Systems Introspection

A story about the tools we use to determine if the system is running properly

Me

@andrewhowdencom

Anything useful will be posted to Twitter.

Software engineer @ Sitewards

I built much of this stuff

Please Interrupt.

Please Contribute!

Thanks <3

Behrouz · Anton

Problem #1: Node is dead

It starts with an alert

What is “Down” even?

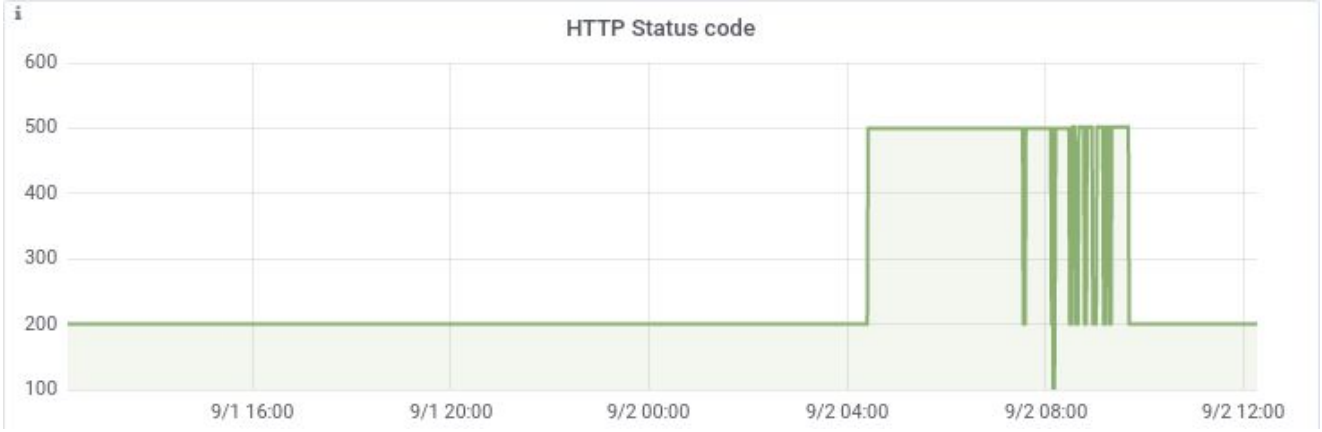
It’s down

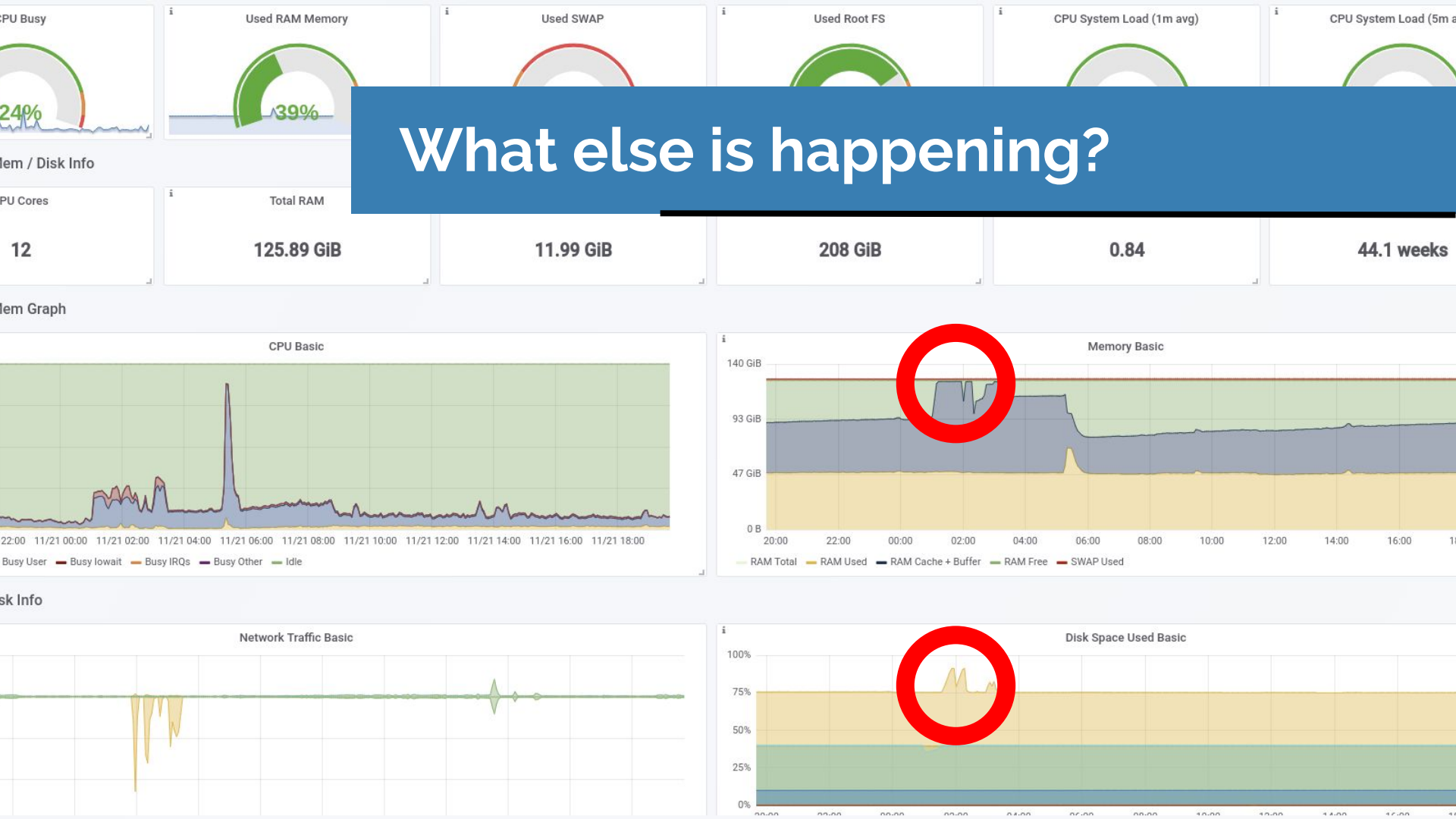
```
ALERT InstanceDown
  IF (probe_success{job="blackbox"} == 0) or (probe_success{job="auth_blackbox"}
== 0)
  FOR 5m
  ANNOTATIONS {summary="Address {{ $labels.instance }} appears to be down. See
https://wiki.sitewards.net/index.php/Instance_Down"}
```

“probe_success” means:

- The site responded within 5 seconds
- With a 200 status code
- Over a valid SSL certificate

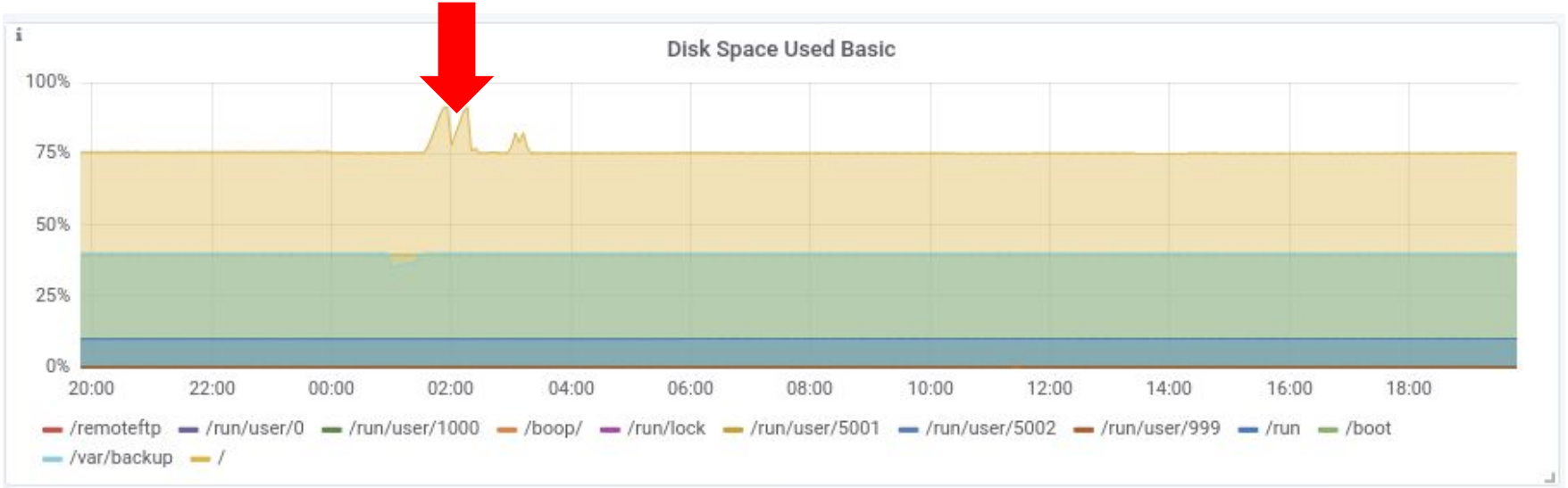
Application level failure





What else is happening?

I see you, full disk



Problem Solved

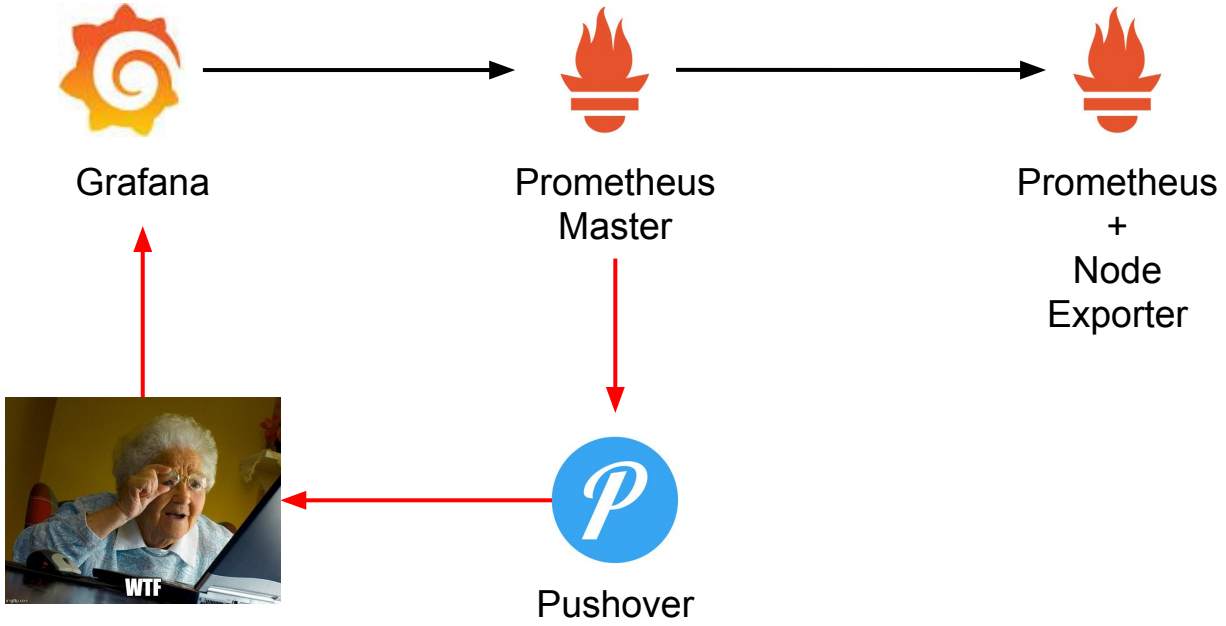
$ sudo ncdu
$ sudo rm -rf /var/lib/docker

- Easy to understand
- Open to all developers
- Gives historical context of problems
- Allows setting defined alerts for known error conditions

Finding full disks

Pushover	Prometheus	Grafana
Sends alerts to my phone. Pretty cheap.	Collects the metrics, stores them in its database Fires alerts (via AlertManager)	Makes pretty dashboards

Time series monitoring stack



Lessons (Time series data)

- It takes time to learn to use time series data
- It’s worth while
- Dashboards are a good intro
- The prometheus query tool is where you end up

Lessons (Alerts)

Alerts need to be:

- Specific; They need to say what's wrong exactly
- Actionable; If you're ignoring them they're not an alert they're a shitty log
- Relevant; If you're the wrong person for the alert, find the right person

Alert fatgiue kills (systems)!

Get time series data!!

Problem #2: Critical form is broken

It starts with a phone call

“Uhm the diesel checkout seems to be broken”

What's the next step?

Logs


```
{
 "time": "2018-11-21T19:23:39+00:00",
 "type": "magento",
 "environment": "production",
 "host": "www1.nope.de,
 "service": "magento",
 "pid": "",
 "request_id": "",
 "payload": {
  "version": "1.0.0",
  "severity": "CRIT",
  "store_code": "default",
  "request_url": "",
  "remote_address": "91.137.96.100",
  "file": "web/app/code/core/Mage/Core/Block/Template.php",
  "line_number": "243",
  "message": "Not valid template
file:frontend/base/default/template/nope/customer/account/dashboard/hello.phtml"
 }
}
```


Querying logs by error level

```
1 resource.type="gce_instance"
2 resource.labels.instance_id="fluentd-aggregation"
3 severity="CRITICAL"
```

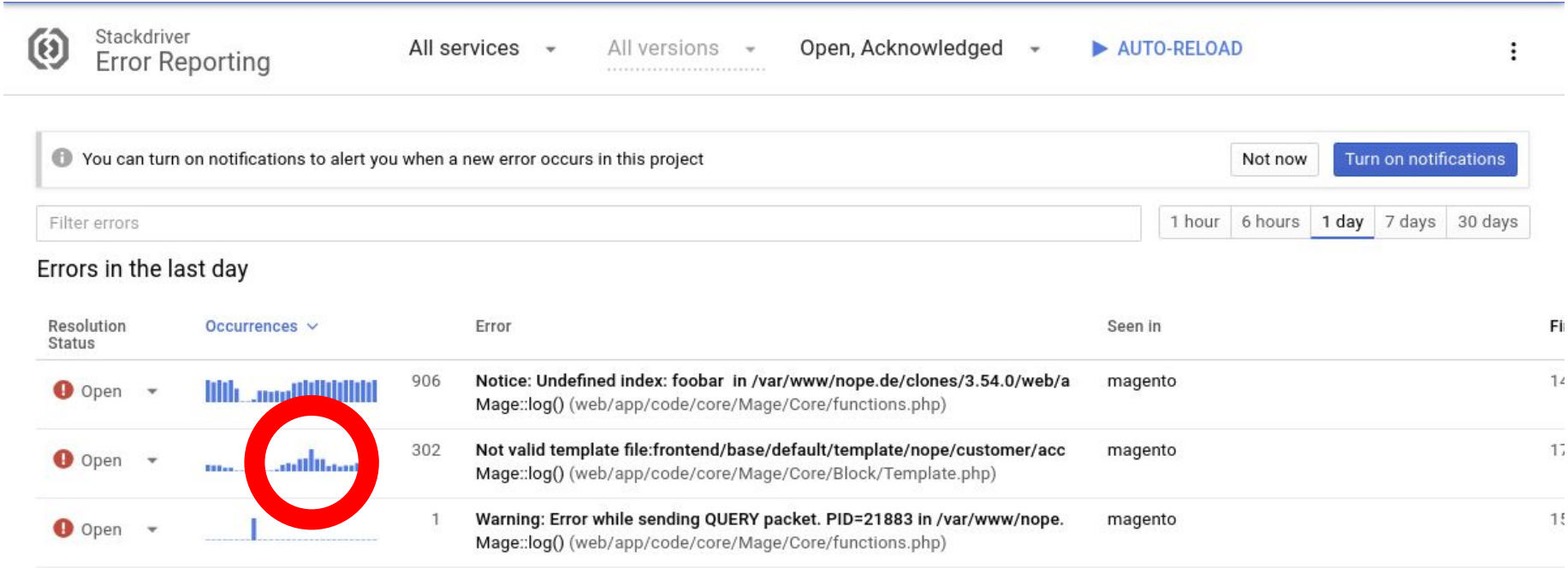
Submit Filter | Last hour | Jump to n

Showing logs from **the last hour** ending at 21:28

2018-11-21 20:29:48.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:33:39.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:36:48.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:36:59.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:37:55.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:43:22.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:44:14.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:45:54.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:48:59.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:49:45.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:52:00.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 20:59:14.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:03:33.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:10:59.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:15:46.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:17:10.000 UTC+2 pam_unix(sudo:auth): auth could not identify password
2018-11-21 21:19:15.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:23:08.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml
2018-11-21 21:23:39.000 UTC+2 Not valid template file:frontend/base/default/template/ customer/account/dashboard/hello.phtml

Load newer logs

I see you, faulty template file*



Problem Solved

(Rolled back release)

- Does not require access to production
- Can be searched en-masse by log attributes
- Faults can be automatically detected

FInding Faults

systemd-journald	fluentbit	Google Cloud Logging
Collects all logs on machine to a central location via Syslog interface	Sends logs to Google Cloud	Analyses Logs

Requires binary (json) logging

text/plain	application/json
12:58:00 “Hello, World” + Easy to read + What you’re used to - Hard to parse - Hard to handle when there’s 30,000 logs - Hard to analyse	{“time”: “12:58:00”, “message”: “Hello, World”} + Easy to analyse/handle + Can be read with `jq` + Well supported - Hard to read in less without jq

Lessons

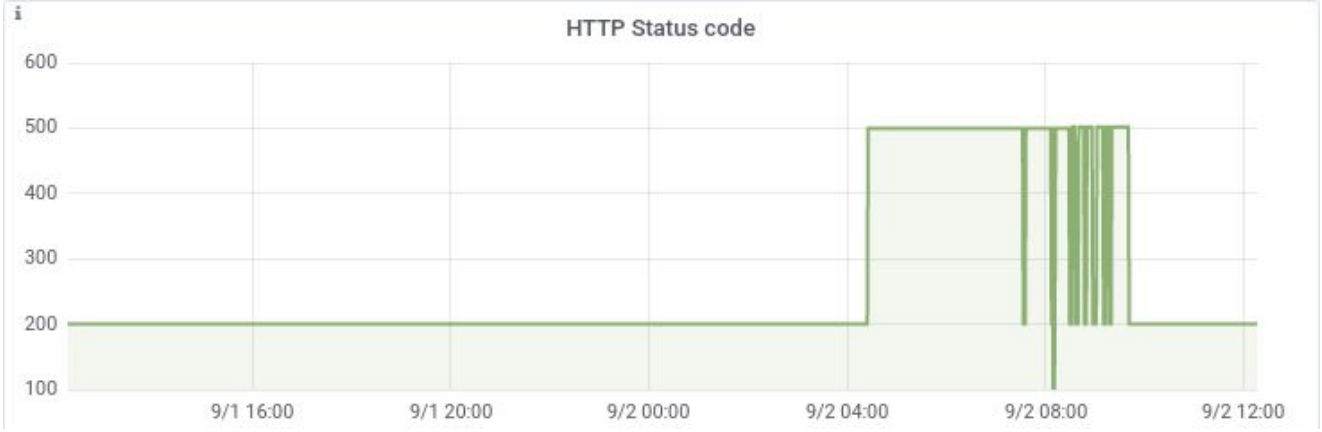
- Finding a binary log format to use consistently is hard
- I now adopt Googles
- It can take a while to get used to json logging

Log in JSON and forward!!

Problem #3: Cascading failure due to 3rd party

It starts with an alert

Application level failure (Again)



What else is happening?

Basic CPU / Mem / Disk Gauge

CPU Busy

Used RAM Memory

Used SWAP

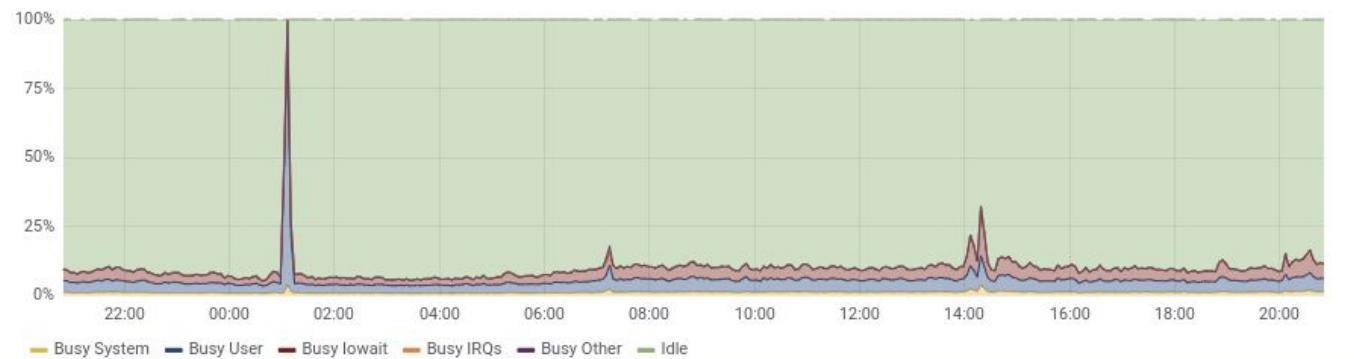
Used Root FS

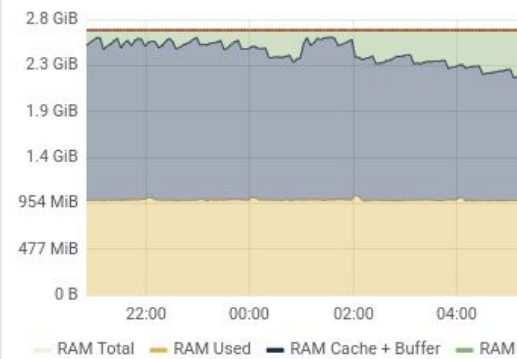
Basic CPU / Mem / Disk Info

CPU Cores	Total RAM	Total SWAP	Total RootFS
1	2.69 GiB	0 B	194 GiB

Basic CPU / Mem Graph

CPU Basic





Basic Net / Disk Info

Logs are showing … MySQL connections?

```
SQLSTATE[08004] [1040] Too many connections";i:1;s:3490:"#0
/var/www/__FILEPATH__ /web/lib/Zend/Db/Adapter/Pdo/Mysql.php(111):
Zend_Db_Adapter_Pdo_Abstract->_connect()
```

But connections are sleepy

```
# show full processlist; # edited for brevity

+------+---------+---------+------+-------+------------------+
| Id   | db      | Command | Time | State | Info             |
+------+---------+---------+------+-------+------------------+
| 2047 | myDB    | Sleep   |   81 |       | NULL             |
...
| 2049 | myDB    | Sleep   |   81 |       | NULL             |
| 2050 | myDB    | Sleep   |   81 |       | NULL             |
+------+---------+---------+------+-------+------------------+
```

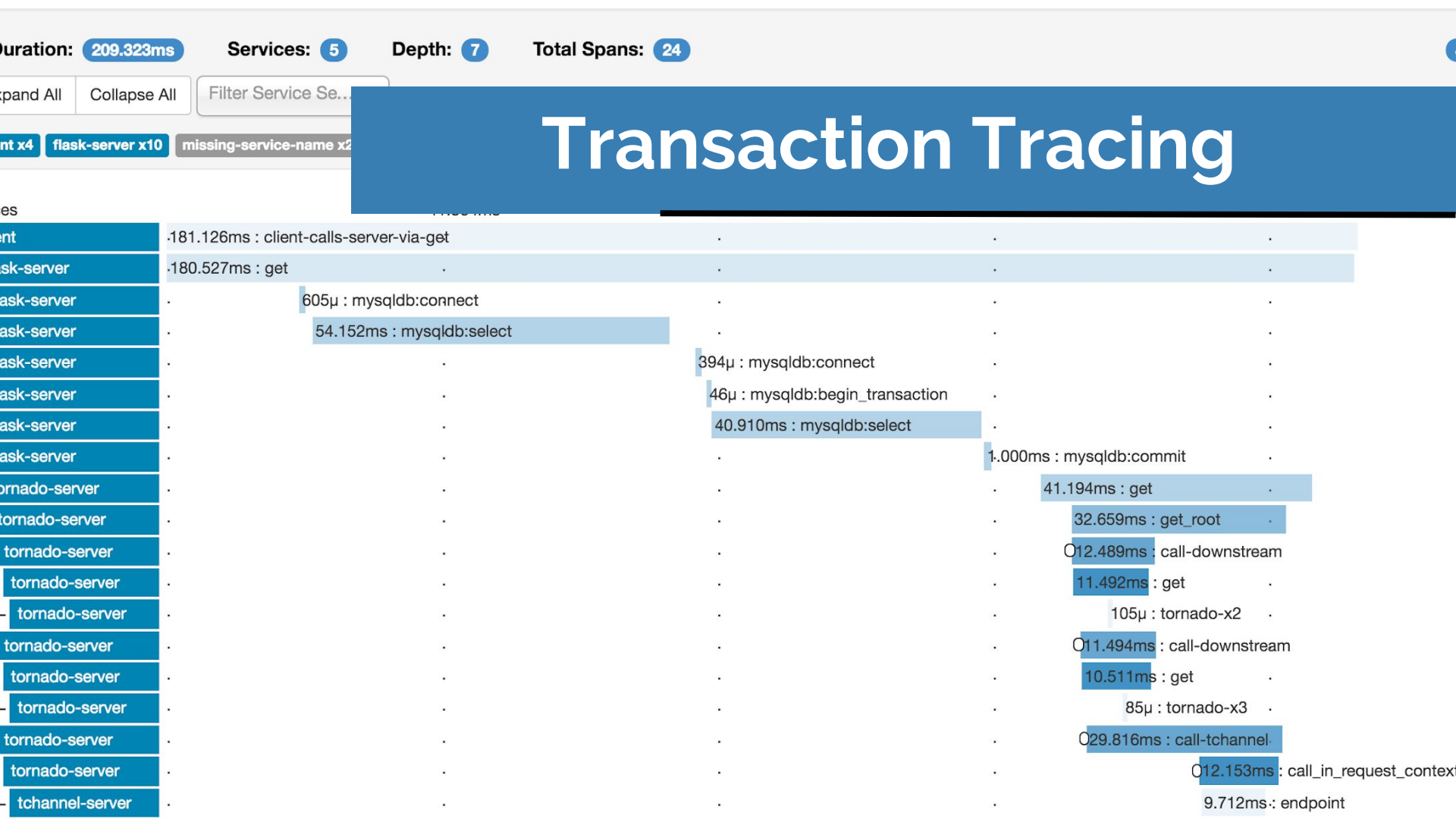
Ah, found you

```
Core: Exception handler (WEB): Uncaught TYPO3 Exception: #1: PHP
Warning: file_get_contents(http://third.party.service/api/bork.php?
x=1,y=2)
```

Problem Badly solved

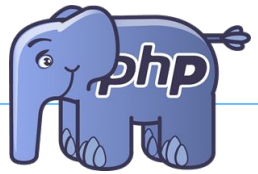
Dropped `default_socket_timeout` to 5s

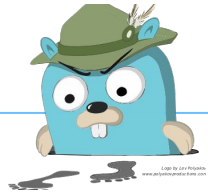
- Took an hour to find
- No easy way of separating out a multitude of issues
- Issues hiding behind issues


Duration: 209.323ms
Services: 5
Depth: 7
Total Spans: 24
Expand All
Collapse All
Filter Service Se...
nt x4
flask-server x10
missing-service-name x2
Transaction Tracing
181.126ms : client-calls-server-via-get
180.527ms : get
605µ : mysqldb:connect
54.152ms : mysqldb:select
394µ : mysqldb:connect
46µ : mysqldb:begin_transaction
40.910ms : mysqldb:select
1.000ms : mysqldb:commit
41.194ms : get
32.659ms : get_root
12.489ms : call-downstream
11.492ms : get
105µ : tornado-x2
11.494ms : call-downstream
10.511ms : get
85µ : tornado-x3
29.816ms : call-tchannel
12.153ms : call_in_request_context
9.712ms : endpoint

Finding Long Requests

molton	jaeger	Cassandra
A PHP extension that modifies automatically instruments and propagates tracing context:	Collects and reviews transaction traces	Stores transaction traces





Lessons

- New relic does this
- You don’t need it until you really do
- It becomes more relevant for distributed systems
- If you’re making an API call, that’s a “distributed” system

We will do this soon!

In Summary

Instrumentation is good!

Time Series Data · Structured & Aggregated Logging · Transaction Traces

I didn’t get what you just said

Brian Brazil, and Fabxc are very nice maintainers. There’s also a mailing list. Lastly, there’s a tonne of stuff on youtube.

What questions do you have?

Find all information at:

https://git.io/TODO

Give feedback at:

https://goo.gl/forms/NOPE