

Discover

Visualize

Dashboard

Timeline

Management

fields.requestId

fields.restaurantName

fields.userEmail

lambdaVersion

level

logGroup

logStream

tags

type

▼ March 30th 2018, 02:28:23.512 placing order...

Table

JSON

[View surrounding documents](#)

[View single document](#)

| | |
|-----------------------|---|
| @timestamp | March 30th 2018, 02:28:23.512 |
| fields.orderId | 15d0faea-0e62-53d8-863f-986680e2a1a8 |
| fields.requestId | a2bc369d-33b9-11e8-b454-2b4ce5bedc6d |
| fields.restaurantName | Fangtasia |
| fields.userEmail | theburningmonk@gmail.com |
| functionName | big-mouth-dev-place-order |
| lambdaVersion | \$LATEST |
| level | debug |
| logGroup | /aws/lambda/big-mouth-dev-place-order |
| logStream | 2018/03/30/[\$LATEST]d32999707b14449f8280d21a838572f8 |
| message | placing order... |
| tags | _logz_json_tcp_5050 |
| type | cloudwatch |

▶ March 30th 2018, 02:28:23.512 request body is a valid JSON

▶ March 30th 2018, 02:28:17.903 generated HTML [18318 bytes]

▶ March 30th 2018, 02:28:17.870 loaded 8 restaurants

▶ March 30th 2018, 02:28:17.852 fetched 8 restaurants

▶ March 30th 2018, 02:28:16.906 loaded HTML template

traffic

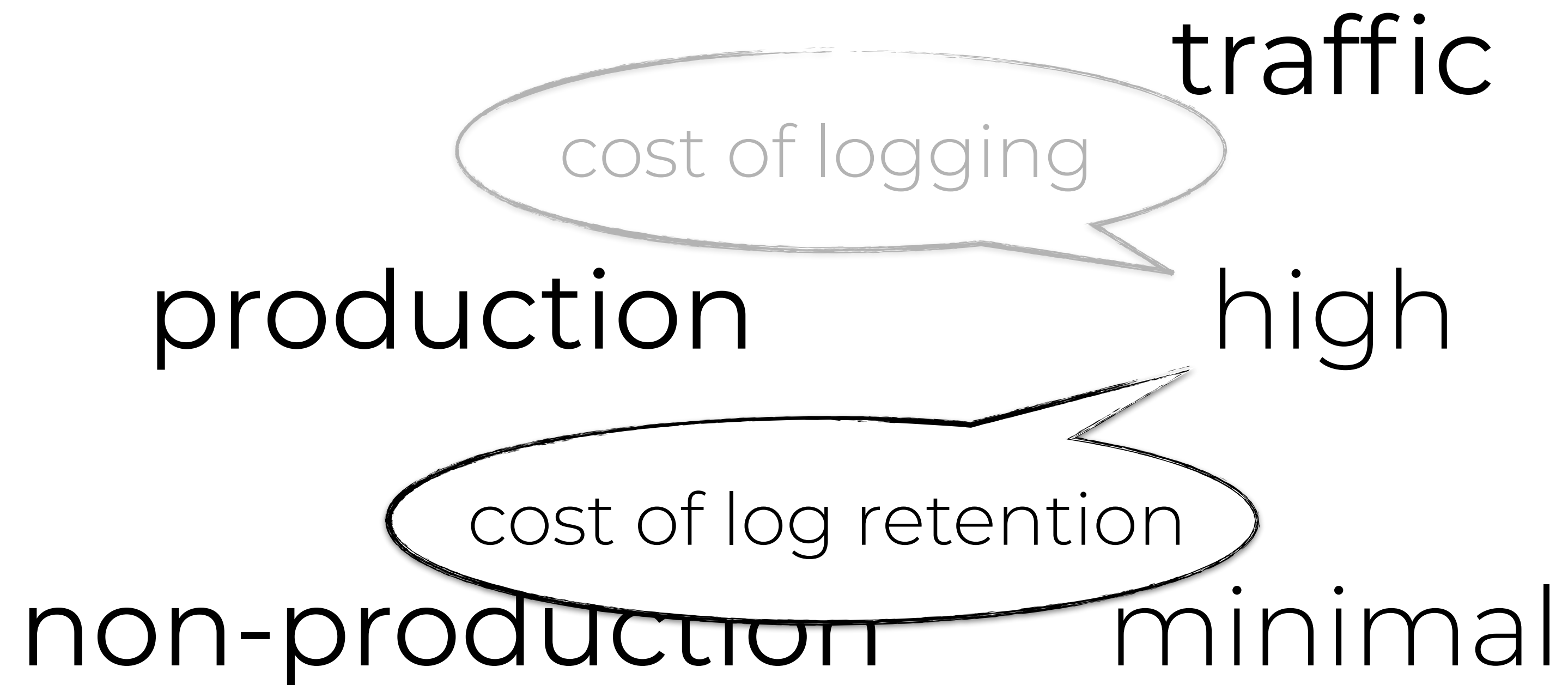
production

high

non-production

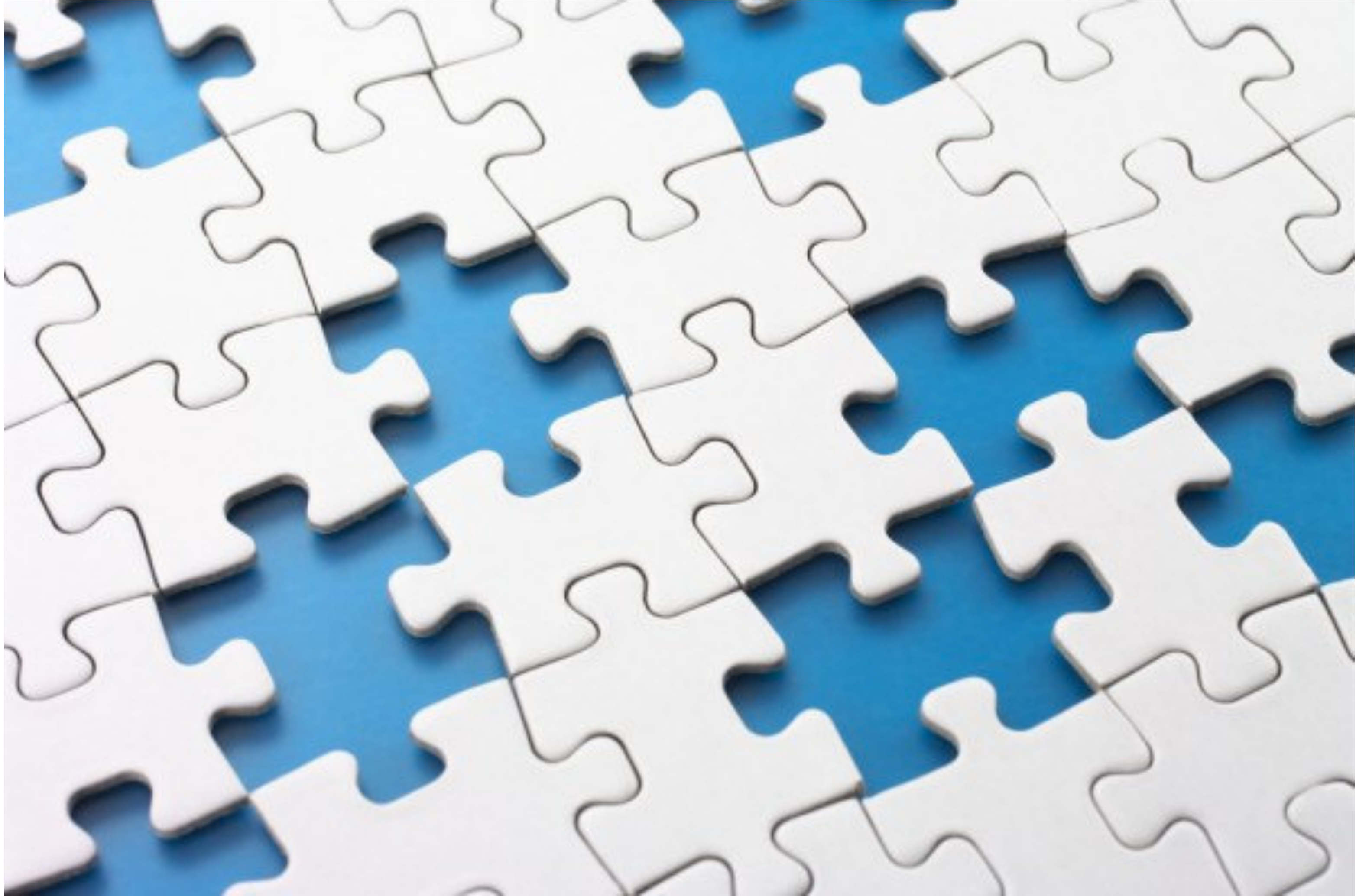
minimal

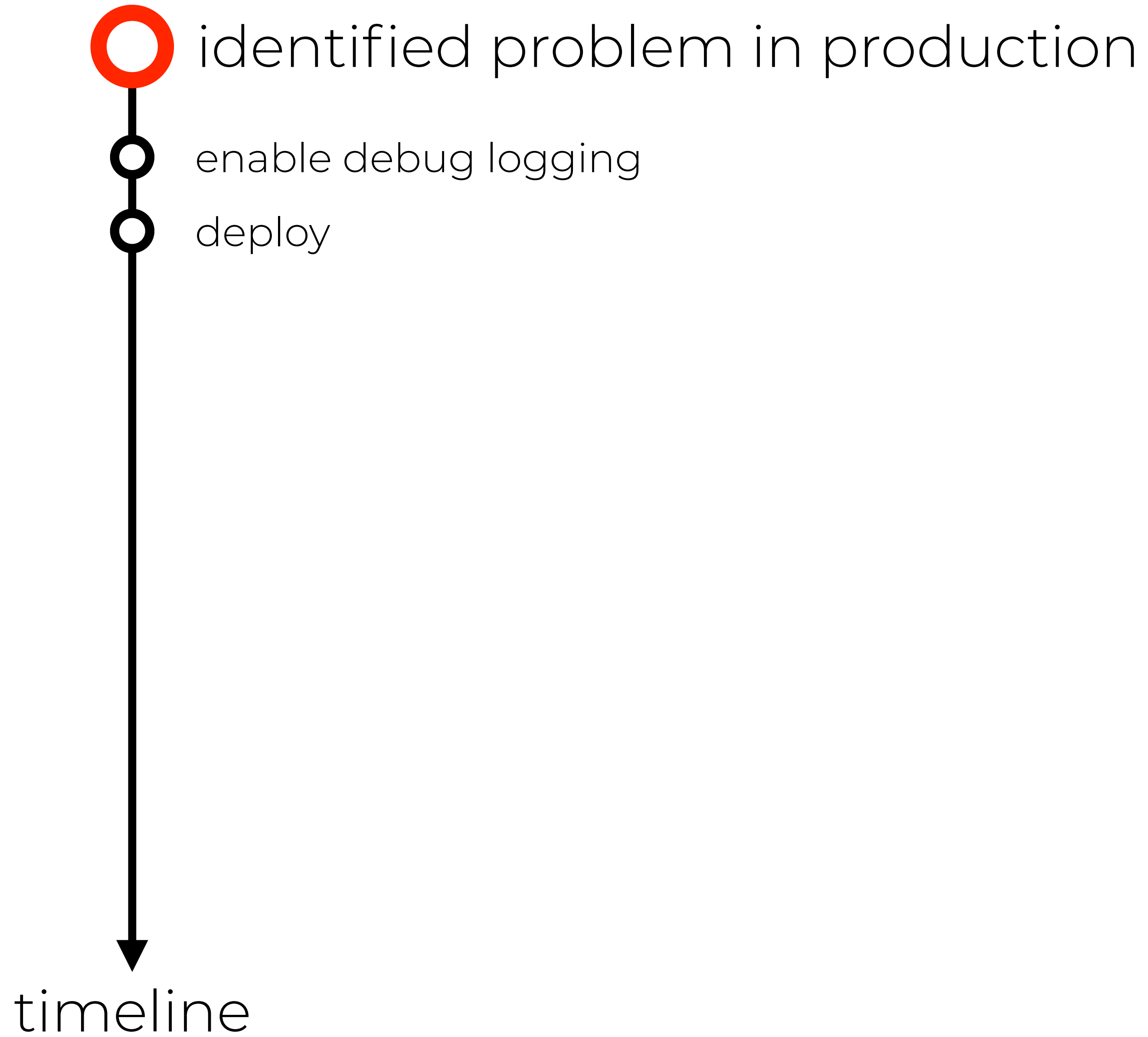
| | |
|-----------------|---------|
| | traffic |
| cost of logging | |
| production | high |
| non-production | minimal |





DON'T leave debug logging **ON** in production



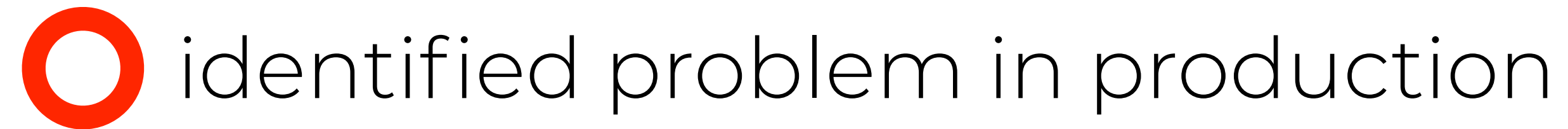


identified problem in production

enable debug logging

deploy

timeline



identified problem in production

enable debug logging

deploy

analyse logs

identify problem

timeline

identified problem in production

enable debug logging

deploy

analyse logs

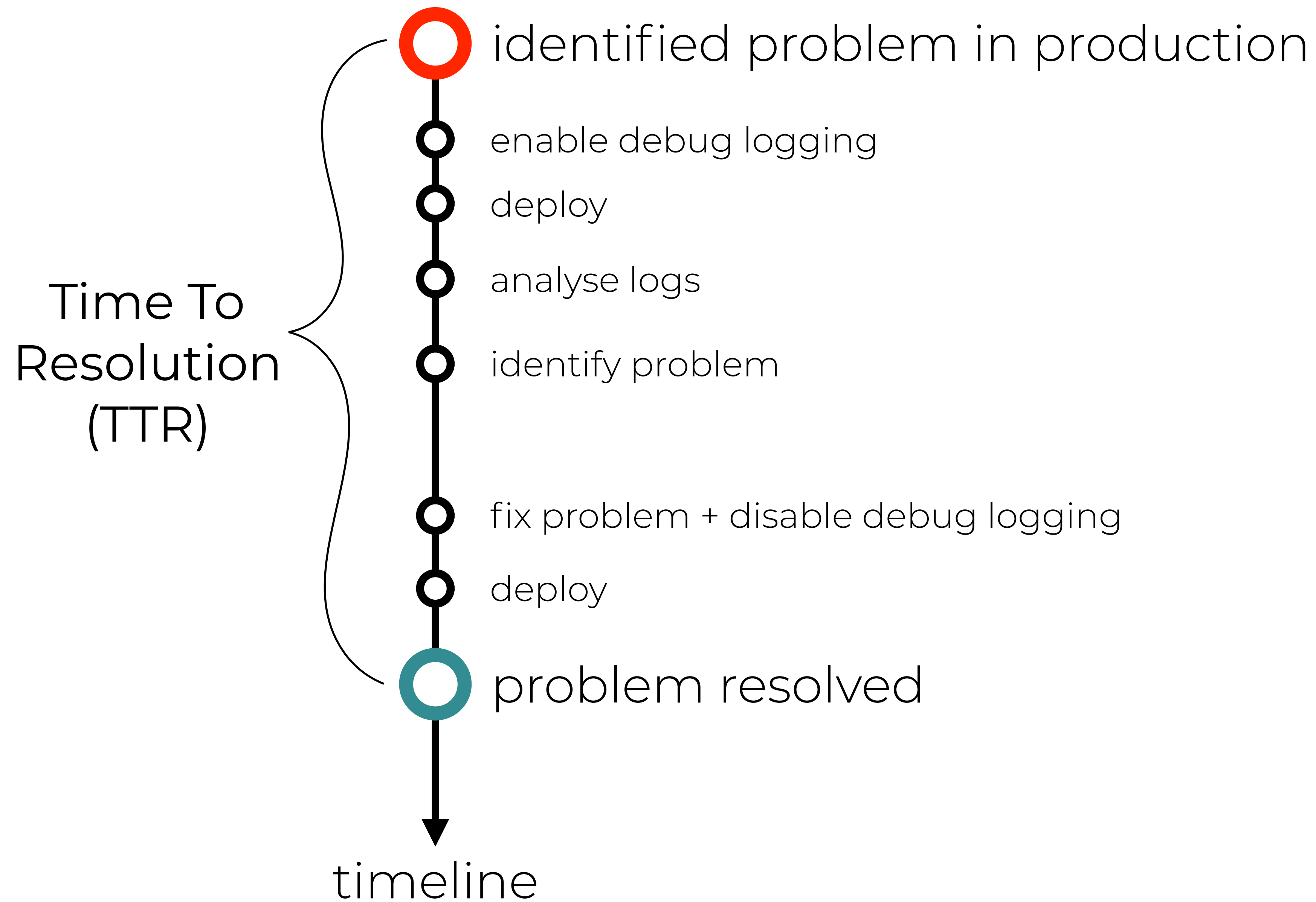
identify problem

fix problem + disable debug logging

deploy

problem resolved

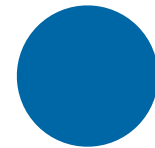
timeline



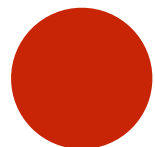
observability



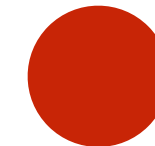
sampled debug logs



no debug logs



all debug logs



volume of logs

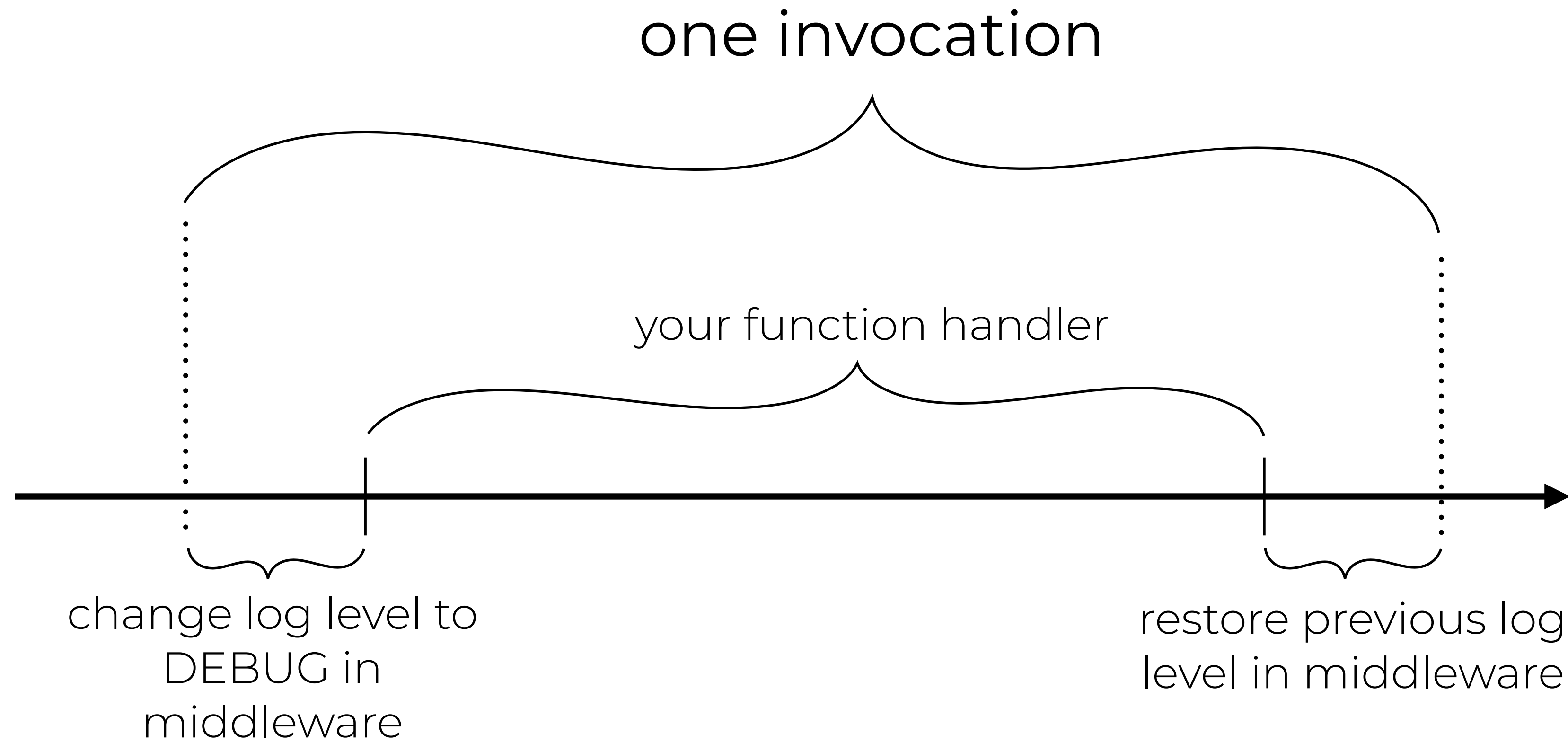


what we'll do in this module

- meet Middy, the middleware engine

what we'll do in this module

- meet Middy, the middleware engine
- sample debug logs by invocation



taking it further

- flush all debug logs when an invocation fails