

Ask your logs

A story



Dashboard

Busy

Queues

Retries

Scheduled

Dead

**38,486**

Processed

**234**

Failed

**45**

Busy

**801,543**

Enqueued

**0**

Retries

**0**

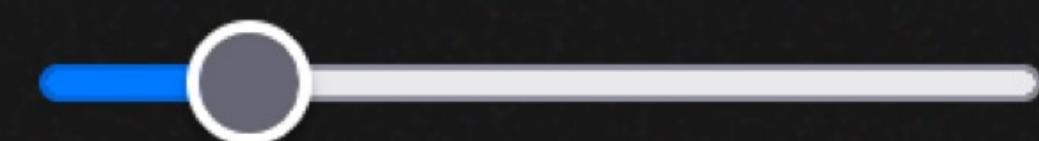
Scheduled

**0**

Dead

Dashboard •

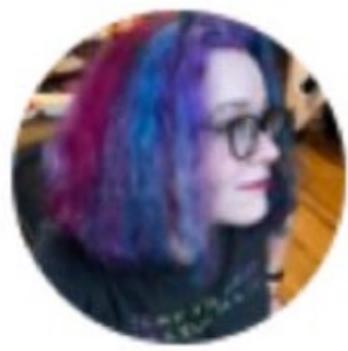
Polling interval: 5 sec



"Can't we check our logs?"



Our logs



**Charity Majors**

@mipsytipsy

...

<people> what's the best logging solution?

<me> logs are trash. we don't keep trash

10:22 PM · May 10, 2022

Our logs are trash

Our logs are trash not data

What if they were?

Hi! 

# Youssef



# Senior Engineer @ Workato

# Youssef



## Ruby and developer tools

# Youssef Photography







# Agenda

1. The default Rails logs
2. Towards structured logs
3. Structured logs showcase

# Agenda

1. The default Rails logs
2. Towards structured logs
3. Structured logs showcase

# A basic rails action

```
● ● ●  
def update  
  @user = User.find(params[:id])  
  @user.update(user_params)  
  redirect_to action: :show  
end
```

# The default Rails logs

```
● ● ●  
1 Started PATCH "/users/1" for 127.0.0.1 at 2024-04-12 11:57:30  
-0500  
2 Processing by UsersController#update as HTML  
3   Parameters: {"user"=>{"name"=>"Youssef", "country_code"=>"SE",  
"locale"=>"en", "runs_visible_to_admins"=>"1"}, "commit"=>"Update  
User", "id"=>"1"}  
4 Redirected to http://localhost:3000/users/1  
5 Completed 302 Found in 16ms ( ActiveRecord: 3.2ms | Allocations:  
3311)
```

Questions we  
can ask these  
logs

How many PATCH requests  
do we process per day?

Which controllers are our  
busiest?

What's our rate of 404  
responses?

*Are these really the  
questions we want to ask?*

“Judge a man  
by his questions  
rather than by  
his answers.”

— Voltaire



Questions we  
want to ask

Which requests did this  
specific user perform?

Why did we suddenly  
enqueue a large amount of  
jobs?

How many users are using  
this legacy API endpoint?  
What do they have in  
common?

Is this feature flag affecting  
the database  
performance...

Is this feature flag affecting  
the database  
performance...

... on revision 1b56dc...

Is this feature flag affecting  
the database performance...

... on revision 1b56dc...

... in us-east-1?

Why can't we  
answer these  
questions?



```
1 Started PATCH "/users/1" for 127.0.0.1 at 2024-04-12 11:57:30 -0500
2 Processing by UsersController#update as HTML
3   Parameters: {"user"=>{"name"=>"Youssef", "country_code"=>"SE", "locale"=>"en",
"runs_visible_to_admins"=>"1"}, "commit"=>"Update User", "id"=>"1"
4 Redirected to http://localhost:3000/users/1
5 Completed 302 Found in 16ms ( ActiveRecord: 3.2ms | Allocations: 3311)
```

1. Multi-line
2. Unstructured
3. Missing data
4. Uncorrelated

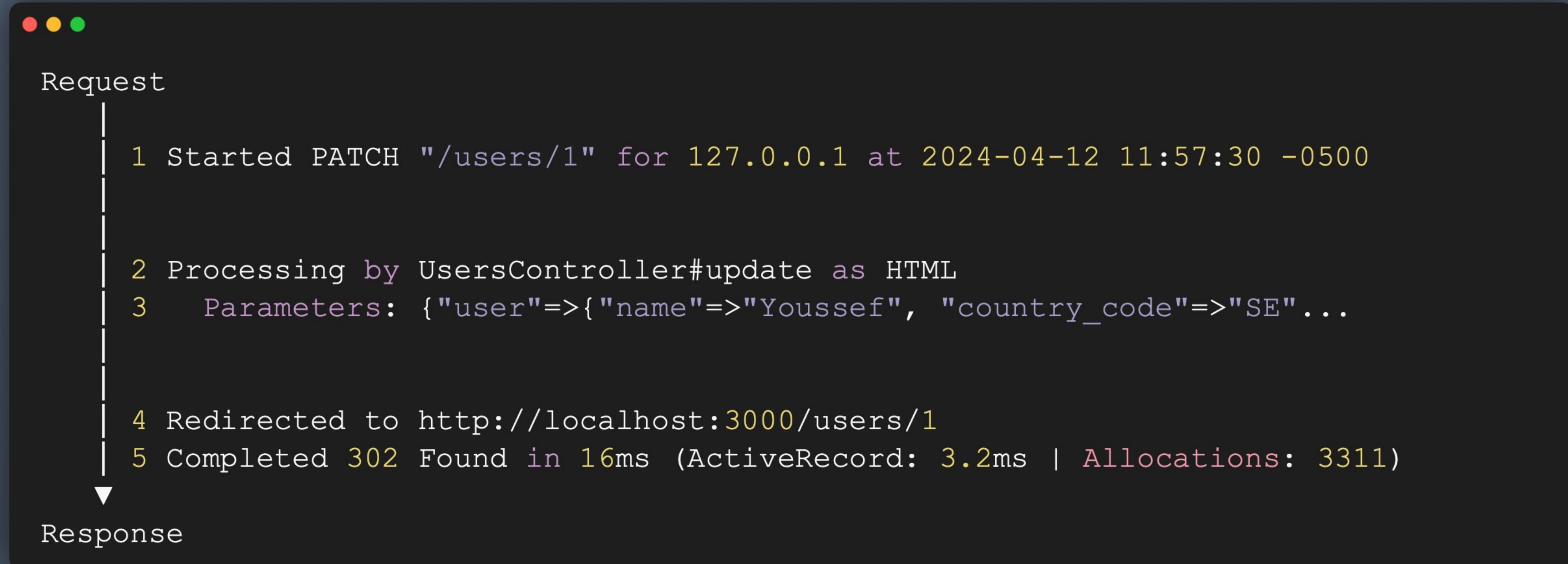
Can we fix this?

# Agenda

1. The default Rails logs
2. Towards structured logs
3. Structured logs showcase

Problem #1:  
logs are multiline

# "Print-as-you-go" logs



The screenshot shows a terminal window with a dark background and light-colored text. At the top left are three small colored dots (red, yellow, green). The window title is "Request". Below the title, a vertical dashed line separates the title from the log entries. The log entries are numbered 1 through 5:

- 1 Started PATCH "/users/1" for 127.0.0.1 at 2024-04-12 11:57:30 -0500
- 2 Processing by UsersController#update as HTML
- 3 Parameters: {"user"=>{ "name"=>"Youssef", "country\_code"=>"SE"...}}
- 4 Redirected to http://localhost:3000/users/1
- 5 Completed 302 Found in 16ms ( ActiveRecord: 3.2ms | Allocations: 3311)

At the bottom left, below the log entries, is a small downward-pointing arrow icon. The word "Response" is written in a smaller font at the very bottom left.

```
Request
1 Started PATCH "/users/1" for 127.0.0.1 at 2024-04-12 11:57:30 -0500
2 Processing by UsersController#update as HTML
3 Parameters: {"user"=>{ "name"=>"Youssef", "country_code"=>"SE"...}}
4 Redirected to http://localhost:3000/users/1
5 Completed 302 Found in 16ms ( ActiveRecord: 3.2ms | Allocations: 3311)

Response
```

# 1. Consolidate



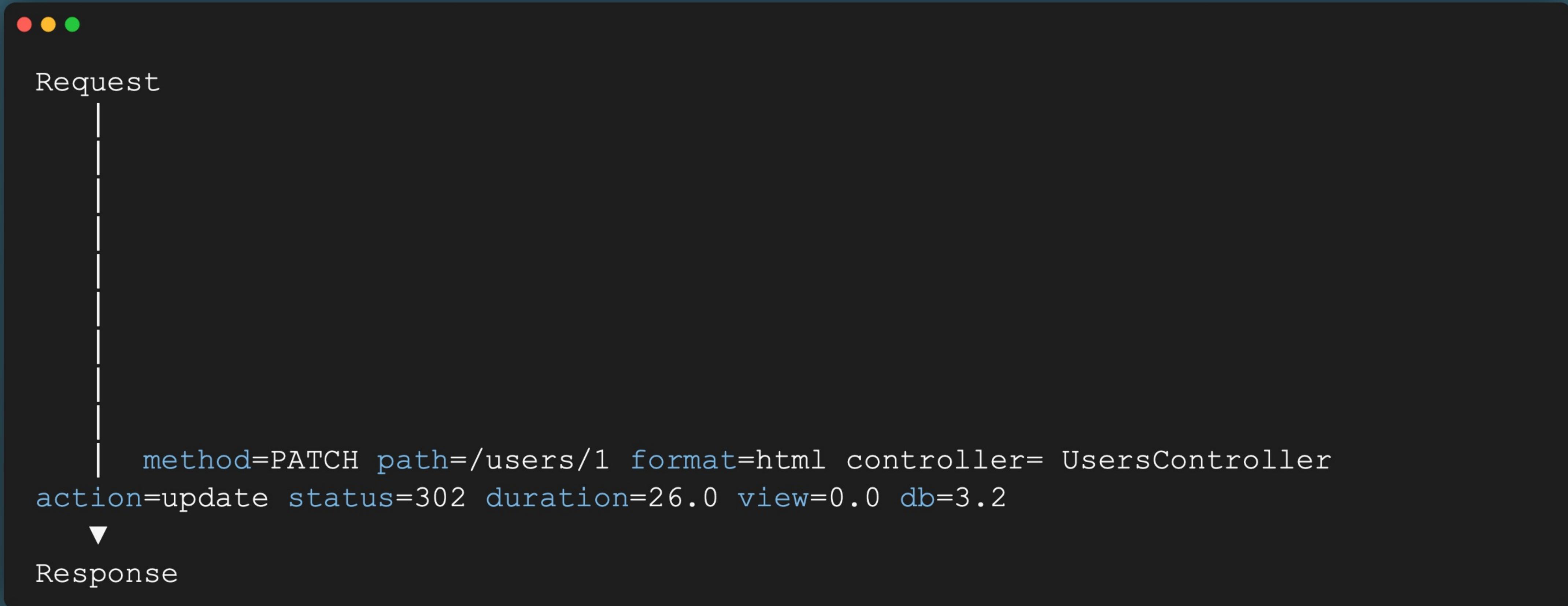
# Consolidating logs

```
gem "lograge"
```

or

```
gem "rails_semantic_logger"
```

# Consolidated logs



The image shows a dark-themed terminal window on a Mac OS X system. In the top-left corner, there are three colored window control buttons: red, yellow, and green. The main area of the terminal is labeled "Request" on the left side. A vertical dashed line separates the "Request" label from the log entry. The log entry itself is a single line of text: "method=PATCH path=/users/1 format=html controller= UsersController action=update status=302 duration=26.0 view=0.0 db=3.2". Below this line, there is a small black downward-pointing arrow. At the bottom of the terminal window, the word "Response" is visible.

```
method=PATCH path=/users/1 format=html controller= UsersController  
action=update status=302 duration=26.0 view=0.0 db=3.2
```

▼

Response

Problem #2:  
logs are unstructured

1. Consolidate
2. Structure



# Enabling structured logs



```
# config/environments/production.rb
Rails.application.configure do
  config.lograge.formatter = Lograge::Formatters::Json.new
end
```

# JSON logs

```
 1 {"controller": "UsersController", "action": "update", "format":  
  "HTML", "duration": "16ms", "method": "PATCH", "path": "/users/1",  
  "status": 302, "db_runtime": 3.2, "allocations": 3311, "ip":  
  127.0.0.1, "timestamp": "2024-04-12 11:57:30 -0500", "params": {  
    "user": { "name": "Youssef", "country_code": "SE", "locale": "en",  
    "runs_visible_to_admins": "1" }, "commit": "Update User", "id": "1"  
  } }
```

# JSON logs

```
● ● ●

{
  "controller": "UsersController",
  "action": "update",
  "format": "HTML",
  "duration": "16ms",
  "method": "PATCH",
  "path": "/users/1",
  "status": 302,
  "db_runtime": 3.2,
  "allocations": 3311,
  "ip": 127.0.0.1 ,
  "timestamp": "2024-04-12 11:57:30 -0500",
  "params": { "user": { "name": "Youssef", "country_code": "SE", "locale": "en", "runs_visible_to_admins": "1" }, "commit": "Update User", "id": "1" }
}
```

We turned logs into data 

Problem #3:  
logs are missing data

1. Consolidate

2. Structure

3. Enrich



# Enriching logs with domain data

● ● ●

```
# In ApplicationController#append_info_to_payload

payload.merge!(
  # ENV data
  proctype: ENV["PROCTYPE"],
  revision: ENV["REVISION"],
  aws_region: ENV["AWS_REGION"],

  # Request data
  request_id: request.request_id,
  user_agent: request.user_agent,

  # Current context data
  locale: I18n.locale,
  user: {
    id: current_user&.id,
    email: current_user&.email,
  }
)
```

# Enriching logs with domain data

```
{  
  [...]  
  "proctype": "web",  
  "revision": "67bcd8c",  
  "aws_region": "us-east-1",  
  "request_id": "b7f21610-d502-4da7-a34e-f53d3a41b249",  
  "user_agent": "MobileApp/2.3",  
  "locale": "en",  
  "user": { "id": 1, "email": "youssef@example.com" },  
  # ...  
}
```



Footguns ahead





# Performance

Only log information that you have already computed or is cheap to compute



## ⚠ Accidentally exposing PII

- Be careful when logging PII or other sensitive data
- `params` often contain sensitive data

# ⚠️ Accidentally exposing PII

- ActiveSupport::ParameterFilter  
can be used to filter sensitive data

```
● ○ ●  
{  
  "password": "[REDACTED]"  
  // [...]  
}
```



# Facebook Stored Hundreds of Millions of User Passwords in Plain Text for Years

March 21, 2019

Problem #4:  
logs are uncorrelated

1. Consolidate

2. Structure

3. Enrich

4. Correlate

# Example #1: distributed systems

Generate a correlation ID at the beginning of each request and propagate it



```
correlation_id: request.headers["X-Correlation-Id"]
```

# Example #2: jobs and web requests

Add data using a Sidekiq middleware:

```
{
  "thread": "processor",
  "name": "Sidekiq",
  "message": "start",
  "enqueued_by_request_id": "fc1ab3f7-7cda013718b9"
}
```

# Recap

# 1. Consolidate

# 2. Structure

# 3. Enrich

# 4. Correlate

```
1 {"controller": "UsersController",
  "action": "update", "format": "HTML",
  "duration": "16ms", "method": "PATCH",
  "path": "/users/1", "status": 302,
  "db_runtime": 3.2, "allocations": 3311, "ip": 127.0.0.1, "timestamp": "2024-04-12 11:57:30 -0500", "params": { "user": { "name": "Youssef", "country_code": "SE", "locale": "en", "runs_visible_to_admins": "1" }, "commit": "Update User", "id": "1" }, "react_route": "/home/dashboard", "request_id": "6aebf3e9-1ea3-4be4-b393-d24f50133744"}
```

# Agenda

1. The default Rails logs
2. Towards structured logs
3. Structured logs showcase

Logs are now data.  
We can treat them as such.

**AppSignal** 

 **new relic.**<sup>®</sup>

**splunk**<sup>®</sup> >



**Grafana loki**



  
**kibana**

  
**honeycomb.io**

  
**Honeybadger**



kibana



```
1 {"controller": "UsersController", "action": "update", "format":  
"HTML", "duration": "16ms", "method": "PATCH", "path": "/users/1",  
"status": 302, "db_runtime": 3.2, "allocations": 3311, "ip":  
127.0.0.1, "timestamp": "2024-04-12 11:57:30 -0500", "params": {  
"user": { "name": "Youssef", "country_code": "SE", "locale": "en",  
"runs_visible_to_admins": "1" }, "commit": "Update User", "id": "1"  
} }
```

# Expanded document

View: [Single document](#) [Surrounding documents](#) [?](#) [<](#) [<](#) **1 of 500** [>](#) [>](#)

	ssage": "OK" }}
<b>k</b> host	Youssefs-Air-4
<b>k</b> level	fatal
<b>t</b> message	Completed create
<b>k</b> name	InvoicesController
<b>k</b> named_tags.proctype	worker
<b>k</b> named_tags.revision	86a2381b76
<b>k</b> named_tags.ruby_verse	3.2.2
	ion
<b>k</b> payload.action	create
<b>#</b> payload.allocations	98,639,986
<b>k</b> payload.controller	InvoiceController
<b>#</b> payload.db_runtime	560.43
<b>k</b> payload.format	json
<b>k</b> payload.host	localhost
<b>IP</b> payload.ip	48.255.0.116
<b>k</b> payload.locale	fr
<b>k</b> payload.method	GET

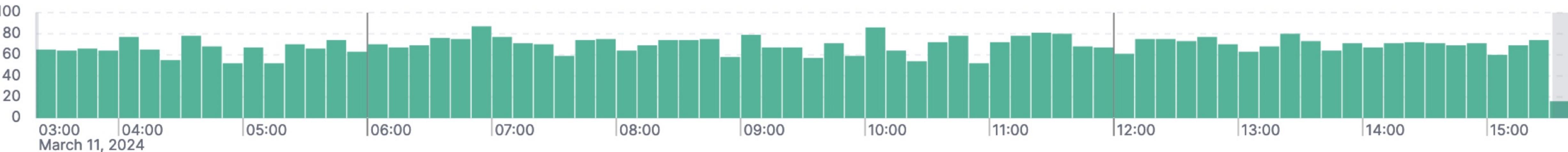
Started POST "/runs" for 127.0.0.1 at 2024-04-12 11:42:47 -0500  
Processing by RunsController#create as HTML  
Parameters: {"authenticity\_token"=>"kdbzA7qCAZAiqyuT3w/ZUJF6wXmlcc465Ae7l5ATkuPKWUF5cpEFiBPVHJ43R1MGQBZKn1/J1YnGv69ga", "model\_file"=>#<ActionDispatch::Http::UploadedFile:0x0000000114d5cb08 @tempfile=#<Tempfile:/var/folders/wn/fd70mprx3g5\_f1q07wr8ksth0000gn/T/RackMultipart20240412-12113-avuhu.jpg>, @original\_filename="525840896.jpg", @content\_type="image/jpeg", @headers="Content-Disposition: form-data; name="model\_file""; filename=\"525840896.jpg\"\r\nContent-Type: image/jpeg\r\n", "data\_file"=>#<ActionDispatch::Http::UploadedFile:0x0000000114d5cb08 @tempfile=/var/folders/wn/fd70mprx3g5\_f1q07wr8ksth0000gn/T/RackMultipart20240412-12113-oyghdt.jpg>, @original\_filename="525840896.jpg", @content\_type="image/jpeg", @headers="Content-Disposition: form-data; name="run[data\_file]""; filename=\"525840896.jpg\"\r\nContent-Type: image/jpeg\r\n", "description"=>"asjkh", "language"=>"en", "server\_type"=>"sidekiq", "notify\_when\_finished"=>"1", "version\_id"=>"1"}, {"commit"=>"S3 Storage (1989.5ms) Uploaded file to key: 7cik8fqe4wg8kb4okt16lemxo0sq (checksum: 5VhCbV/U6rg/6wHR96pVVA==)", "commit"=>"S3 Storage (907.1ms) Uploaded file to key: 1oix1f7dkald4dn6se7pw2n7bb01 (checksum: 5VhCbV/U6rg/6wHR96pVVA==)"}, Enqueued GenerateResJob (Job ID: e59bff77-1501-470d-ad61-ccfaff4d5106) to Sidekiq(low\_priority) with arguments: 2 [ActiveJob] Enqueued ActiveStorage::AnalyzeJob (Job ID: 8a15bae5-15d6-462f-be0b-3a3f27fa54dc) to Sidekiq(active\_storage\_low) [ActiveJob] Enqueued ActiveStorage::AnalyzeJob (Job ID: d5a33fb0-3c21-475c-83e0-6ce8db2c4648) to Sidekiq(active\_storage\_low) Redirected to http://localhost:3000/versions/1  
Completed 302 Found in 3048ms ( ActiveRecord: 21.7ms | Allocations: 21429)

Started GET "/versions/1" for 127.0.0.1 at 2024-04-12 11:42:50 -0500  
Processing by VersionsController#show as HTML  
Parameters: {"id"=>"1"}  
Rendered versions/show.html.slim within layouts/application (Duration: 14.6ms | Allocations: 5594)  
[Webpacker] Everything's up-to-date. Nothing to do  
Rendered layout layouts/application.html.slim (Duration: 29.6ms | Allocations: 14615)  
Completed 200 OK in 36ms (Views: 25.9ms | ActiveRecord: 4.9ms | Allocations: 16663)

Started GET "/cable" for 127.0.0.1 at 2024-04-12 11:42:50 -0500  
Started GET "/cable/" [WebSocket] for 127.0.0.1 at 2024-04-12 11:42:50 -0500  
Successfully upgraded to WebSocket (REQUEST\_METHOD: GET, HTTP\_CONNECTION: keep-alive, Upgrade, HTTP\_UPGRADE: websocket)  
Registered connection (Z2lk0i8vb3NlbW9zeXMtY2xvdWQvVXNlc18x)  
RunChannel is streaming from runs

5,072 hits

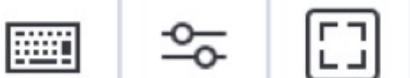
Break down by Select field



Mar 11, 2024 @ 03:21:24.061 - Mar 11, 2024 @ 15:31:30.345 (interval: Auto - 10 minutes)

## Documents Field statistics

1 field sorted



↓ @timestamp ⏱

▼ Document



↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:23.000	@timestamp Mar 11, 2024 @ 15:31:23.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 779.52ms duration_ms 779.52 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:22.000	@timestamp Mar 11, 2024 @ 15:31:22.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 452.99ms duration_ms 452.99 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:20.000	@timestamp Mar 11, 2024 @ 15:31:20.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 645.06ms duration_ms 645.06 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:20.000	@timestamp Mar 11, 2024 @ 15:31:20.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 265.15ms duration_ms 265.15 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:07.000	@timestamp Mar 11, 2024 @ 15:31:07.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 806.32ms duration_ms 806.32 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:31:04.000	@timestamp Mar 11, 2024 @ 15:31:04.000 @version 1 application Semantic Logger data_stream.dataset generic data_stream.namespace default data_stream.type logs duration 270.53ms duration_ms 270.53 environment product...
↗ <input type="checkbox"/> Mar 11, 2024 @ 15:30:55.000	@timestamp Mar 11, 2024 @ 15:30:55.000 @version 1 application Semantic Logger data_stream.dataset generic

Back to our  
questions

Which requests did this  
specific user perform?

D Discover

New Open Share Alerts Inspect

Save

logs-\*



Filter your data using KQL syntax

Last 1 day



Popular fields

7

k action  
k controller  
k duration  
k event.original  
k service

# status  
# user.id

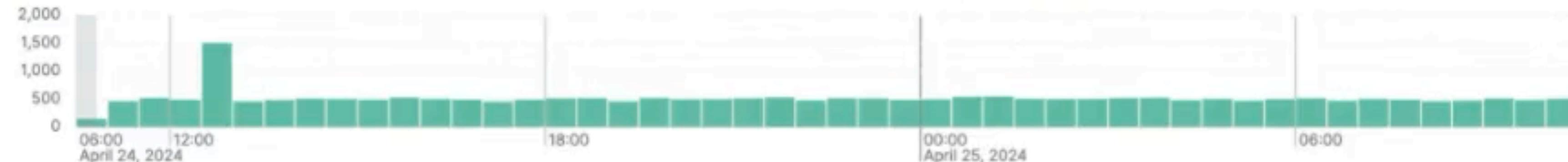
Available fields

40

@timestamp  
k @version  
k action  
# allocations  
k aws\_region  
k client\_request\_id  
k controller  
k data\_stream.dataset  
k data\_stream.namespac  
e  
k data\_stream.type

Add a field

24,721 hits



Break down by Select field



Documents Field statistics

1 field sorted



@timestamp Document

Apr 25, 2024 @ 10:32:36.000	@timestamp Apr 25, 2024 @ 10:32:36.000 @version 1 action create allocations 59,003,177 aws_region us-eas t-1 client_request_id 2f57b3e3-b1a2-4984-a653-2c55acd0a0b0 controller InvoicesController...
Apr 25, 2024 @ 10:32:34.000	@timestamp Apr 25, 2024 @ 10:32:34.000 @version 1 action update allocations 83,424,491 aws_region eu-north h-1 client_request_id a8a52202-ef85-454d-a74b-aff4dc937ea5 controller ConferencesController...
Apr 25, 2024 @ 10:32:25.000	@timestamp Apr 25, 2024 @ 10:32:25.000 @version 1 action create allocations 89,231,346 aws_region us-eas t-1 client_request_id f8bf00d8-ba33-4974-8723-f41cc065629c controller ConferencesController...
Apr 25, 2024 @ 10:32:24.000	@timestamp Apr 25, 2024 @ 10:32:24.000 @version 1 action create allocations 11,989,002 aws_region us-eas t-1 client_request_id 72f6594b-5f2d-4064-b62b-ed31f9846d7b controller ConferencesController...
Apr 25, 2024 @ 10:32:19.000	@timestamp Apr 25, 2024 @ 10:32:19.000 @version 1 action index allocations 21,258,153 aws_region eu-north h-1 client_request_id 950f6c12-17ac-40cb-a9d1-9eae1536aa06 controller TicketsController...
Apr 25, 2024 @ 10:32:19.000	@timestamp Apr 25, 2024 @ 10:32:19.000 @version 1 action create allocations 41,620,789 aws_region us-eas t-1 client_request_id b4594c37-28e5-4dfb-bc50-664052fa91c6 controller AttendeesController...
Apr 25, 2024 @ 10:32:17.000	@timestamp Apr 25, 2024 @ 10:32:17.000 @version 1 action show allocations 54,962,534 aws_region eu-centra

Rows per page: 100

&lt; 1 2 3 4 5 &gt;

Why did we suddenly  
enqueue a large amount of  
jobs?

D Discover

New Open Share Alerts Inspect

Save

logs-\*



Filter your data using KQL syntax

Last 1 day



Search f



0

## Popular fields

9

- k action
- k controller
- k duration
- # user.id
- k enqueueued\_by\_request\_id
- k event.original
- k user.email
- k service
- # status

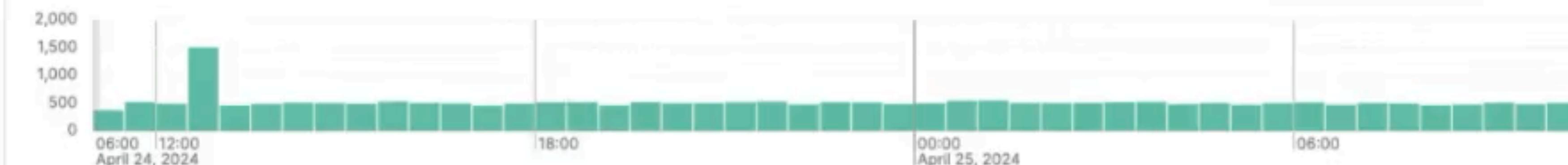
## Available fields

40

- @timestamp
- k @version
- k action
- # allocations
- k aws\_region
- k client\_request\_id
- k controller
- k data\_stream.dataset

Add a field

24,485 hits



Apr 24, 2024 @ 11:06:46.286 - Apr 25, 2024 @ 11:06:46.286 (interval: Auto - 30 minutes)

## Documents Field statistics

1 field sorted

↓ @timestamp



Document

<input type="checkbox"/> Apr 25, 2024 @ 10:32:36.000	@timestamp Apr 25, 2024 @ 10:32:36.000 @version 1 action create allocations 59,003,177 aws_region us-east-1 client_request_id 2f57b3e3-b1a2-4984-a653-2c55acd0a0b0 controller InvoicesController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:34.000	@timestamp Apr 25, 2024 @ 10:32:34.000 @version 1 action update allocations 83,424,491 aws_region eu-north-1 client_request_id a8a52202-ef85-454d-a74b-aff4dc937ea5 controller ConferencesController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:25.000	@timestamp Apr 25, 2024 @ 10:32:25.000 @version 1 action create allocations 89,231,346 aws_region us-east-1 client_request_id f8bf00d8-ba33-4974-8723-f41cc065629c controller ConferencesController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:24.000	@timestamp Apr 25, 2024 @ 10:32:24.000 @version 1 action create allocations 11,989,002 aws_region us-east-1 client_request_id 72f6594b-5f2d-4064-b62b-ed31f9846d7b controller ConferencesController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:19.000	@timestamp Apr 25, 2024 @ 10:32:19.000 @version 1 action index allocations 21,258,153 aws_region eu-north-1 client_request_id 950f6c12-17ac-40cb-a9d1-9eae1536aa06 controller TicketsController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:19.000	@timestamp Apr 25, 2024 @ 10:32:19.000 @version 1 action create allocations 41,620,789 aws_region us-east-1 client_request_id b4594c37-28e5-4dfb-bc50-664052fa91c6 controller AttendeesController...
<input type="checkbox"/> Apr 25, 2024 @ 10:32:17.000	@timestamp Apr 25, 2024 @ 10:32:17.000 @version 1 action show allocations 54,962,534 aws_region eu-central-1 client_request_id 33333333-3333-4333-a333-333333333333 controller EventsController...

Rows per page: 100

&lt; 1 2 3 4 5 &gt;

How many users are using  
this legacy API endpoint?  
What do they have in  
common?

D Visualize Library Create

Explore data in Discover

Inspect Share Settings

Save

logs-\* ▾



Filter your data using KQL syntax



Last 1 day



Search field names 0

Bar vertical stacked ▾



# Records

Available fields 39

@timestamp

@version

action

allocations

aws\_region

client\_request\_id

controller

data\_stream.dataset

data\_stream.namespace

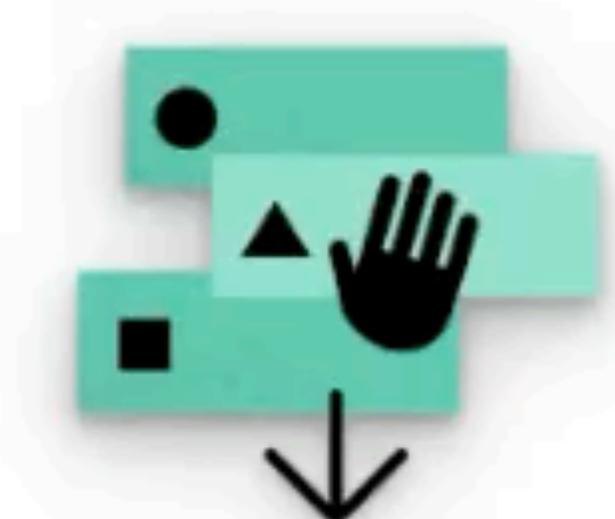
data\_stream.type

db\_runtime

duration

duration\_ms

enqueued\_by\_request\_id



Drop some fields here to start

Lens is the recommended editor for creating visualizations

Make requests and give feedback

Add layer



logs-\* ▾



Filter your data using KQL syntax

Last 26 hours



Search field names

0

# Records

Available fields ① 39

@timestamp

@version

action

allocations

aws\_region

client\_request\_id

controller

data\_stream.dataset

data\_stream.namespace

data\_stream.type

db\_runtime

duration

duration\_ms

enqueued\_by\_request\_id

Bar vertical stacked ▾



Drop some fields here to start

Lens is the recommended editor for creating visualizations

Make requests and give feedback

Bar vertical stacked ▾



logs-\* ▾



Horizontal axis

Optional

+ Add or drag-and-drop a field

Vertical axis

+ Add or drag-and-drop a field

Breakdown

Optional

+ Add or drag-and-drop a field

Add layer

Is this feature flag affecting  
the database performance

on revision 1b56dc in us-

east-1?

D Visualize Library Create

Explore data in Discover

Inspect Share Settings

Save

logs-\* ▾



Filter your data using KQL syntax

Last 26 hours



Search field names 0

# Records

Available fields 39

@timestamp

@version

action

allocations

aws\_region

client\_request\_id

controller

data\_stream.dataset

data\_stream.namespace

data\_stream.type

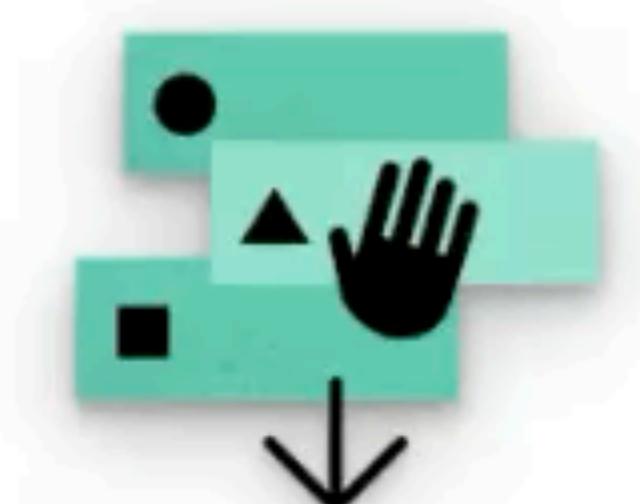
db\_runtime

duration

duration\_ms

enqueued\_by\_request\_id

Bar vertical stacked ▾



Drop some fields here to start

Lens is the recommended editor for creating visualizations

Make requests and give feedback

Add layer

Bar vertical stacked ▾

logs-\* ▾

Horizontal axis

Optional

+ Add or drag-and-drop a field

Vertical axis

Optional

+ Add or drag-and-drop a field

Breakdown

Optional

+ Add or drag-and-drop a field

Structured logs  
rock! 🤘

# Parting words

This is not new

# Heroku

## Use structured logging

Human readable logs are helpful, but structured logs add value, especially as an application's logs become more complex or as request throughput increases. Structured logging provides a well-defined format for your log entries so you can more easily parse, query, and process them for analytics.

The structure of log entries from the Heroku platform is called [logfmt](#), where each line consists of a single level of densely packed key/value pairs. This format is human readable and fairly easy to parse, if needed.

JSON is the de facto standard for structured logging, but consider using key=value pairs, XML, or another format for your application logs. Review the supported formats that your add-on provider can automatically parse.

# Stripe: Canonical Log Lines

**stripe** Products Solutions Developers Resources Pricing Contact sales > Sign in >

Blog / Engineering Stripe on Twitter >

## Fast and flexible observability with canonical log lines

July 30, 2019

 Brandur Leach  
API Experience

Logging is one of the oldest and most ubiquitous patterns in computing. Key to gaining insight into problems ranging from basic failures in test environments to the most tangled problems in production, it's common practice across all software stacks and all types of infrastructure, and has been for decades.

# Wide events

**All you need is Wide Events, not “Metrics, Logs and Traces”**



IVAN BURMISTROV

FEB 15, 2024

38

24

2

Share

...

"Both Shopify and 37signals  
have been using in-house  
structured logging for a  
long time"

— DHH

We are lagging  
behind

Let's fix this

# Rails builders

Try it, use it, spread the word

# Gem maintainers

Consider adding an optional structured  
logging option

This applies to Rails too

# Add structured logging next to developer logging #50452

Open

dhh opened this issue on Dec 26, 2023 · 5 comments



dhh commented on Dec 26, 2023

Member

...

When you want to aggregate the data from your Rails application into a data analysis platform, it's a real pain trying to parse the developer-focused log lines that Rails spits out by default. In addition to these log lines, which are great in development and for manual inspection, we should add structured json logging on a per-request basis that can easily be digested by analytics.

Both Shopify and 37signals have been using in-house structured logging for a long time, so extracting something from those battle-tested scenarios is the path we intend to go.

cc [@jeremy](#) [@rafaelfranca](#)



60



6



25

# Ask your logs

A large pile of cut logs, likely from a single tree, is shown. The logs are stacked in various orientations, revealing their circular cross-sections. The wood has a mix of light-colored sapwood and darker heartwood. Some logs show prominent growth rings and radial cracks. The scene is outdoors, with some green grass and twigs visible at the bottom.

Thank you