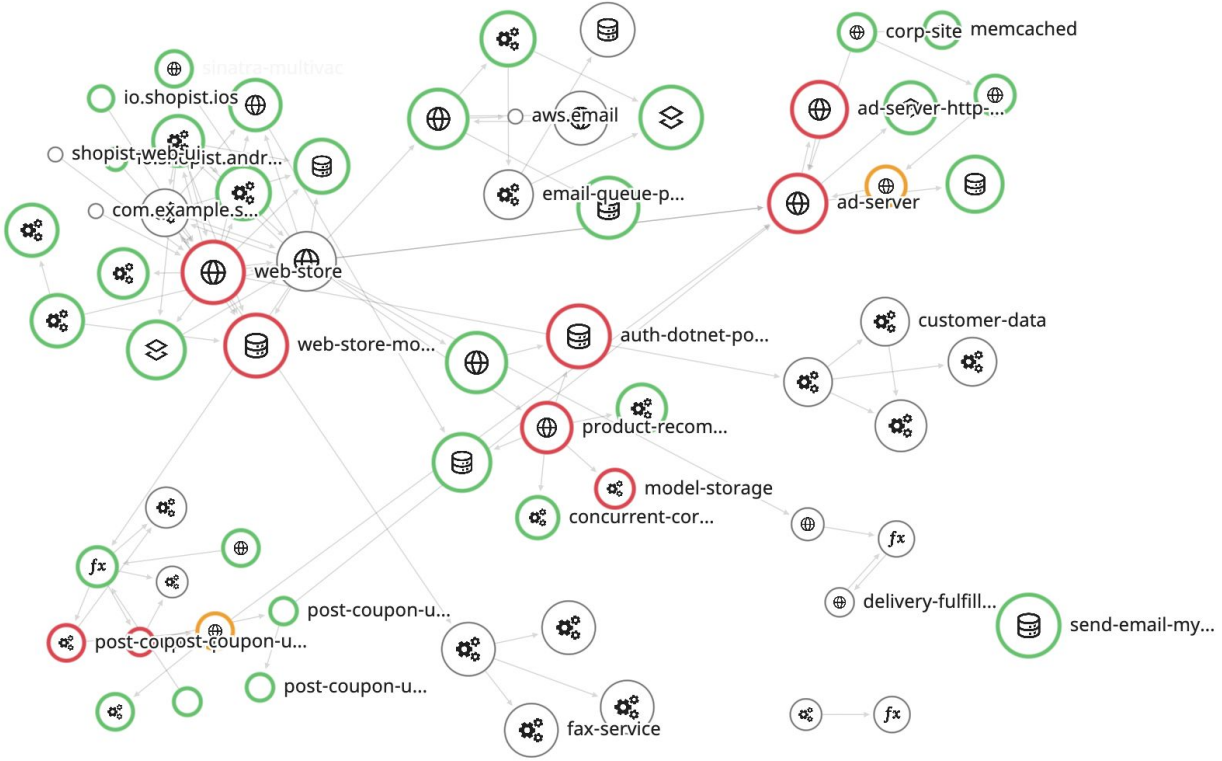
Traces



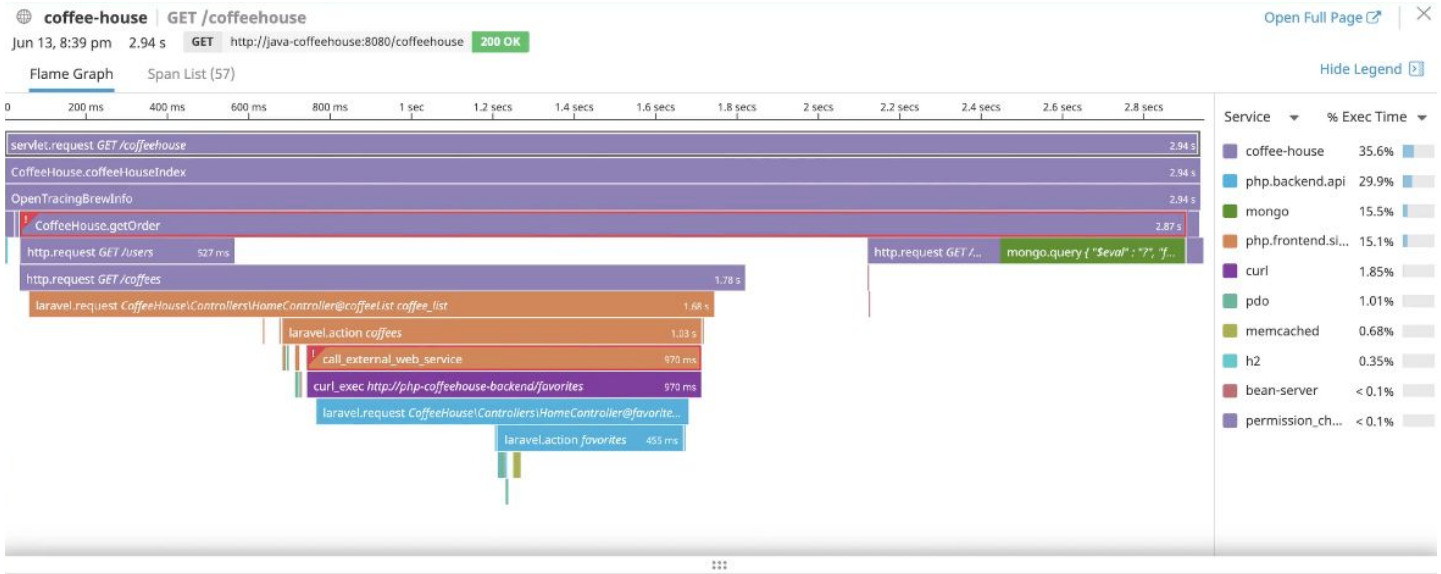
ajunaky

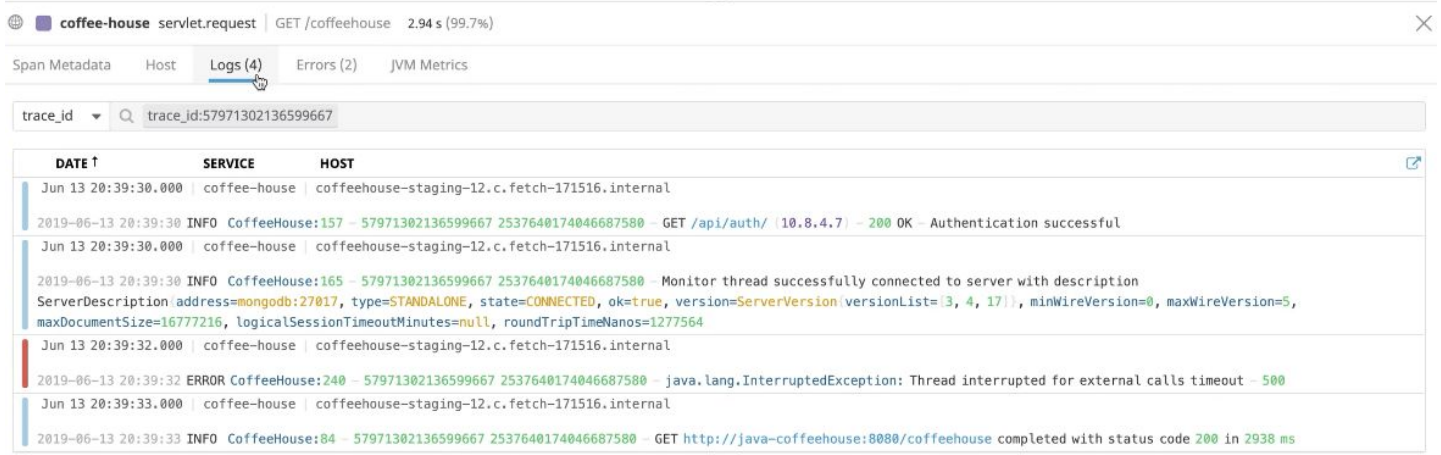
```
package io.opentelemetry.example.jaeger;

import io.opentelemetry.api.OpenTelemetry;
import io.opentelemetry.api.trace.Span;
import io.opentelemetry.api.trace.Tracer;

public final class JaegerExample {

  private final Tracer tracer;

  public JaegerExample(OpenTelemetry openTelemetry) {
    tracer = openTelemetry.getTracer("io.opentelemetry.example.JaegerExample");
  }

  private void myWonderfulUseCase() {
    // Generate a span
    Span span = this.tracer.spanBuilder("Start my wonderful use case").startSpan();
    span.addEvent("Event 0");
    // execute my use case - here we simulate a wait
    doWork();
    span.addEvent("Event 1");
    span.end();
  }

  private void doWork() {
    try {
      Thread.sleep(1000);
    } catch (InterruptedException e) {
      // do the right thing here
    }
  }
```

Example of tracing with Open Telemetry



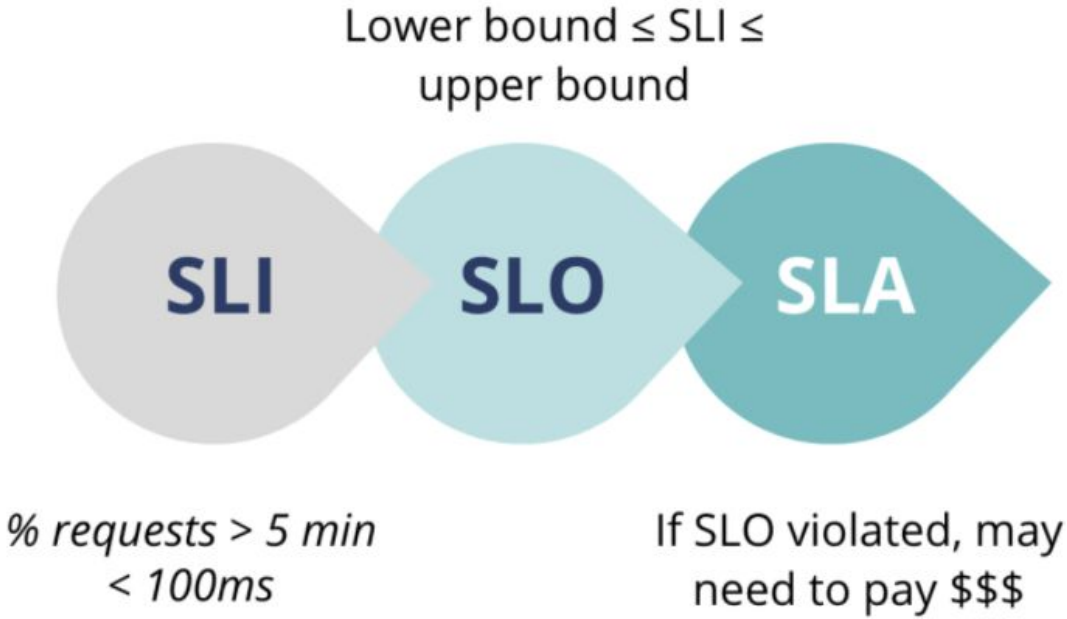
Service Map - Application Performance Monitoring


coffee-house | GET /coffeehouse
Jun 13, 8:39 pm 2.94 s GET http://java-coffeehouse:8080/coffeehouse 200 OK
Open Full Page
Flame Graph
Span List (57)
Hide Legend
0 200 ms 400 ms 600 ms 800 ms 1 sec 1.2 secs 1.4 secs 1.6 secs 1.8 secs 2 secs 2.2 secs 2.4 secs 2.6 secs 2.8 secs
servlet.request GET /coffeehouse 2.94 s
CoffeeHouse.coffeeHouseIndex 2.94 s
OpenTracingBrewInfo 2.94 s
CoffeeHouse.getOrder 2.87 s
http.request GET /users 527 ms
http.request GET /... mongo.query { "$eval" : "?", "f...
http.request GET /coffees 1.78 s
laravel.request CoffeeHouse\Controllers\HomeController@coffeeList coffee_list 1.68 s
laravel.action coffees 1.03 s
call_external_web_service 970 ms
curl_exec http://php-coffeehouse-backend/favorites 970 ms
laravel.request CoffeeHouse\Controllers\HomeController@favorite...
laravel.action favorites 455 ms
Service % Exec Time
coffee-house 35.6%
php.backend.api 29.9%
mongo 15.5%
php.frontend.si... 15.1%
curl 1.85%
pdo 1.01%
memcached 0.68%
h2 0.35%
bean-server < 0.1%
permission_ch... < 0.1%


coffee-house servlet.request | GET /coffeehouse 2.94 s (99.7%)
Span Metadata
Host
Logs (4)
Errors (2)
JVM Metrics
trace_id
trace_id:57971302136599667
DATE ↑ SERVICE HOST
Jun 13 20:39:30.000 | coffee-house | coffeehouse-staging-12.c.fetch-171516.internal
2019-06-13 20:39:30 INFO CoffeeHouse:157 - 57971302136599667 2537640174046687580 - GET /api/auth/ [10.8.4.7] - 200 OK - Authentication successful
Jun 13 20:39:30.000 | coffee-house | coffeehouse-staging-12.c.fetch-171516.internal
2019-06-13 20:39:30 INFO CoffeeHouse:165 - 57971302136599667 2537640174046687580 - Monitor thread successfully connected to server with description
ServerDescription{address=mongodb:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[3, 4, 17]}, minWireVersion=0, maxWireVersion=5,
maxDocumentSize=16777216, logicalSessionTimeoutMinutes=null, roundTripTimeNanos=1277564
Jun 13 20:39:32.000 | coffee-house | coffeehouse-staging-12.c.fetch-171516.internal
2019-06-13 20:39:32 ERROR CoffeeHouse:240 - 57971302136599667 2537640174046687580 - java.lang.InterruptedException: Thread interrupted for external calls timeout - 500
Jun 13 20:39:33.000 | coffee-house | coffeehouse-staging-12.c.fetch-171516.internal
2019-06-13 20:39:33 INFO CoffeeHouse:84 - 57971302136599667 2537640174046687580 - GET http://java-coffeehouse:8080/coffeehouse completed with status code 200 in 2938 ms

ajunaky

Service Level .*



Icons made by eucalyp from flaticon.com

Monitoring

Alerting

Icons made by eucalyp from flaticon.com

ajunaky

Honestly Black Lives Matter
@honest_update

Things were unstable for a bit, but we fixed* them!

* Adjusted the monitoring thresholds so they'd shut up.

When do you start monitoring?

4 Golden Signals (LETS)

Latency

- Time taken to service requests

Errors

- Rate of failing requests

Traffic

- Measure of demand on servers

Saturation

- How “full” your service is

Collecting Data is Cheap;

Not having it when you need it is expensive

https://www.datadoghq.com/blog/monitoring-101-collecting-data/

Java observability tooling options

- Enterprise Observability e.g Datadog
- Open source telemetry e.g OpenTelemetry
- Open source application monitoring e.g Micrometer
- Build your own
- Find one with integration with existing services e.g Tomcat or Spring
- Lots of these tools use JMX to expose your metrics and for tracing

Lots of other Facets of Observability

- Metrics
- Logs
- Traces
- SLOs
- Alerting
- Real User Monitoring
- Synthetics
- Security
- Network monitoring
- Dashboards
- ...

ajuna

Questions?

Email: ajuna@datadoghq.com

ajunaky

spring initializr

Project
- (●) Maven Project
- ( ) Gradle Project

Language
- (●) Java
- ( ) Kotlin
- ( ) Groovy

Spring Boot
- ( ) 3.0.0 (SNAPSHOT)
- ( ) 3.0.0 (M2)
- ( ) 2.7.0 (SNAPSHOT)
- ( ) 2.7.0 (M3)
- ( ) 2.6.8 (SNAPSHOT)
- (●) 2.6.7
- ( ) 2.5.14 (SNAPSHOT)
- ( ) 2.5.13

Project Metadata

Group	com.example
Artifact	demo
Name	demo
Description	Demo project for Spring Boot
Package name	com.example.demo
Packaging	(●) Jar ( ) War

Dependencies ADD DEPENDENCIES... ⌘ + B

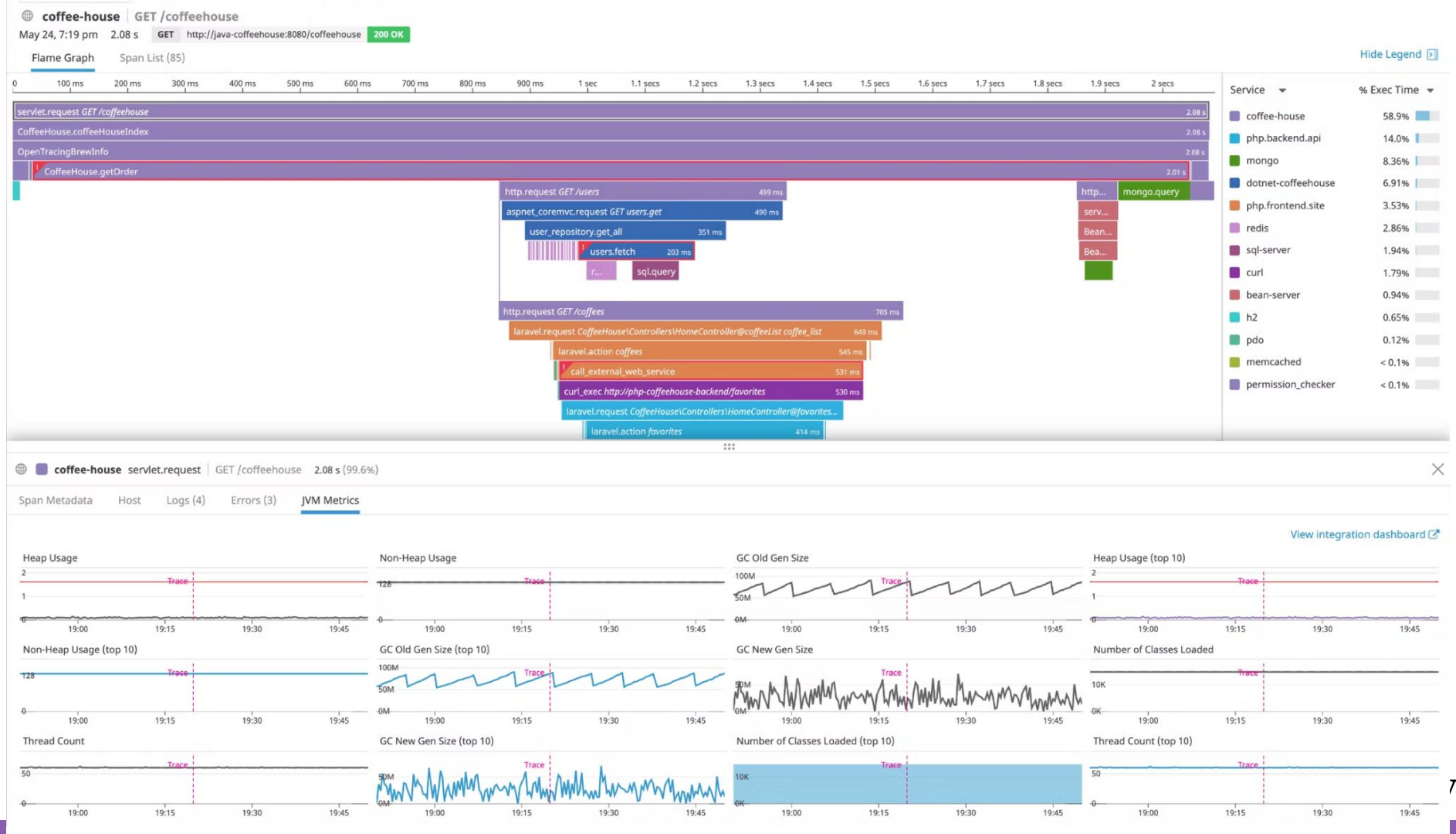
Datadog OBSERVABILITY
Publish Micrometer metrics to Datadog, a dimensional time-series SaaS with built-in dashboarding and alerting.

Spring Boot Actuator OPS
Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.

Spring
Provide
enhance

Spring
Build we
Tomcat

METRIC TYPE	USE IT FOR...	EXAMPLES
Gauge	Measuring resource usage, capacity, etc. Values that can rise and fall, and that have fixed upper bounds	Size of a collection, number of running threads, number of messages on a queue, memory usage
Counter	Measuring a number of events or actions - a value that only increases, and never decreases.	Total number of orders processed, total tasks completed, etc.
Timer	Measuring short-duration events and their frequency	Method execution time, request duration, time taken to boil an egg.


coffee-house | GET /coffeehouse
May 24, 7:19 pm 2.08 s GET http://java-coffeehouse:8080/coffeehouse 200 OK
Flame Graph
Span List (85)
Hide Legend
0 100 ms 200 ms 300 ms 400 ms 500 ms 600 ms 700 ms 800 ms 900 ms 1 sec 1.1 secs 1.2 secs 1.3 secs 1.4 secs 1.5 secs 1.6 secs 1.7 secs 1.8 secs 1.9 secs 2 secs
servlet.request GET /coffeehouse 2.08 s
CoffeeHouse.coffeeHouseIndex 2.08 s
OpenTracingBrewInfo 2.08 s
CoffeeHouse.getOrder 2.01 s
http.request GET /users 499 ms
aspnet_coremvc.request GET users.get 490 ms
user_repository.get_all 351 ms
users.fetch 203 ms
sql.query
http.request GET /coffees 705 ms
laravel.request CoffeeHouse\Controllers\HomeController@coffeeList coffee_list 649 ms
laravel.action coffees 545 ms
call_external_web_service 531 ms
curl_exec http://php-coffeehouse-backend/favorites 530 ms
laravel.request CoffeeHouse\Controllers\HomeController@favorites...
laravel.action favorites 414 ms
mongo.query
Service
% Exec Time
coffee-house 58.9%
php.backend.api 14.0%
mongo 8.36%
dotnet-coffeehouse 6.91%
php.frontend.site 3.53%
redis 2.86%
sql-server 1.94%
curl 1.79%
bean-server 0.94%
h2 0.65%
pdo 0.12%
memcached < 0.1%
permission_checker < 0.1%
coffee-house servlet.request | GET /coffeehouse 2.08 s (99.6%)
Span Metadata
Host
Logs (4)
Errors (3)
JVM Metrics
View integration dashboard
Heap Usage
Non-Heap Usage
GC Old Gen Size
Heap Usage (top 10)
Non-Heap Usage (top 10)
GC Old Gen Size (top 10)
GC New Gen Size
Number of Classes Loaded
Thread Count
GC New Gen Size (top 10)
Number of Classes Loaded (top 10)
Thread Count (top 10)
Trace
19:00 19:15 19:30 19:45

References

- https://www.digitalocean.com/community/tutorials/an-introduction-to-metrics-monitoring-and-alerting
- https://www.datadoghq.com/blog/monitoring-101-collecting-data/
- https://www.datadoghq.com/blog/monitoring-101-alerting/
- https://www.datadoghq.com/blog/monitoring-101-investigation/
- https://blog.appoptics.com/6-steps-to-get-started-with-application-performance-monitoring-apm/
- https://www.dynatrace.com/news/blog/open-observability-part-1-distributed-tracing-and-observability/
- https://sflanders.net/2019/03/28/an-intro-to-distributed-tracing/
- https://slack.engineering/tracing-at-slack-thinking-in-causal-graphs/