Application Performance Monitoring using DataDog

Case Study of Performance monitoring for a client in the Media and Mobile domain

Application Monitoring

- Ensure that a software application processes and performs in an expected manner and scope

APM

Runtime Metrics

Health of the application

Health of infrastructure

Provide system/application feedback

Continuous Improvement

What is Datadog?

- Monitoring as a Service
- Agent Based
- Python
- Integrations
- Dashboards
- Tagging
- Alerts
- Checks

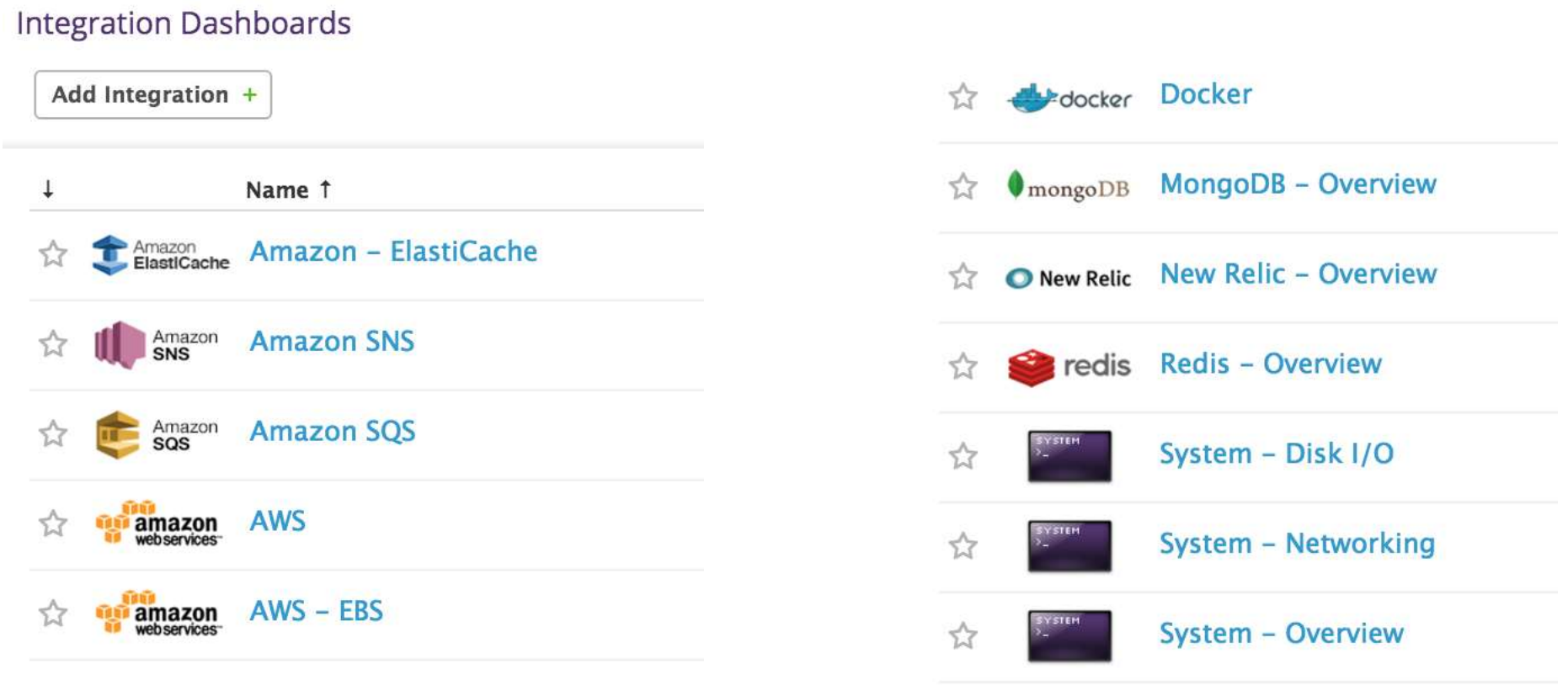
Environment

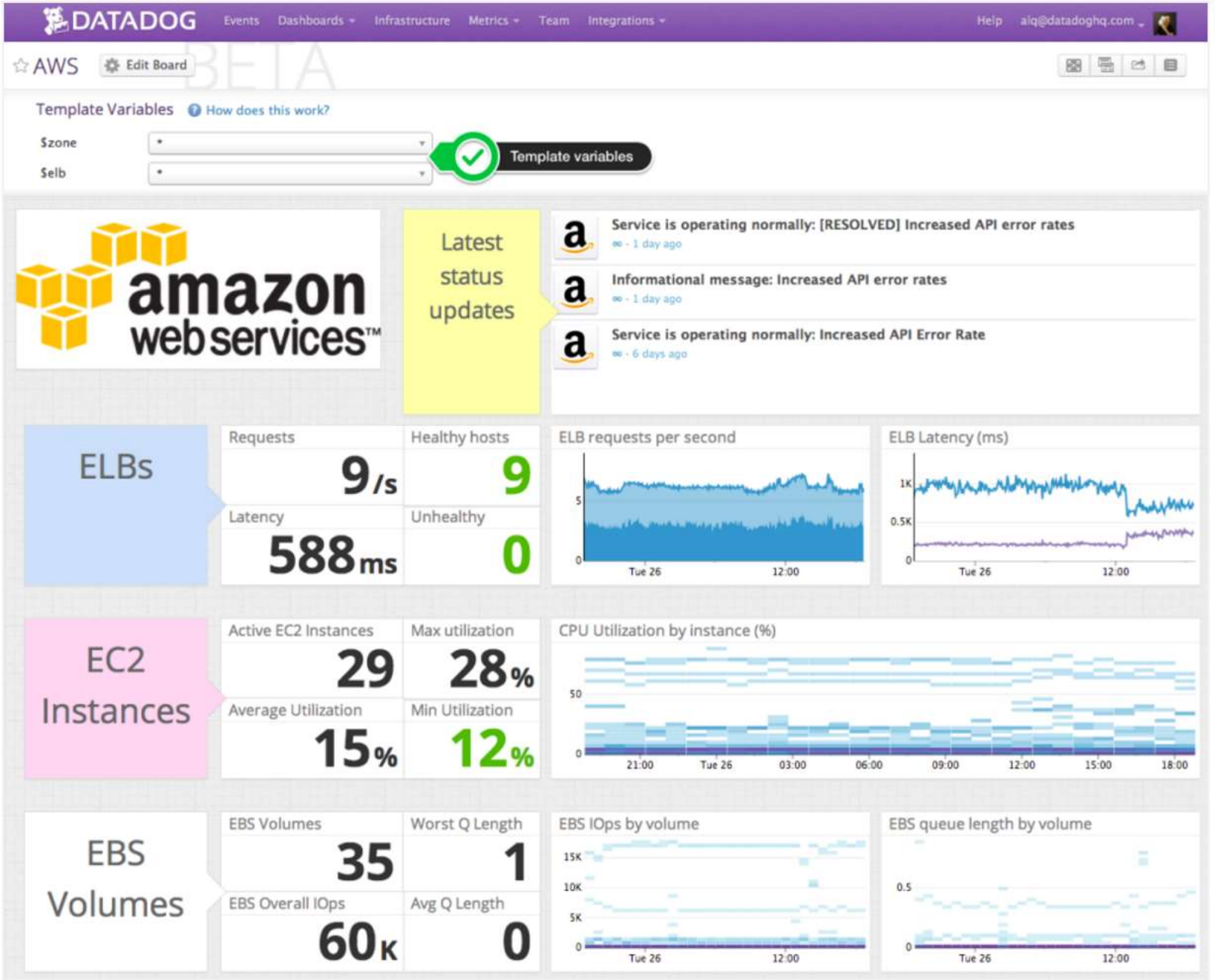
Challenge

- Monitor System Health
- Monitor Redis, MongoDB
- Application (API Server) runs as docker container
- *Monitor application performance*
- Logging slowed down performance
- *Rapid development, No feedback mechanism about the application performance*
- *Insight in application needed by management*
- *Insight needed by support, devops and developers*

Integration Dashboards

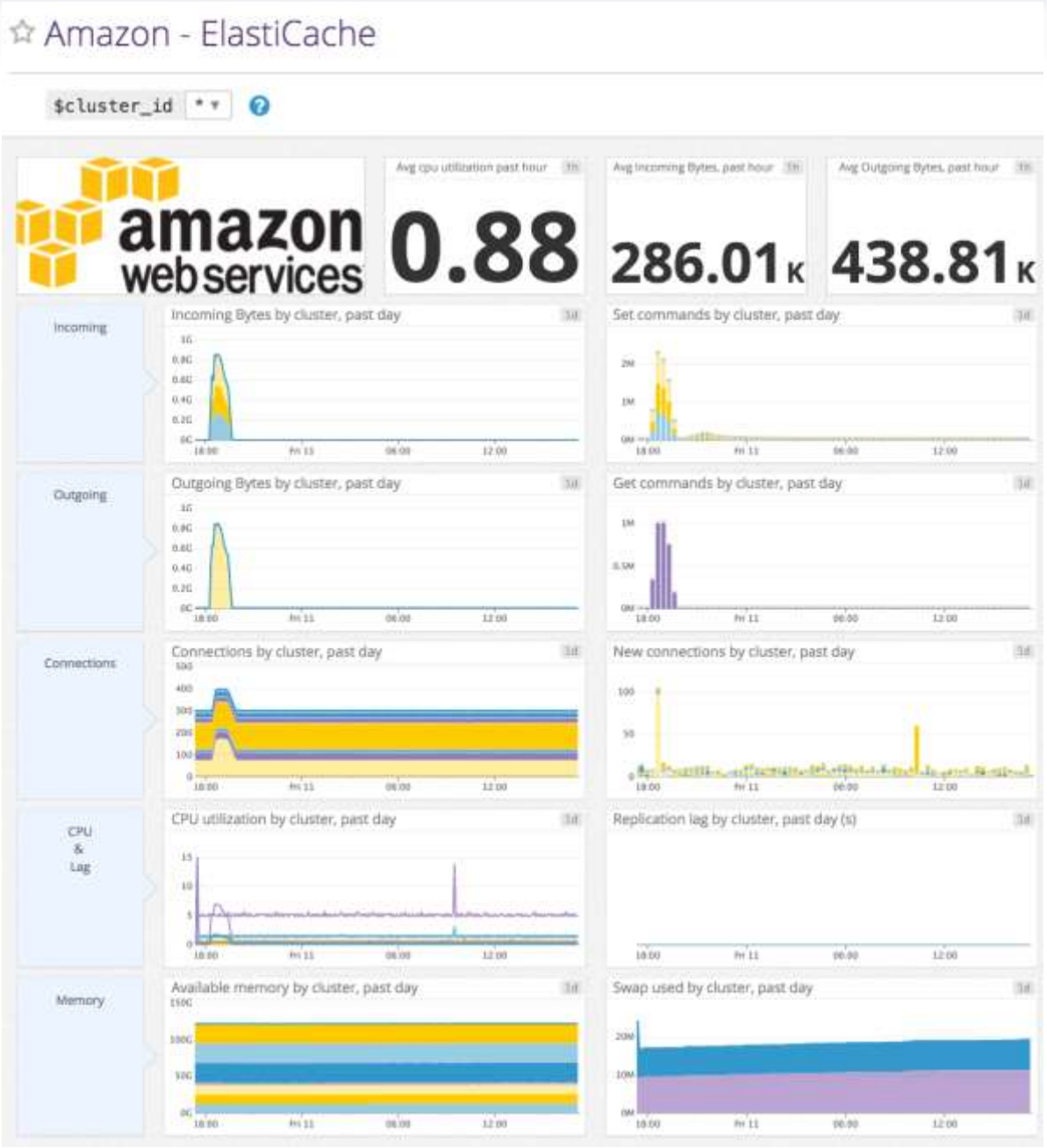
WhiteHedge





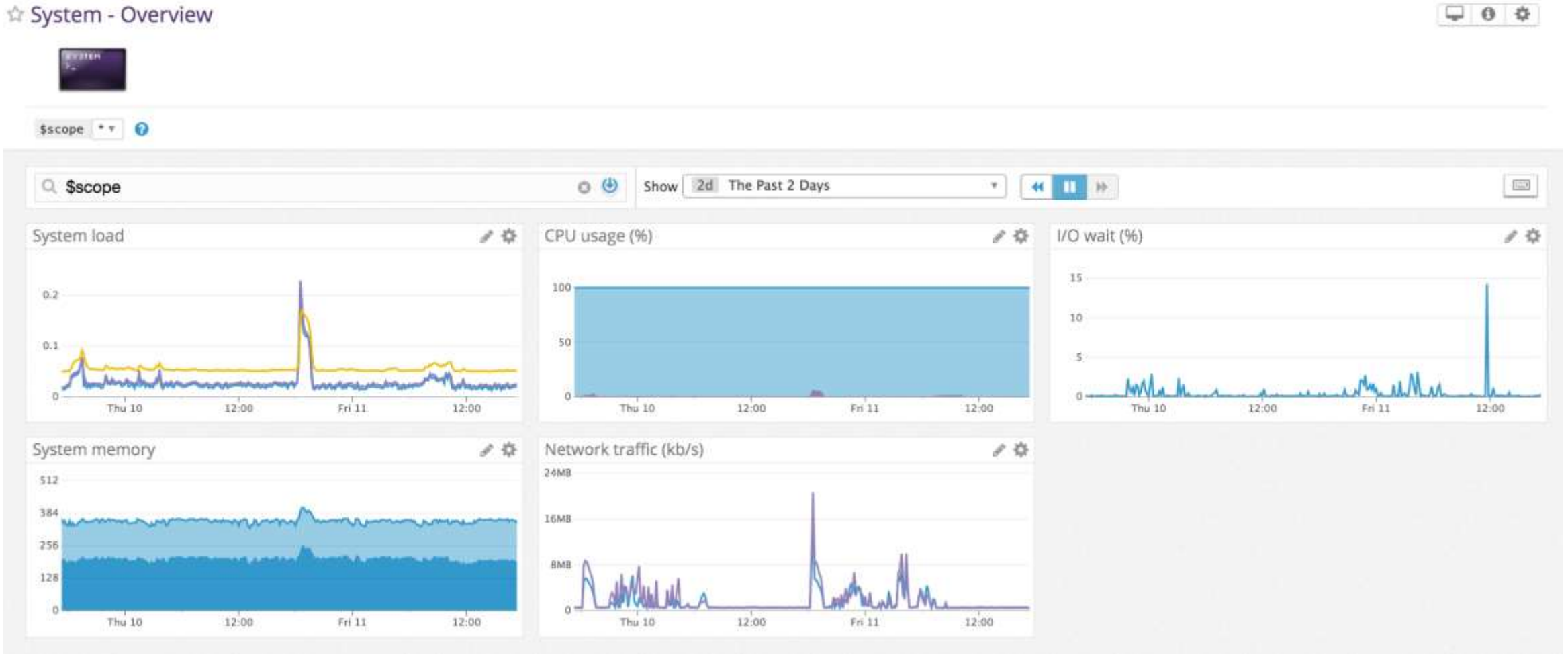
info@WhiteHedge.com

Amazon - ElastiCache



System Overview

WhiteHedge

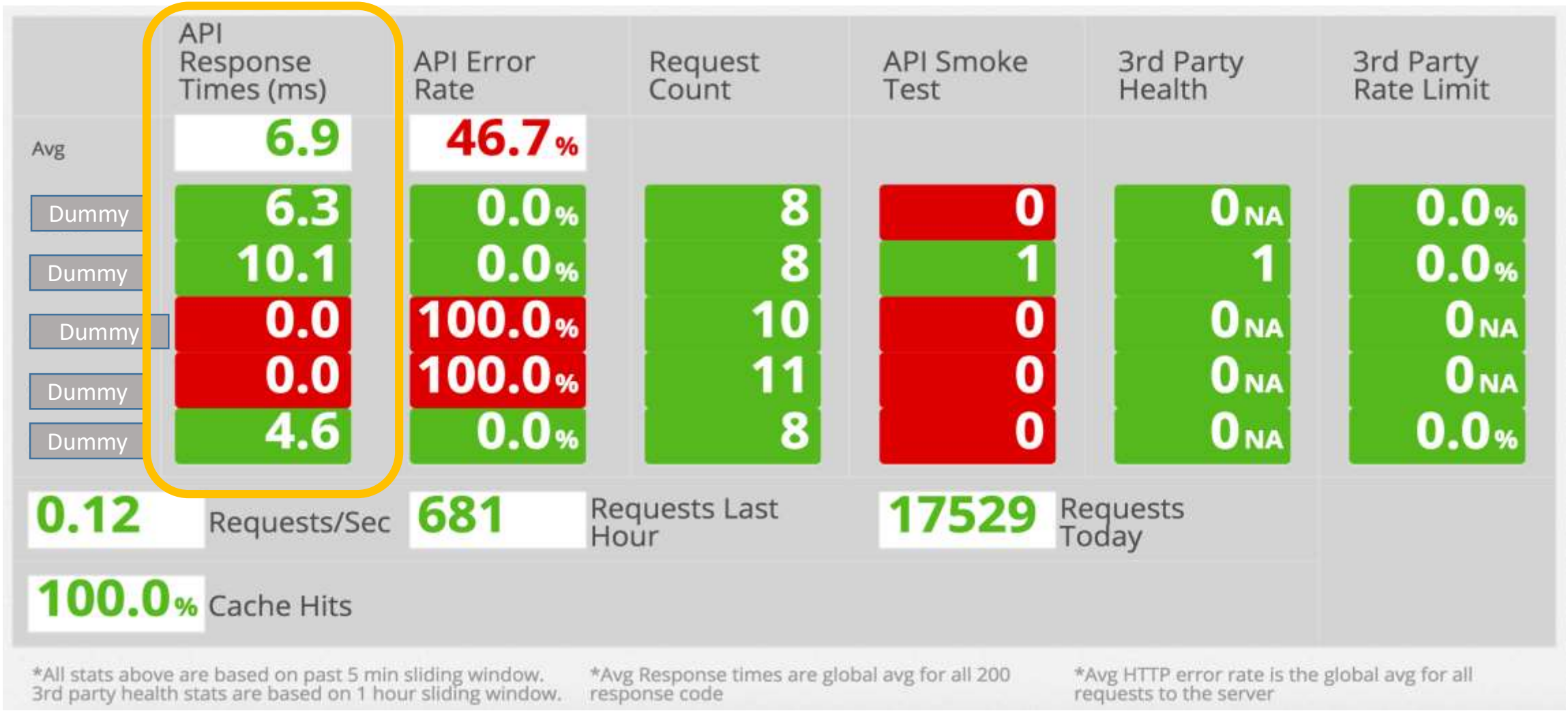


But…

- How to impress the management?! ☺
- How to deal with X environments and Y versions?
- How to measure performance, without affecting performance?

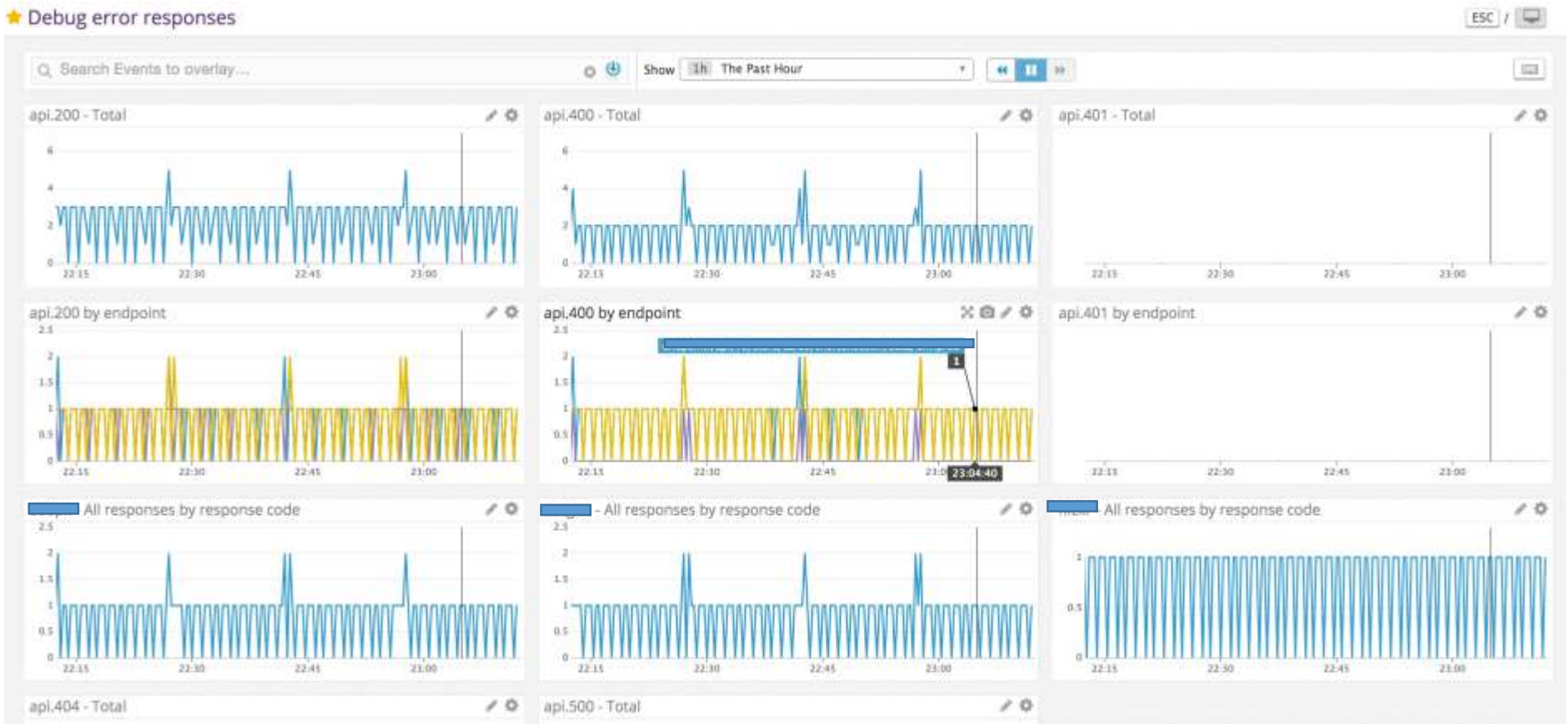
Overview Dashboard

WhiteHedge



Debug: API Error Rate

WhiteHedge



Which endpoint is having more errors? Which error codes are being thrown? Search logs in Loggly if needed.

- We can code:
  - Instance configuration
  - Infrastructure
  - Deployments
- Why not monitoring?!
- Datadog has great API

Datadog module for nodejs


```
1   var StatsD = require('node-dogstatsd').StatsD;
2   var dogstatsd = new StatsD();
3
4   var tag1 = "env:" + process.env.NODE_ENV;
5   var tag2 = "endpoint:dummy";
6
7   dogstatsd.increment("task.hits", 1, [ tag1, tag2 ]);
```

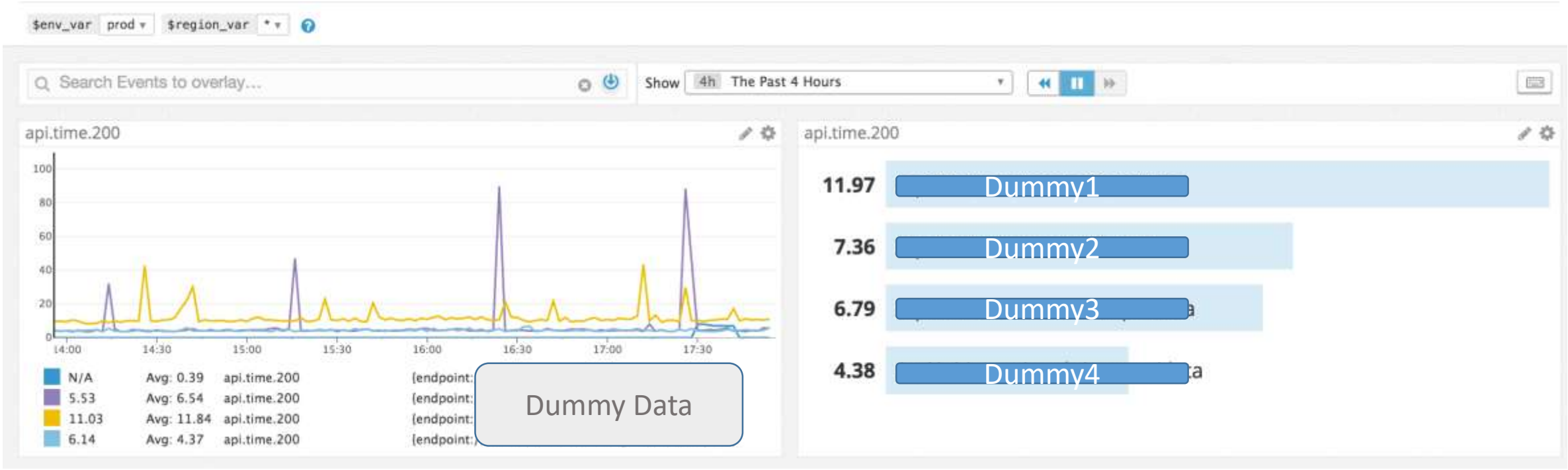
Measuring http response times from application code

WhiteHedge

```
1   if (!(api.config.env.isLocal || api.config.env.isTest)) {
2       var tag1 = "response_code:" + raw.responseHttpCode;
3       var tag2 = "env:" + process.env.NODE_ENV;
4       var tag3 = "endpoint:" + raw.parsedURL.pathname;
5
6       var metric = raw.parsedURL.pathname;
7
8       // Send API metrics (count)
9       metric = "api." + raw.responseHttpCode;
10      dogstatsd.increment(metric, 1, [ tag3, tag2 ]);
11
12      // Send response times
13      metric = "api.time." + raw.responseHttpCode;
14      dogstatsd.gauge(metric, data.duration, [ tag2, tag3 ]);
15  }
```

Plot it!

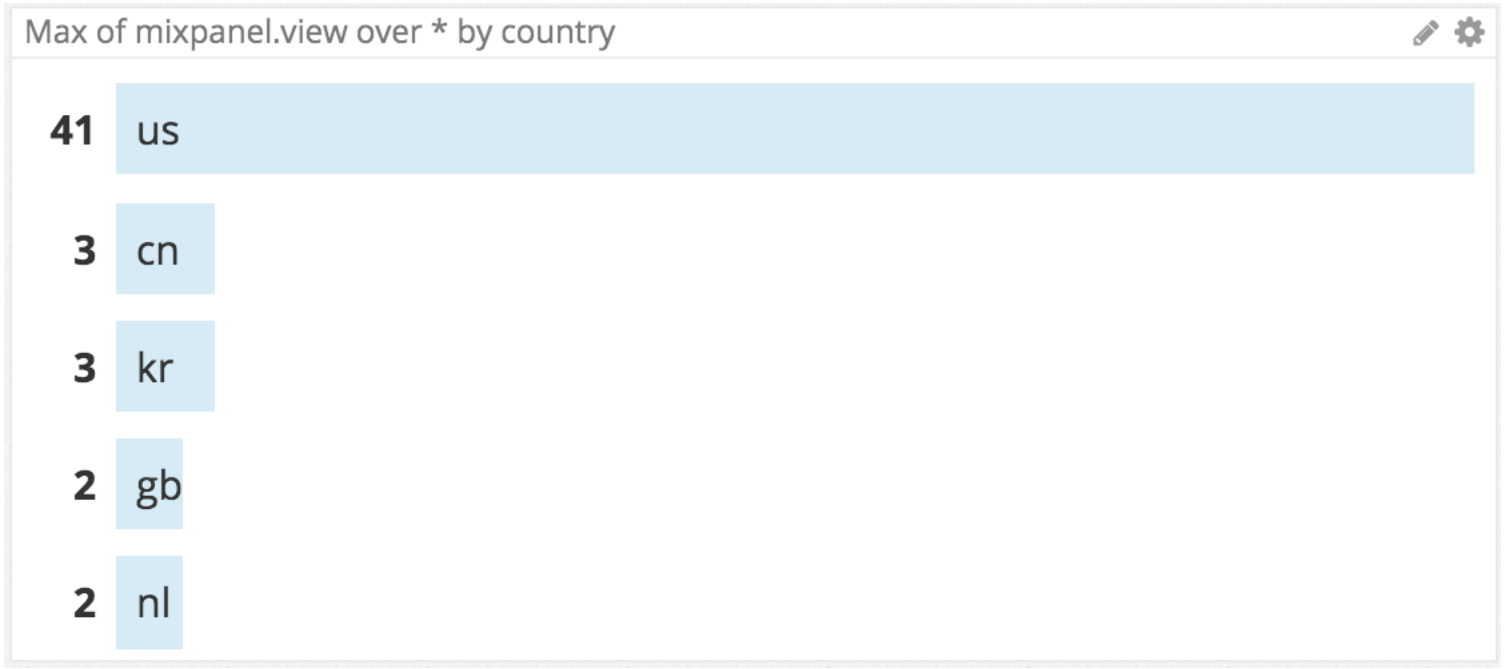
WhiteHedge



What about performance while tracking performance?

- Metrics from code are sent to local datadog agent using UDP
- Local datadog agent syncs the metrics to the datadog server
- Datadog dashboard reflects the metrics with some delay
- Application performance does not get affected

- Metrics are sent from device (mobile/tablet/web) to MixPanel
- Real Data
- Integrate MixPanel with DataDog



Top 5 users of the application by Country

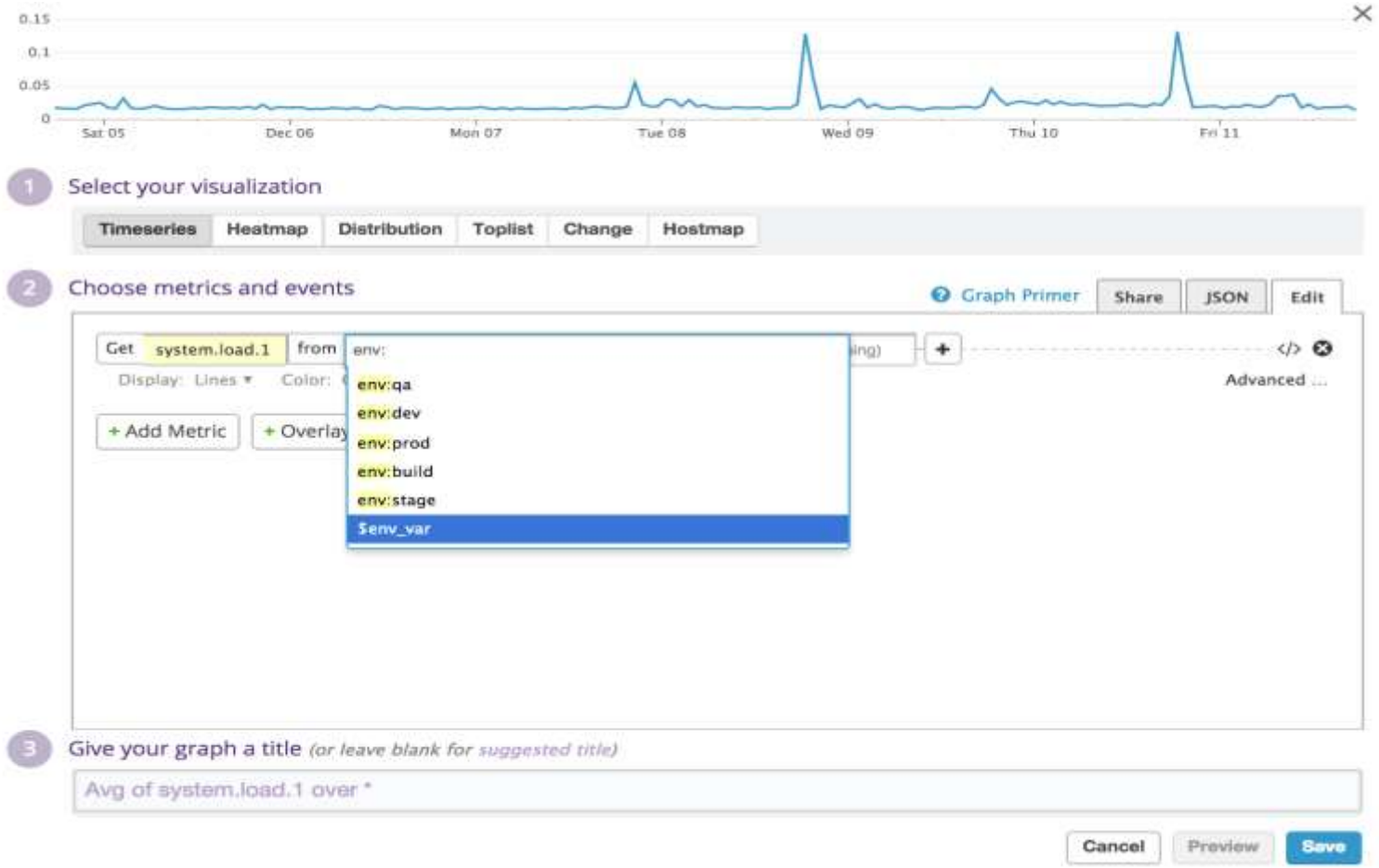
Extending the approach to Test Env


```
1  if (!(api.config.env.isLocal || api.config.env.isTest)) {
2      var tag1 = "response_code:" + raw.responseHttpCode;
3      var tag2 = "env:" + process.env.NODE_ENV;
4      var tag3 = "endpoint:" + raw.parsedURL.pathname;
5
6      var metric = raw.parsedURL.pathname;
7
8      // Send API metrics (count)
9      metric = "api." + raw.responseHttpCode;
10     dogstatsd.increment(metric, 1, [ tag3, tag2 ]);
11
12     // Send response times
13     metric = "api.time." + raw.responseHttpCode;
14     dogstatsd.gauge(metric, data.duration, [ tag2, tag3 ]);
15 }
```

Datadog Tags

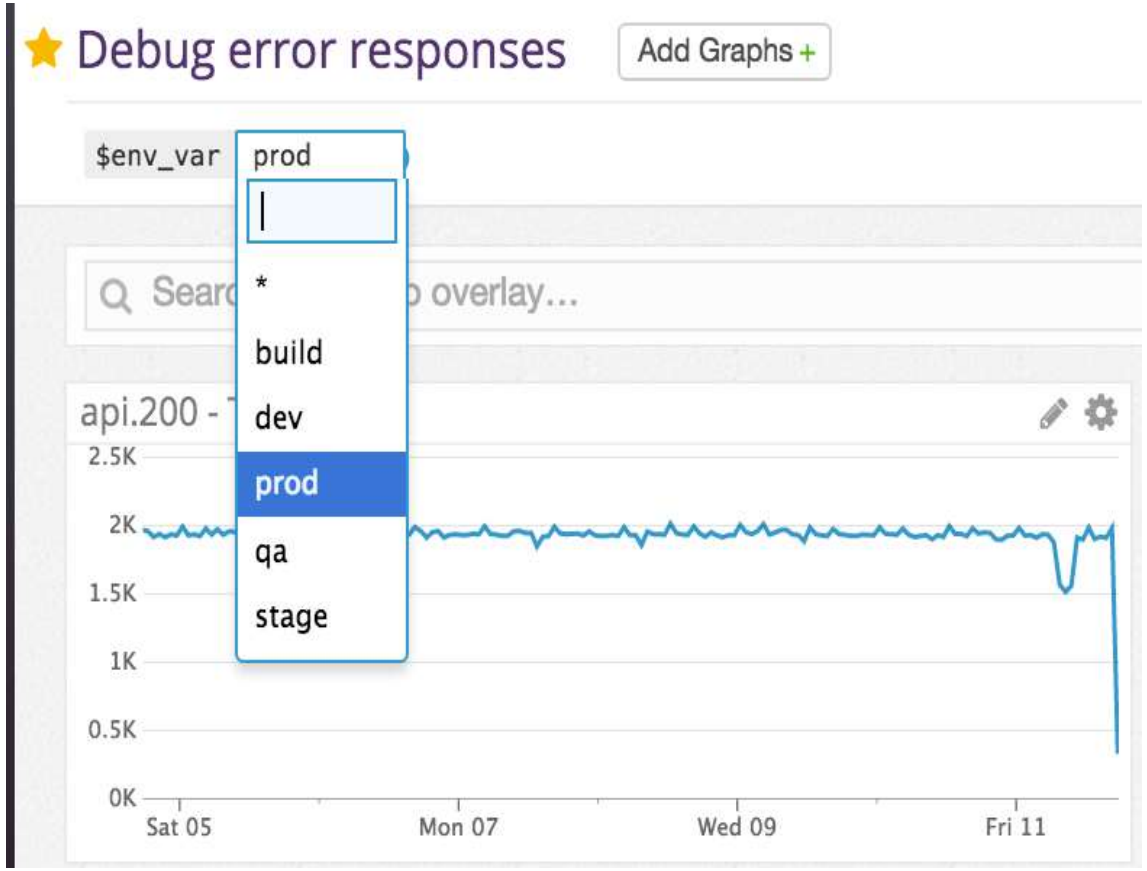
WhiteHedge

- Inherited from Integrations
- Custom tags



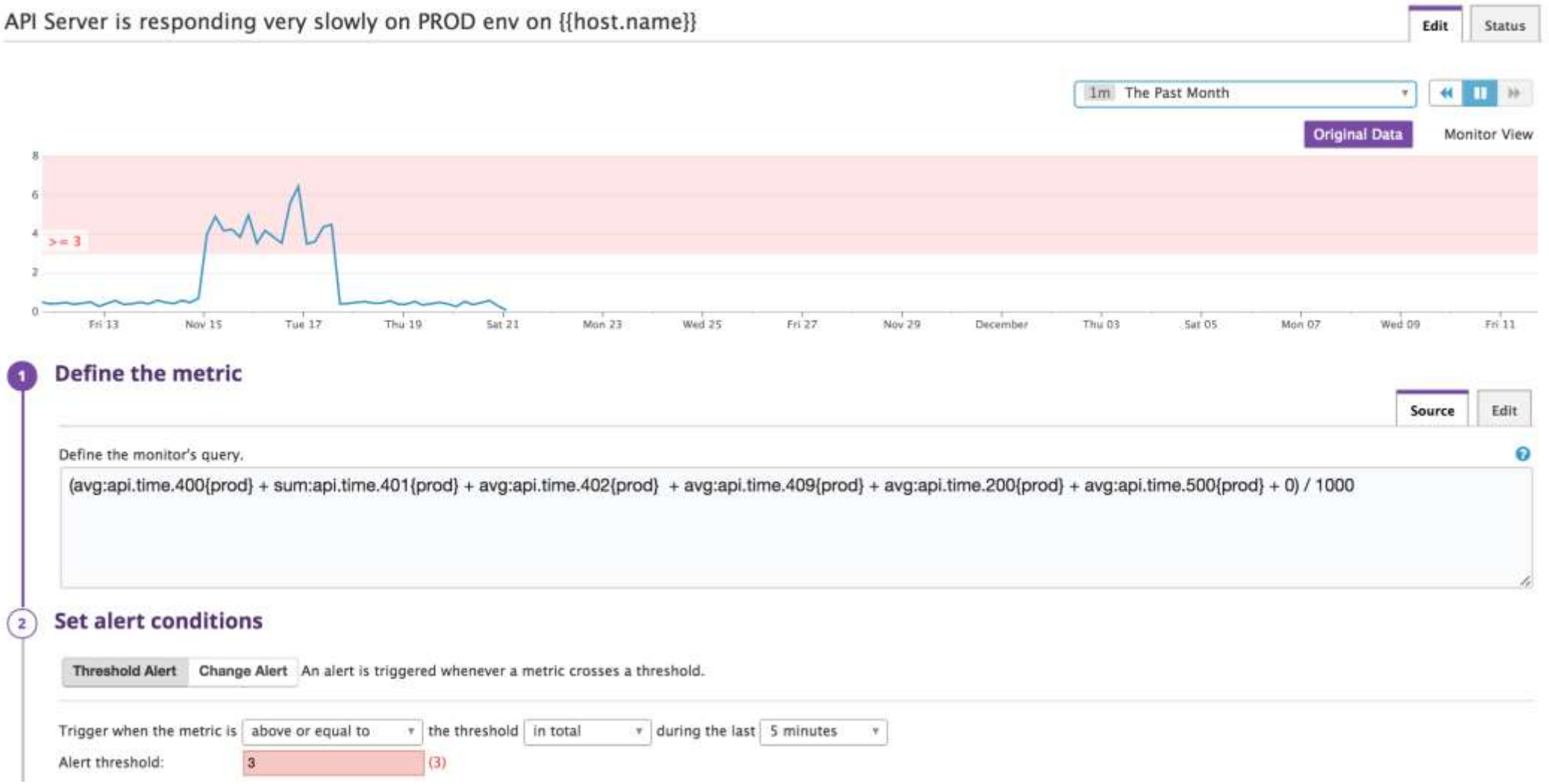
Templated Dashboards

- Dashboard variables
- Dynamically explore metrics



Alerts

WhiteHedge



Datadog Checks

- Collect metrics from datadog agent check
- Out of the box agent checks
- Custom agent checks in Python
- Interesting use cases:
  - Keep alive check (Service is up)
  - Network Check (HTTP, TCP)
  - Validate response for expected data

Using Datadog **we could…**

WhiteHedge

- Monitor dynamic infrastructure
- Monitor system health
- Monitor application availability
- Monitor application performance
- Show application & infra health graphically
- Provide feedback about health of system

GET IN TOUCH WITH US

Go Ahead, Don't Hesitate!

http://www.whitehedge.com/devops.html

http://www.whitehedge.com/docker-microservices/