Monitoring InfluxCloud with InfluxDB, Grafana, Telegraf, and Kapacitor

Paul Dix
CTO & cofounder of InfluxData
@pauldix


MISSION ACCOMPLISHED

Who am I?

What is InfluxCloud?

Cost to monitor with SaaS > $10,000/month

We do it for < $700/ month

Raise your hand if you use InfluxDB

Raise your hand if you use Telegraf

Raise your hand if you use Kapacitor

Two kinds of time series data…

InfluxDB Intro (optional)

What is time series data?

Stock trades and quotes

129.62 -2.92 (-2.20%)
After Hours: 129.75 +0.13 (0.10%)
May 26, 7:59PM EDT
NASDAQ real-time data - Disclaimer
Currency in USD

Range	129.12 - 132.91	Div/yield	0.52/1.60
52 week	86.64 - 134.54	EPS	8.09
Open	132.60	Shares	5.76B
Vol / Avg.	70.70M/49.90M	Beta	0.84
Mkt cap	757.60B	Inst. own	61%
P/E	16.03		

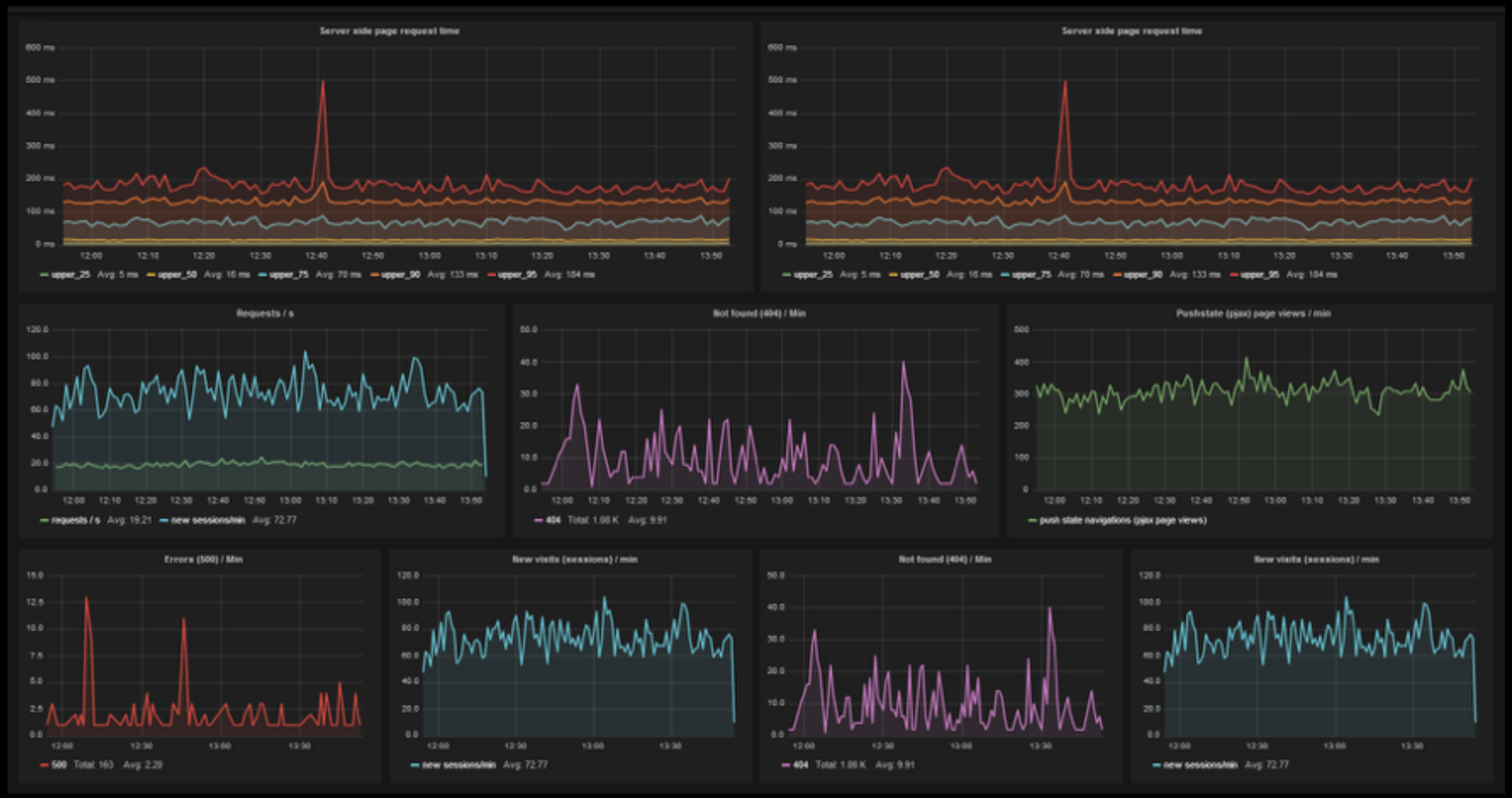
g+1 9k

Compare: Enter ticker here Add ☐ Dow Jones ☐ Nasdaq ☐ SNDK ☐ MSFT ☐ SSNNF ☐ HPQ ☐ IBM ☐ HTCKF ☐ ZAGG

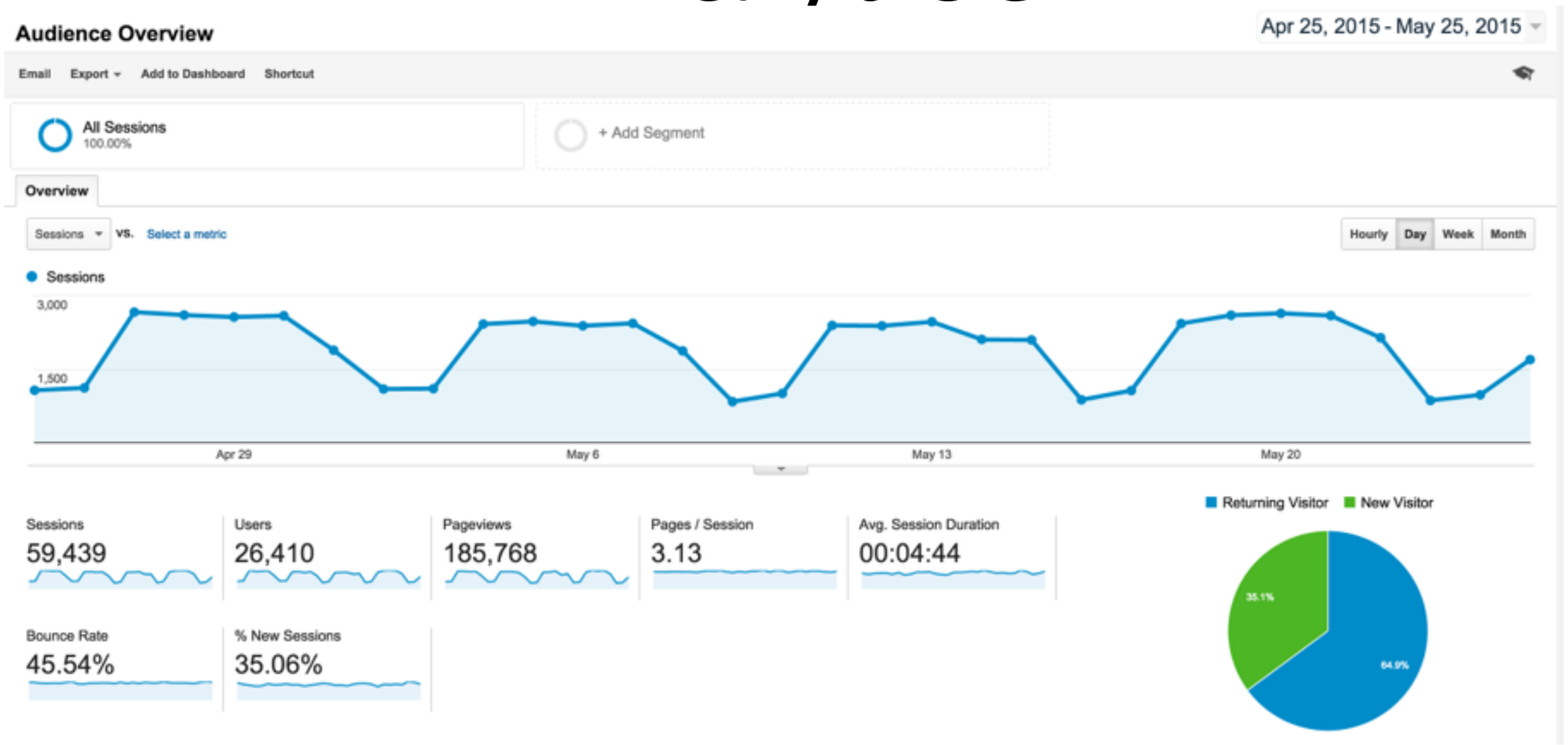
Zoom: 1d 5d 1m 3m 6m YTD 1y 5y 10y All

May 22, 2015 12:12 Price: 132.45 Vol: 107.75k

Metrics


Server side page request time
600 ms
500 ms
400 ms
300 ms
200 ms
100 ms
0 ms
12:00 12:10 12:20 12:30 12:40 12:50 13:00 13:10 13:20 13:30 13:40 13:50
upper_25 Avg: 5 ms upper_50 Avg: 16 ms upper_75 Avg: 70 ms upper_90 Avg: 133 ms upper_95 Avg: 184 ms
Server side page request time
600 ms
500 ms
400 ms
300 ms
200 ms
100 ms
0 ms
12:00 12:10 12:20 12:30 12:40 12:50 13:00 13:10 13:20 13:30 13:40 13:50
upper_25 Avg: 5 ms upper_50 Avg: 16 ms upper_75 Avg: 70 ms upper_90 Avg: 133 ms upper_95 Avg: 184 ms
Requests / s
120.0
100.0
80.0
60.0
40.0
20.0
0.0
12:00 12:10 12:20 12:30 12:40 12:50 13:00 13:10 13:20 13:30 13:40 13:50
requests / s Avg: 19.21 new sessions/min Avg: 72.77
Not found (404) / Min
50.0
40.0
30.0
20.0
10.0
0.0
12:00 12:10 12:20 12:30 12:40 12:50 13:00 13:10 13:20 13:30 13:40 13:50
404 Total: 1.08 K Avg: 9.91
Pushstate (pjax) page views / min
500
400
300
200
100
0
12:00 12:10 12:20 12:30 12:40 12:50 13:00 13:10 13:20 13:30 13:40 13:50
push state navigations (pjax page views)
Errors (500) / Min
15.0
12.5
10.0
7.5
5.0
2.5
0.0
12:00 12:30 13:00 13:30
500 Total: 163 Avg: 2.29
New visits (sessions) / min
120.0
100.0
80.0
60.0
40.0
20.0
0.0
12:00 12:30 13:00 13:30
new sessions/min Avg: 72.77
Not found (404) / Min
50.0
40.0
30.0
20.0
10.0
0.0
12:00 12:30 13:00 13:30
404 Total: 1.08 K Avg: 9.91
New visits (sessions) / min
120.0
100.0
80.0
60.0
40.0
20.0
0.0
12:00 12:30 13:00 13:30
new sessions/min Avg: 72.77

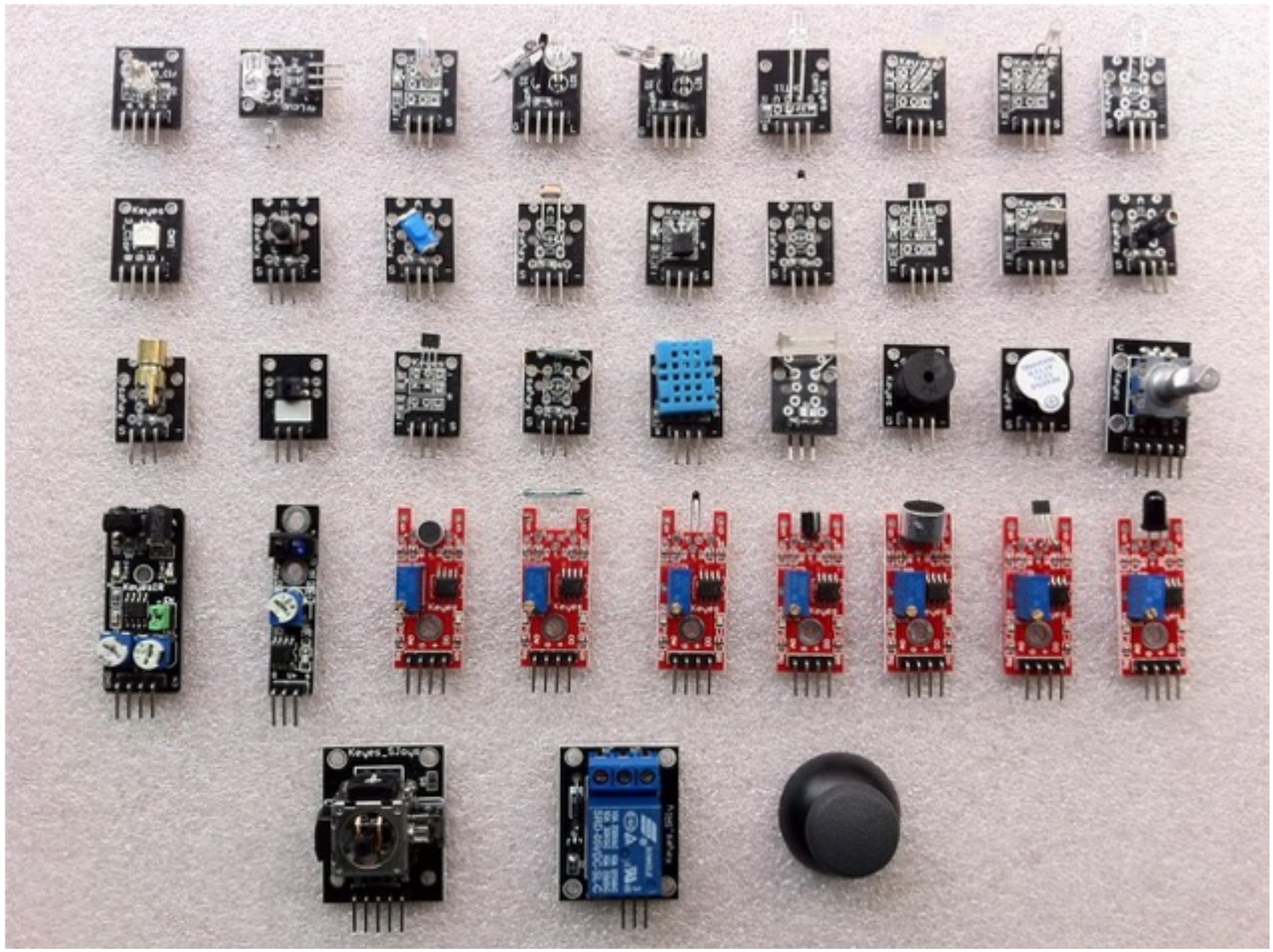
Analytics


Audience Overview
Apr 25, 2015 - May 25, 2015
Email
Export
Add to Dashboard
Shortcut
All Sessions
100.00%
+ Add Segment
Overview
Sessions
vs.
Select a metric
Hourly
Day
Week
Month
Sessions
3,000
1,500
Apr 29
May 6
May 13
May 20
Sessions
59,439
Users
26,410
Pageviews
185,768
Pages / Session
3.13
Avg. Session Duration
00:04:44
Bounce Rate
45.54%
% New Sessions
35.06%
Returning Visitor
New Visitor
35.1%
64.9%

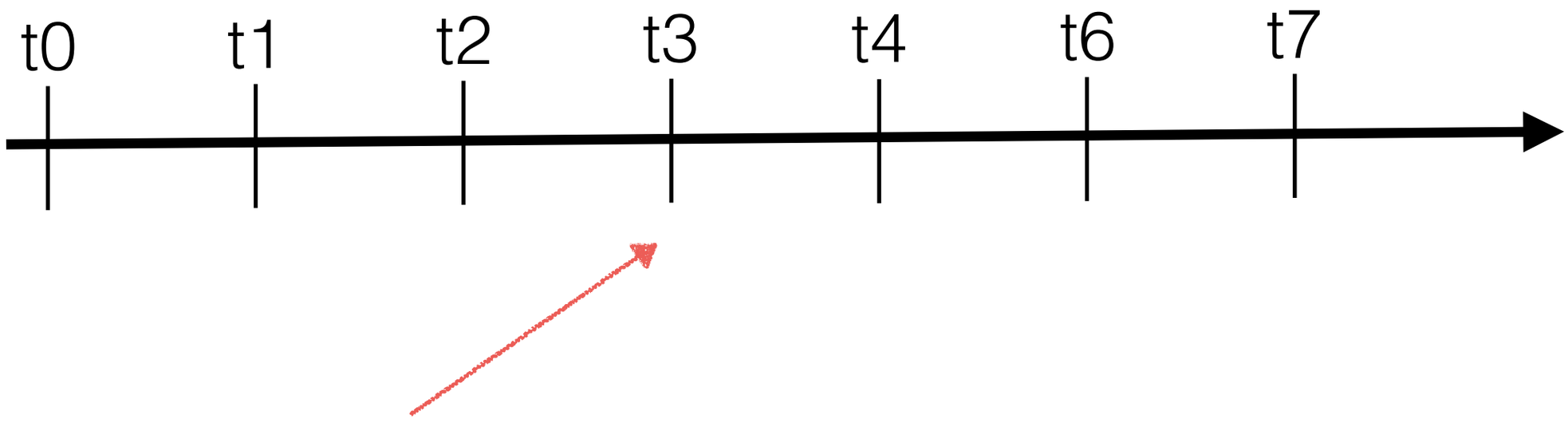
Events

```
64.242.88.10 - - [07/Mar/2004:16:05:49 -0800] "GET /twiki/bin/edit/Main/Double_bounce_sender?topicparent=Main.ConfigurationVariables HTTP/1.1" 401 12846
64.242.88.10 - - [07/Mar/2004:16:06:51 -0800] "GET /twiki/bin/rdiff/TWiki/NewUserTemplate?rev1=1.3&rev2=1.2 HTTP/1.1" 200 4523
64.242.88.10 - - [07/Mar/2004:16:10:02 -0800] "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
64.242.88.10 - - [07/Mar/2004:16:11:58 -0800] "GET /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 200 7352
64.242.88.10 - - [07/Mar/2004:16:20:55 -0800] "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200 5253
64.242.88.10 - - [07/Mar/2004:16:23:12 -0800] "GET /twiki/bin/oops/TWiki/AppendixFileSystem?template=oopsmore¶m1=1.12¶m2=1.12 HTTP/1.1" 200 11382
64.242.88.10 - - [07/Mar/2004:16:24:16 -0800] "GET /twiki/bin/view/Main/PeterThoeny HTTP/1.1" 200 4924
64.242.88.10 - - [07/Mar/2004:16:29:16 -0800] "GET /twiki/bin/edit/Main/Header_checks?topicparent=Main.ConfigurationVariables HTTP/1.1" 401 12851
64.242.88.10 - - [07/Mar/2004:16:30:29 -0800] "GET /twiki/bin/attach/Main/OfficeLocations HTTP/1.1" 401 12851
64.242.88.10 - - [07/Mar/2004:16:31:48 -0800] "GET /twiki/bin/view/TWiki/WebTopicEditTemplate HTTP/1.1" 200 3732
64.242.88.10 - - [07/Mar/2004:16:32:50 -0800] "GET /twiki/bin/view/Main/WebChanges HTTP/1.1" 200 40520
64.242.88.10 - - [07/Mar/2004:16:33:53 -0800] "GET /twiki/bin/edit/Main/Smtpd_etrn_restrictions?topicparent=Main.ConfigurationVariables HTTP/1.1" 401 12851
64.242.88.10 - - [07/Mar/2004:16:35:19 -0800] "GET /mailman/listinfo/business HTTP/1.1" 200 6379
64.242.88.10 - - [07/Mar/2004:16:36:22 -0800] "GET /twiki/bin/rdiff/Main/WebIndex?rev1=1.2&rev2=1.1 HTTP/1.1" 200 46373
64.242.88.10 - - [07/Mar/2004:16:37:27 -0800] "GET /twiki/bin/view/TWiki/DontNotify HTTP/1.1" 200 4140
64.242.88.10 - - [07/Mar/2004:16:39:24 -0800] "GET /twiki/bin/view/Main/TokyoOffice HTTP/1.1" 200 3853
64.242.88.10 - - [07/Mar/2004:16:43:54 -0800] "GET /twiki/bin/view/Main/MikeMannix HTTP/1.1" 200 3686
64.242.88.10 - - [07/Mar/2004:16:45:56 -0800] "GET /twiki/bin/attach/Main/PostfixCommands HTTP/1.1" 401 12846
64.242.88.10 - - [07/Mar/2004:16:47:12 -0800] "GET /robots.txt HTTP/1.1" 200 68
64.242.88.10 - - [07/Mar/2004:16:47:46 -0800] "GET /twiki/bin/rdiff/Know/ReadmeFirst?rev1=1.5&rev2=1.4 HTTP/1.1" 200 5724
64.242.88.10 - - [07/Mar/2004:16:49:04 -0800] "GET /twiki/bin/view/Main/TWikiGroups?rev=1.2 HTTP/1.1" 200 5162
64.242.88.10 - - [07/Mar/2004:16:50:54 -0800] "GET /twiki/bin/rdiff/Main/ConfigurationVariables HTTP/1.1" 200 59679
64.242.88.10 - - [07/Mar/2004:16:52:35 -0800] "GET /twiki/bin/edit/Main/Flush_service_name?topicparent=Main.ConfigurationVariables HTTP/1.1" 401 12851
64.242.88.10 - - [07/Mar/2004:16:53:46 -0800] "GET /twiki/bin/rdiff/TWiki/TWikiRegistration HTTP/1.1" 200 34395
64.242.88.10 - - [07/Mar/2004:16:54:55 -0800] "GET /twiki/bin/rdiff/Main/NicholasLee HTTP/1.1" 200 7235
64.242.88.10 - - [07/Mar/2004:16:56:39 -0800] "GET /twiki/bin/view/Sandbox/WebHome?rev=1.6 HTTP/1.1" 200 8545
64.242.88.10 - - [07/Mar/2004:16:58:54 -0800] "GET /mailman/listinfo/administration HTTP/1.1" 200 6459
lordgun.org - - [07/Mar/2004:17:01:53 -0800] "GET /razor.html HTTP/1.1" 200 2869
64.242.88.10 - - [07/Mar/2004:17:09:01 -0800] "GET /twiki/bin/search/Main/SearchResult?scope=text®ex=on&search=Joris%20*Benschop[^A-Za-z] HTTP/1.1" 200 4284
64.242.88.10 - - [07/Mar/2004:17:10:20 -0800] "GET /twiki/bin/oops/TWiki/TextFormattingRules?template=oopsmore¶m1=1.37¶m2=1.37 HTTP/1.1" 200 11400
64.242.88.10 - - [07/Mar/2004:17:13:50 -0800] "GET /twiki/bin/edit/TWiki/DefaultPlugin?t=1078688936 HTTP/1.1" 401 12846
64.242.88.10 - - [07/Mar/2004:17:16:00 -0800] "GET /twiki/bin/search/Main/?scope=topic®ex=on&search=^g HTTP/1.1" 200 3675
64.242.88.10 - - [07/Mar/2004:17:17:27 -0800] "GET /twiki/bin/search/TWiki/?scope=topic®ex=on&search=^d HTTP/1.1" 200 5773
lj1036.inktomisearch.com - - [07/Mar/2004:17:18:36 -0800] "GET /robots.txt HTTP/1.0" 200 68
lj1090.inktomisearch.com - - [07/Mar/2004:17:18:41 -0800] "GET /twiki/bin/view/Main/LondonOffice HTTP/1.0" 200 3860
```

Sensor data

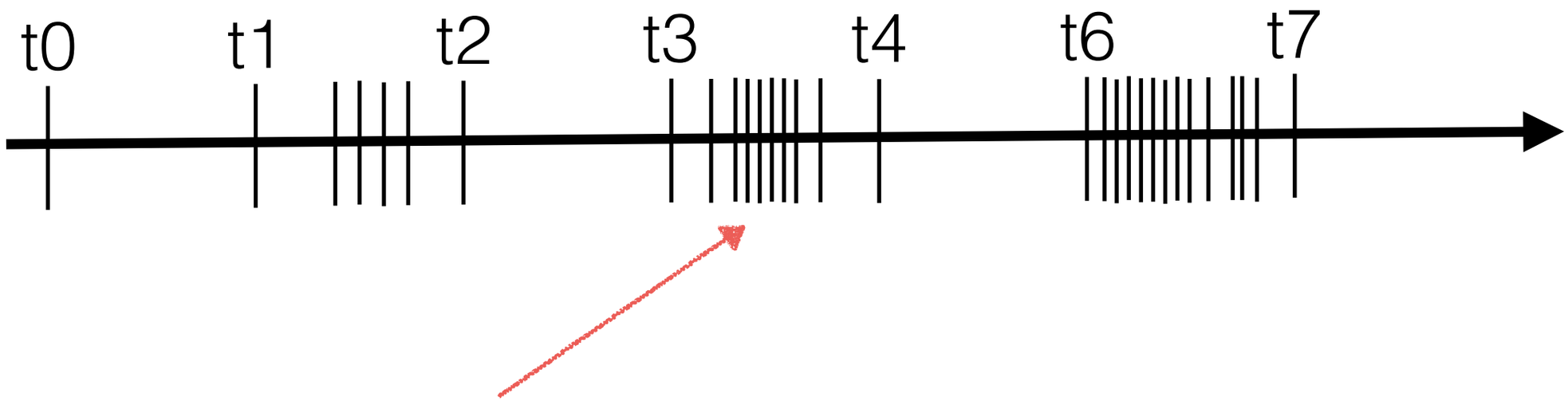


Regular time series



Samples at regular intervals

Irregular time series



Events whenever they come in

API - 2 endpoints

API - 2 endpoints

Write: HTTP POST /write?db=mydb&rp=foo

API - 2 endpoints

Write: HTTP POST /write?db=mydb&rp=foo

Read: HTTP GET /query?db=mydb&rp=foo

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=23422.0 1422568543702900257
```

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=23422.0 1422568543702900257
```

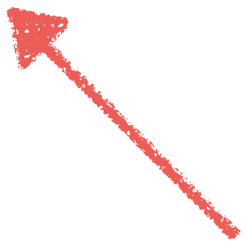
Line Protocol

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=23422.0 1422568543702900257
```



Measurement name

Writing Data

/write?db=mydb&rp=foo

BODY:

cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,**direction=in,host=server01,region=us-west** value=23422.0 1422568543702900257

Tags (should be sorted by key, value)

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=23422.0 1422568543702900257
```

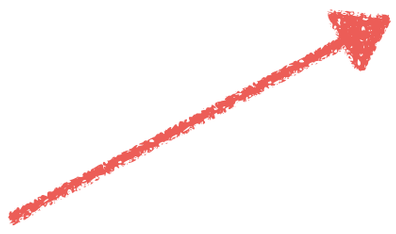
Tags (values are always strings)

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=23422.0 1422568543702900257
```



Fields

Writing Data

/write?db=mydb&rp=foo

BODY:

cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
some_measurement **value=23422.0,context="asdf jkl"** 1422568543702900257

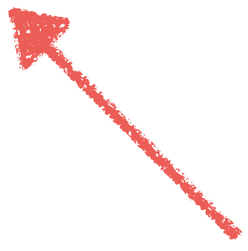
Unlimited Fields

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
some_measurement value=23422 1422568543702900257
```



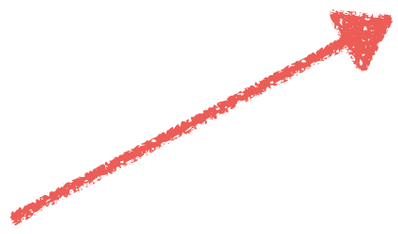
Fields (types: int64, float64, string, bool)

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=0i 1422568543702900257
```



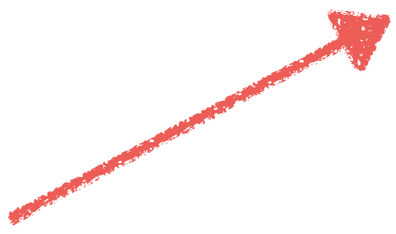
Fields (ints must have trailing i)

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=0.0 1422568543702900257
```



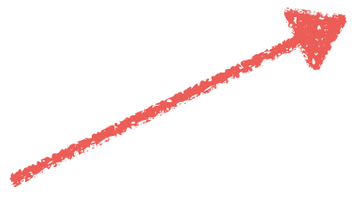
Fields (types must remain consistent)

Writing Data

/write?db=mydb&rp=foo

BODY:

```
cpu_load,host=server01,region=us-west value=0.64
cpu_load,host=server02,region=us-west value=0.55 1422568543702900257
network,direction=in,host=server01,region=us-west value=0.0 1422568543702900257
```



Timestamp (nanosecond epoch)

Our setup

Telegraf on all hosts

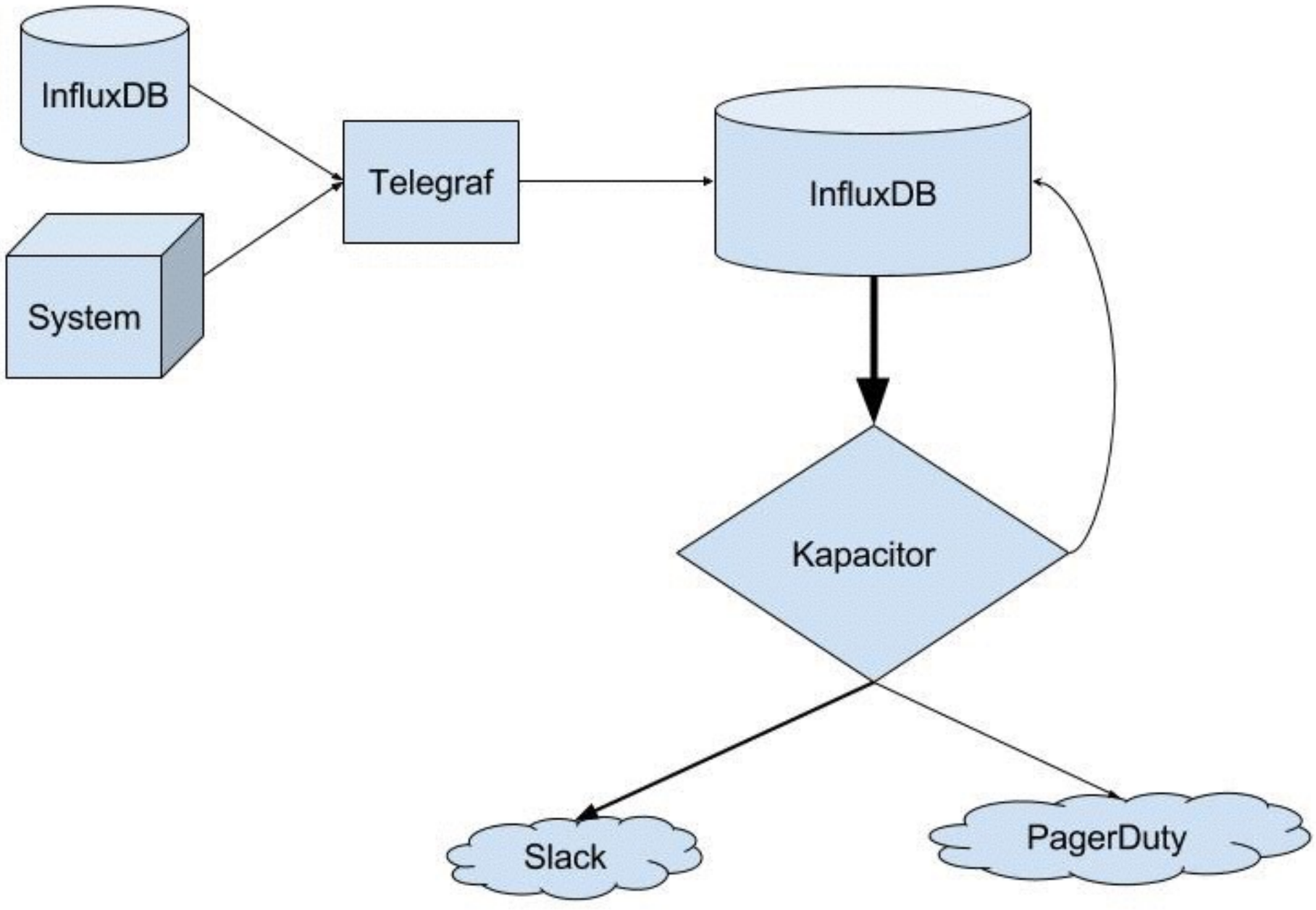
system metrics, InfluxDB metrics

InfluxDB

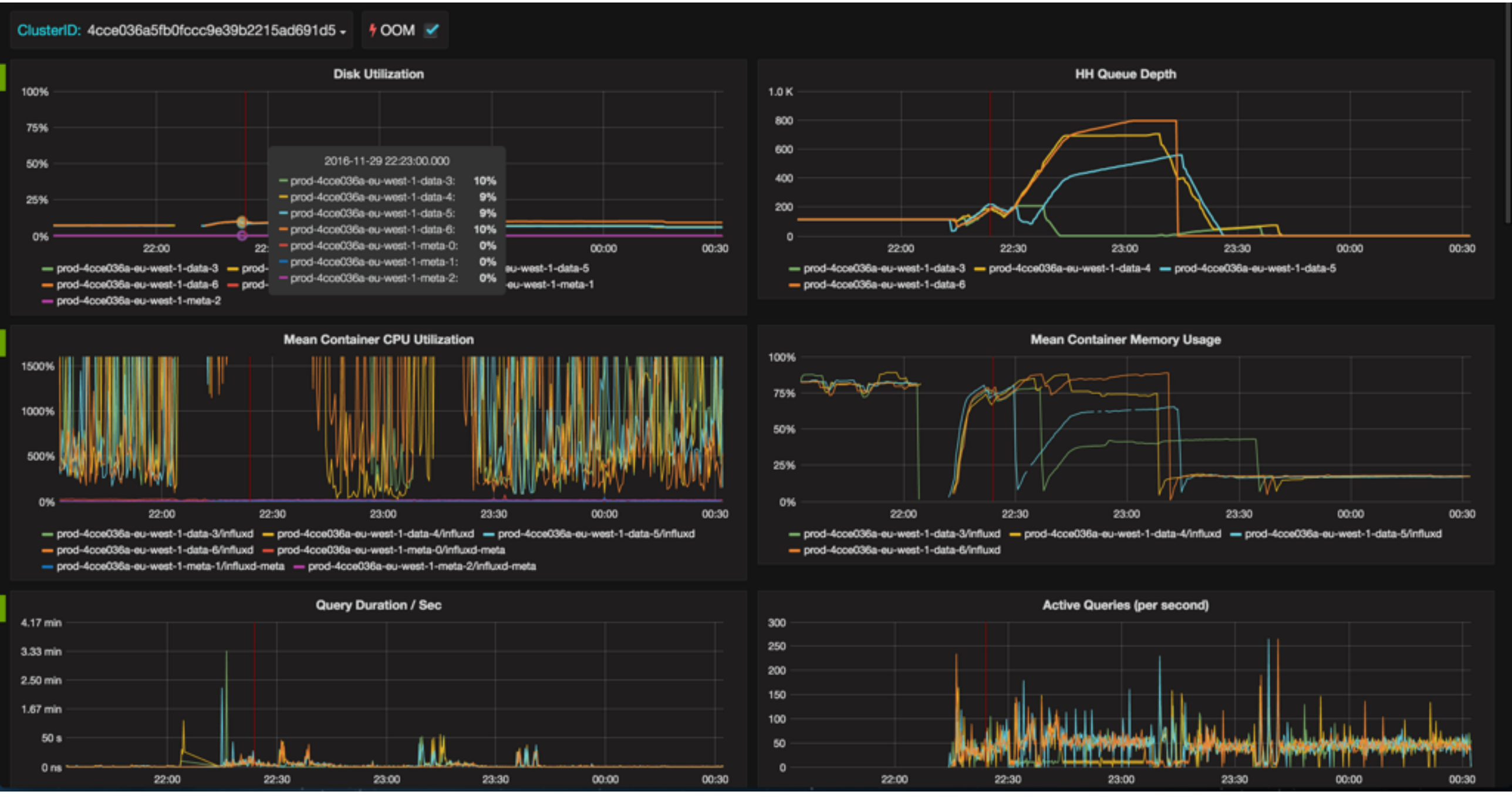
monitoring instance
m4.4xlarge

Kapacitor

monitoring instance
same instance as InfluxDB


InfluxDB
System
Telegraf
InfluxDB
Kapacitor
Slack
PagerDuty

Grafana


ClusterID: 4cce036a5fb0fccc9e39b2215ad691d5
OOM
Disk Utilization
100%
75%
50%
25%
0%
22:00
00:00
00:30
2016-11-29 22:23:00.000
prod-4cce036a-eu-west-1-data-3: 10%
prod-4cce036a-eu-west-1-data-4: 9%
prod-4cce036a-eu-west-1-data-5: 9%
prod-4cce036a-eu-west-1-data-6: 10%
prod-4cce036a-eu-west-1-meta-0: 0%
prod-4cce036a-eu-west-1-meta-1: 0%
prod-4cce036a-eu-west-1-meta-2: 0%
prod-4cce036a-eu-west-1-data-3
prod-4cce036a-eu-west-1-data-6
prod-4cce036a-eu-west-1-meta-2
HH Queue Depth
1.0 K
800
600
400
200
0
22:00
22:30
23:00
23:30
00:00
00:30
prod-4cce036a-eu-west-1-data-3
prod-4cce036a-eu-west-1-data-4
prod-4cce036a-eu-west-1-data-5
prod-4cce036a-eu-west-1-data-6
Mean Container CPU Utilization
1500%
1000%
500%
0%
22:00
22:30
23:00
23:30
00:00
00:30
prod-4cce036a-eu-west-1-data-3/influxd
prod-4cce036a-eu-west-1-data-4/influxd
prod-4cce036a-eu-west-1-data-5/influxd
prod-4cce036a-eu-west-1-data-6/influxd
prod-4cce036a-eu-west-1-meta-0/influxd-meta
prod-4cce036a-eu-west-1-meta-1/influxd-meta
prod-4cce036a-eu-west-1-meta-2/influxd-meta
Mean Container Memory Usage
100%
75%
50%
25%
0%
22:00
22:30
23:00
23:30
00:00
00:30
prod-4cce036a-eu-west-1-data-3/influxd
prod-4cce036a-eu-west-1-data-4/influxd
prod-4cce036a-eu-west-1-data-5/influxd
prod-4cce036a-eu-west-1-data-6/influxd
Query Duration / Sec
4.17 min
3.33 min
2.50 min
1.67 min
50 s
0 ns
22:00
22:30
23:00
23:30
00:00
00:30
Active Queries (per second)
300
250
200
150
100
50
0
22:00
22:30
23:00
23:30
00:00
00:30

Grafana = Visibility

Only part of the monitoring story

hundreds or thousands
of time series per server

~ 1M total series

You can’t visualize that!

Kapacitor

time series processing engine

Kapacitor

- Open Source
- MIT
- Go
- TICKscript
- Send alerts to log, web hook, email, exec, Sensu, HipChat, Alerta, Slack, OpsGenie, VictorOps, PagerDuty, Talk, Telegram

Stream Processing

Batch Processing

Some of our scripts

```
// detect system down
var period = 2m
var every = 30s

// select the stream
var sys_data = stream
    |from()
        .database('telegraf')
        .measurement('system')
        .where(lambda: "host" =~ /tot.*/ OR "host" =~ /prod.*/)
        .groupBy('host', 'cluster_id')
    |window()
        .period(period)
        .every(every)
```

```
msg = 'Node {{ index .Tags "host" }} {{ if ne .Level "OK" }}is
down!{{ else }}is back online.{{ end }}'

// alert if no data in period
sys_data
    |where(lambda: "host" != '')
|deadman(0.0, period)
    .id('deadman/{{ index .Tags "host" }}')
    .message(msg)
    .details('')
    .stateChangesOnly()
    .post('http://localhost:8080/health-check')
```

```
// Alert when disks are this % full
var alert_threshold = 80

// Use a larger period here, as the telegraf data can be a little late
// if the server is under load.
var period = 10m

// How often to query for the period.
var every = 1m

batch
    |query('''
  SELECT max(used_percent) FROM "telegraf"."default".disk
  WHERE ("host" =~ /prod-.*/ OR "host" =~ /tot-.*/) AND "path" = '/influxdb/conf'''')
    .period(period)
    .every(every)
    .groupBy('host')
|alert()
    .crit(lambda: "max" > alert_threshold)
    .message('Host {{ index .Tags "host" }}\'s disk is {{ index .Fields "max" }}% full!')
    .details('')
    .stateChangesOnly()
    .slack()
```

```
// Alert when this much % write failure occur over the specified period
var error_threshold = 80
var period = 2m
var every = 30s

var error_delta = stream
    |from()
        .database('telegraf')
        .retentionPolicy('default')
        .measurement('influxdb_httpd')
        .where(lambda: "host" =~ /tot.*/ OR "host" =~ /prod.*/)
    |groupBy('host', 'cluster_id')
    |window()
        .period(period)
        .every(every)
    |default()
        .field('serverError', 0)
    |derivative('serverError')
        .nonNegative()
        .as('derivative')
        .unit(period)
    |eval(lambda: "derivative")
        .as('derivative')
```

```
var ok_delta = stream
    |from()
        .database('telegraf')
        .retentionPolicy('default')
        .measurement('influxdb_httpd')
        .where(lambda: "host" =~ /tot.*/ OR "host" =~ /prod.*/)
    |groupBy('host', 'cluster_id')
    |window()
        .period(period)
        .every(every)
    |default()
        .field('req', 0)
    |derivative('req')
        .nonNegative()
        .as('derivative')
        .unit(period)
    |eval(lambda: "derivative")
        .as('derivative')
```

```
error_delta
    |join(ok_delta)
        .as('error', 'ok')
    |eval(lambda: (float("error.derivative") / float("error.derivative" + "ok.derivative")) * 100.0)
        .as('error_percent')
    |alert()
     .id(‘…’)
     .message(‘…’)
    .crit(lambda: "error_percent" >= error_threshold)
     .stateChangesOnly()
     .slack()
```

User Defined Functions

some Grafana dashboards…

Questions?

Paul Dix
@pauldix