

# Introduction into the ELK stack

Alexander Reelsen

[alexander.reelsen@elasticsearch.com](mailto:alexander.reelsen@elasticsearch.com)

[@spinscale](#)

# Agenda

- Introduction
- The ELK stack
- Samples, samples, samples
- Summary

# About Elasticsearch

- Founded 2012 in Amsterdam
- Funded by Benchmark, Index Ventures and NEA Ventures
- Distributed company  
Offices in Los Altos, Amsterdam, London, Berlin, Phoenix
- Offering support subscriptions & trainings
- We're hiring

# About me

- Joined early 2013
- Interested in all things scale, search & concurrency
- Elasticsearch developer, doing trainings, support, blog posts, conferences, presentations

# About me

- Joined early 2013
- Interested in all things scale, search & concurrency
- Elasticsearch developer, doing trainings, support, blog posts, conferences, presentations



**Lukas Grebe**

@LukasGrebe

 Follow

@spinscale die JSON chars sind größer als dein Kopf! #dchh

 Reply  Retweet  Favorite  More

# Introduction

# How do **you** decide?

- What is the core asset of your company?  
Ideas, patents, employees, customers, warehouse, software, ...
- Where to invest/develop next?
- Data driven decisions

# How do **you** decide?

- What is the core asset of your company?  
Ideas, patents, employees, customers, warehouse, software, ...
- Where to invest/develop next?
- Data driven decisions
  - logfiles for scaling up/down
  - warehouse withdrawal triggers orders
  - history for fraud detection
  - assembly line, throughput improvement

... data explosion

# More data is *Big Data*

- More and more data  
Recommendations, page views, IoT, social media
- Better decisions == more data?

**but ...**

# The *Big Data* promise

# The *Big Data* ~~promise~~ problem

# The *Big Data* ~~promise~~ problem



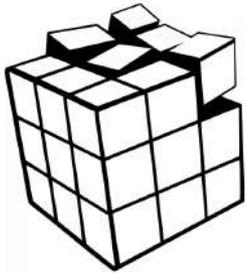
reaction time

Time between storing and analysing an event

# The *Big Data* ~~promise~~ problem



reaction time



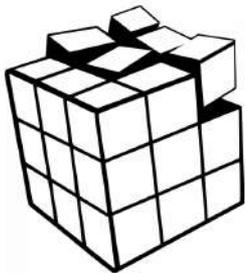
enrichment

Increase event value by enriching

# The *Big Data* ~~promise~~ problem



reaction time



enrichment



insights

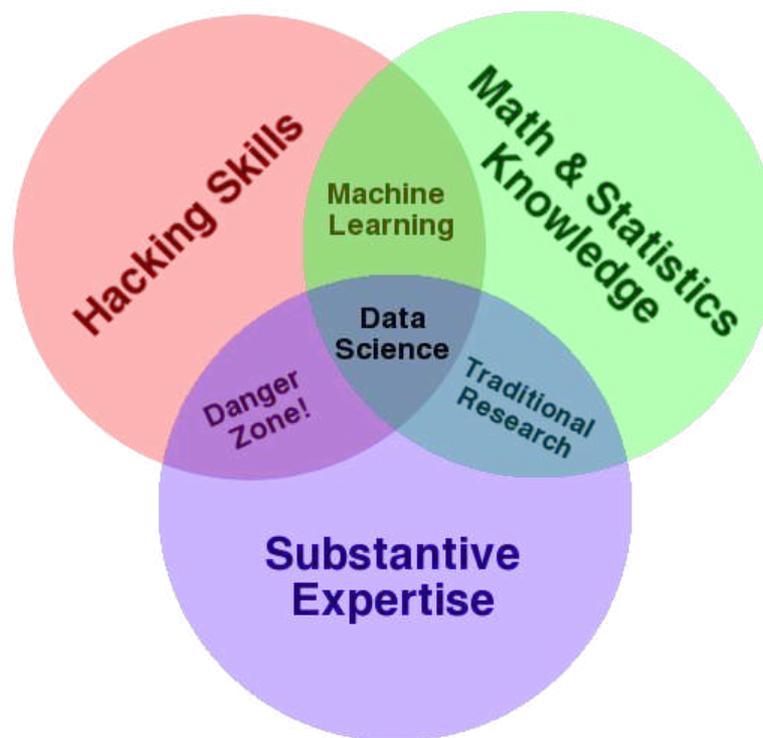
optimize for query, not for storage

# No problem, lets make up a new job title

- We failed so hard in this industry, that we created a new job to clean up this mess

# No problem, lets make up a new job title

- We failed so hard in this industry, that we created a new job to clean up this mess



Source: <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>

# Data scientist problem

- Result of a flawed infrastructure
- Result of a flawed process/company politics
- Often doing someone else job  
Enriching data, getting data, creating reports
- Data scientists are important, lets help them to do their real job, which is not ETL but providing information!

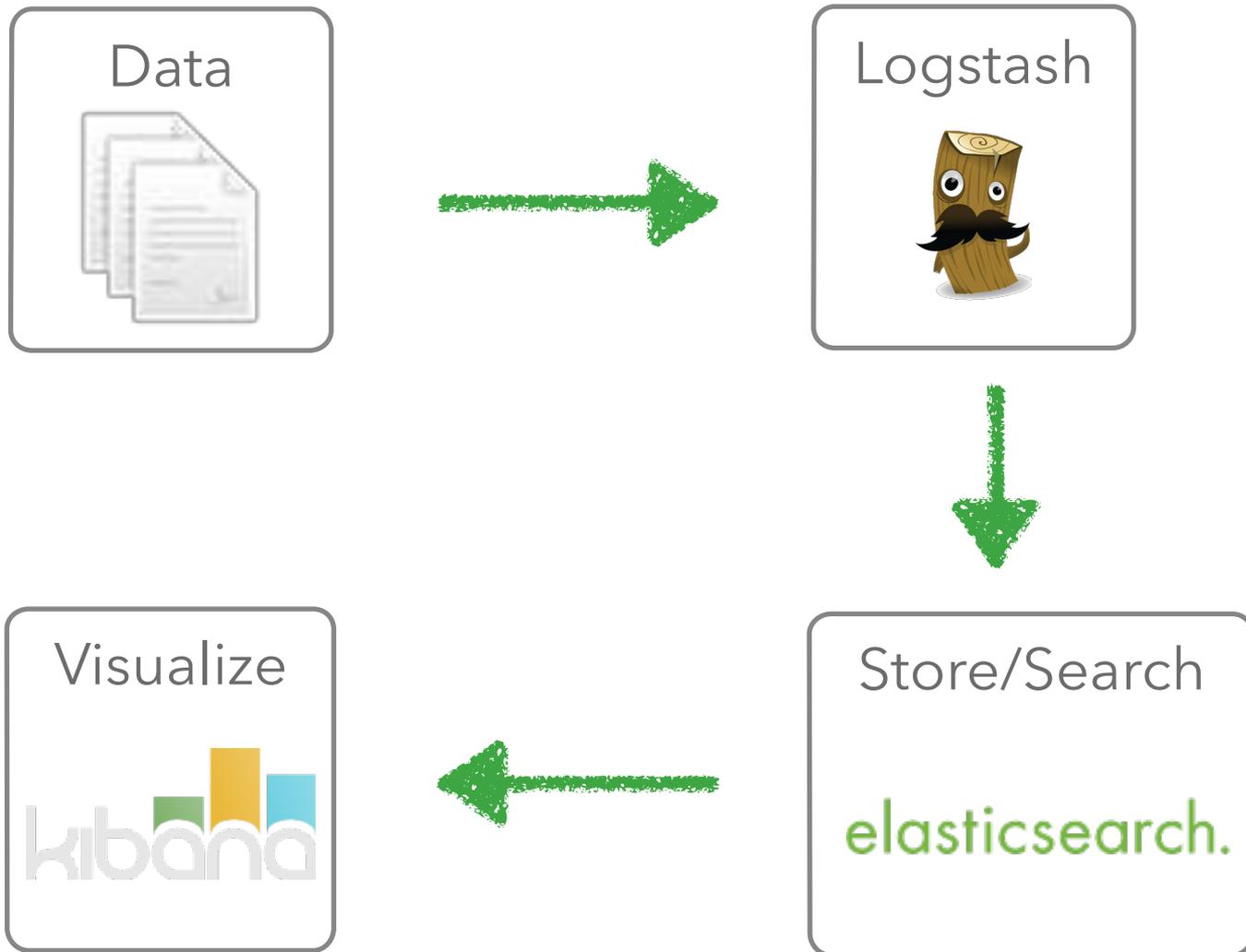
# Requirements

- Clean data to work on
- Fast analysis chain  
near real-time
- Easy to use user interface  
Everyone is able to create own reports

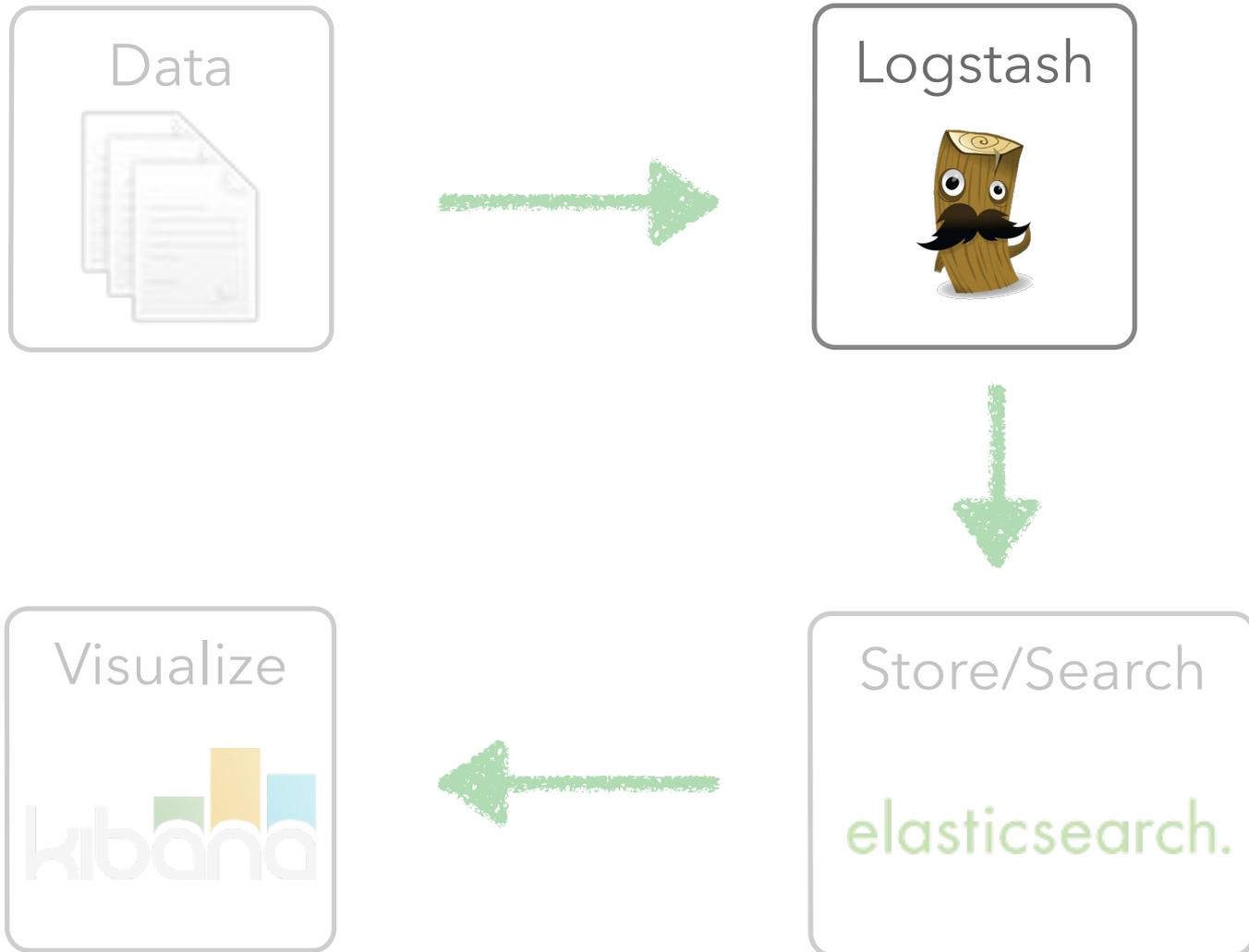
## Meet the ELK stack

# The ELK stack

# The ELK stack



# Logstash



# Logstash

- Managing events and logs
- Collect data
- Parse data
- Enrich data
- Store data
- Open Source: Apache License 2.0



# Logstash architecture

## Input

datastore  
stream  
log files  
files  
monitoring  
queues  
network



## Filter



parse, enrich, tag, drop

## Output

datastore  
files  
email  
pager  
monitoring  
chat  
API  
queues



# Logstash architecture

## Input

datastore  
stream  
log files  
files  
monitoring  
queues  
network



ip: 141.1.1.1

## Filter

Logstash



parse, enrich, tag, drop

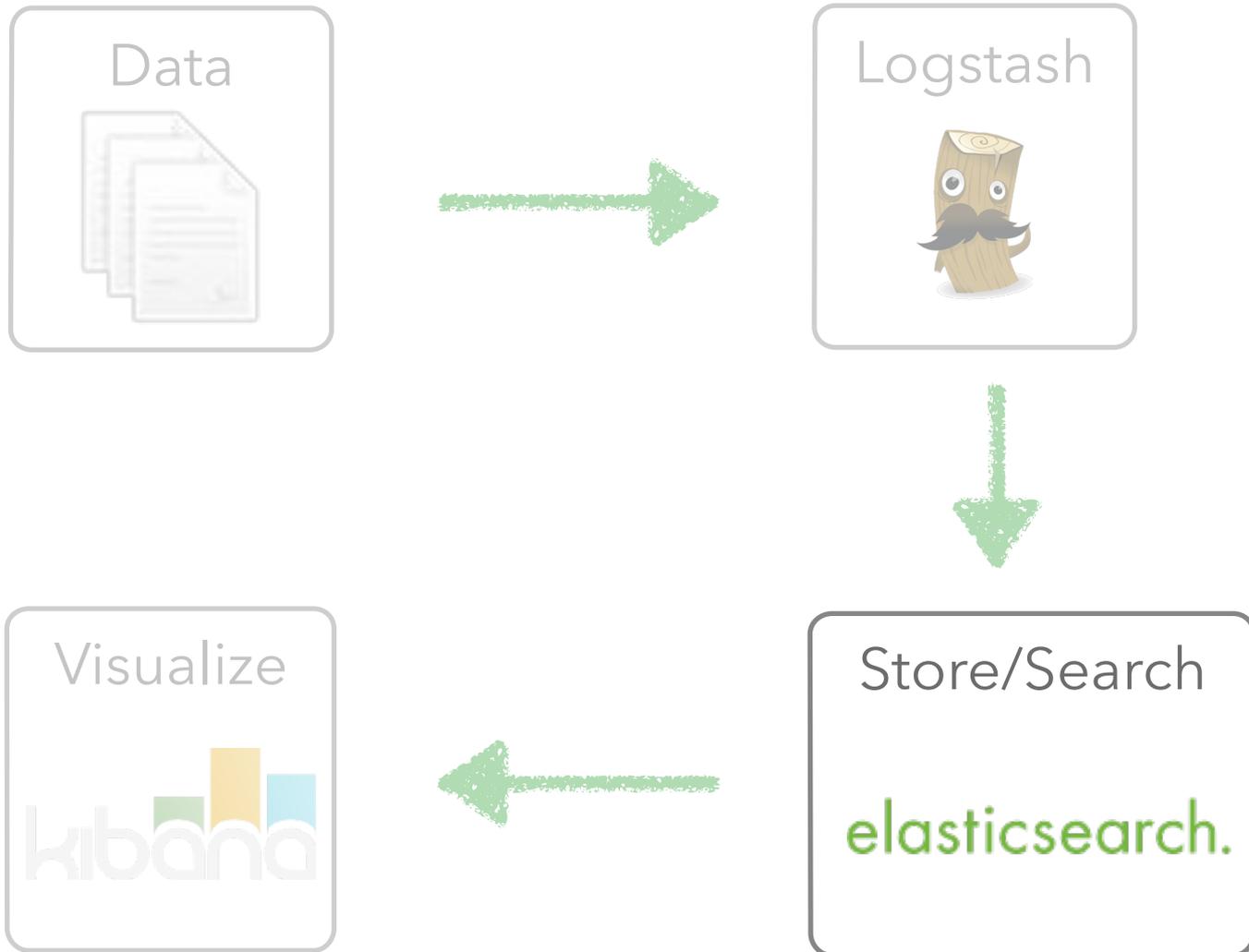
## Output

ip: 141.1.1.1  
city: Zurich  
country: CH



datastore  
files  
email  
pager  
monitoring  
chat  
API  
queues

# Elasticsearch



# Elasticsearch

- Schema-free, REST & JSON based distributed search engine
- Open Source: Apache License 2.0
- Easy to understand, yet very powerful query language
  - Full text search (phrase, fuzzy)
  - Numeric search (support ranges, dates, ipv4 addresses)
  - Highlighting
  - Aggregations
  - Suggestions

# Wenn Suchboxen nicht funktionieren

Wie am besten die Qualitaet der eigenen Suchapplikation sicherstellen?

Isabel Drost-Fromm

Freitag, 15:00 Uhr, Kinosaal 8

# Kibana

- Execute queries on your data & visualize results
- Add/remove widgets
- Share/Save/Load dashboards
- Open Source: Apache License 2.0

# Kibana



Samples, samples, samples

# Samples

- Guardian case study
- Web server logs
- meetup.com RSVP stream
- Wikipedia update stream
- sysdig output

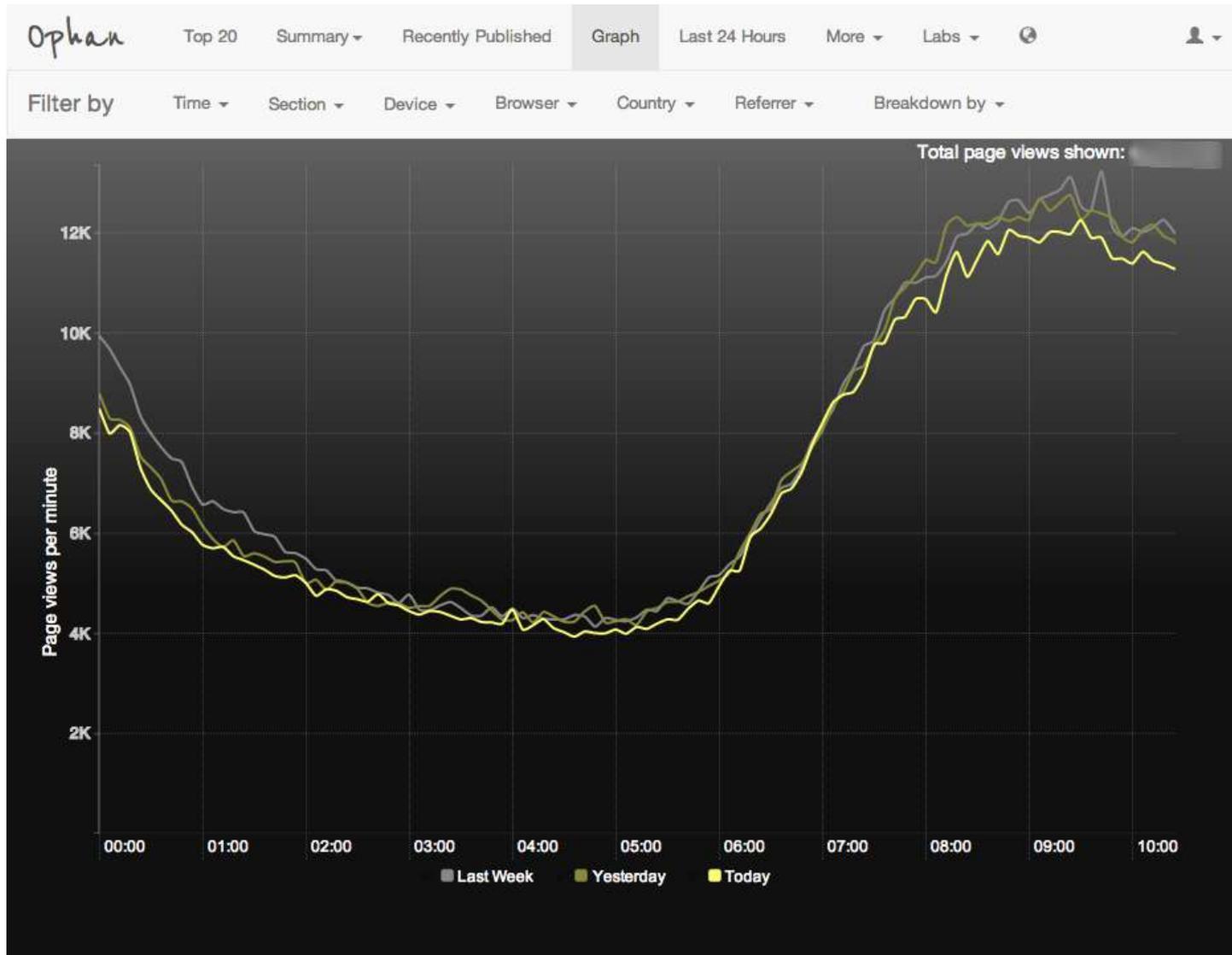
# Case Study: The Guardian

**theguardian**

- Ophan: In-house analytics software
- Empower the organization
  - Give the entire organization real-time insight into audience engagement
  - Democratize analytics access for more than 500 users
  - Encourage a culture of exploration and innovation for all employees
- Leverage real-time analytics
  - Easily query 360 million documents
  - See traffic for all content as it happens

**elasticsearch.**

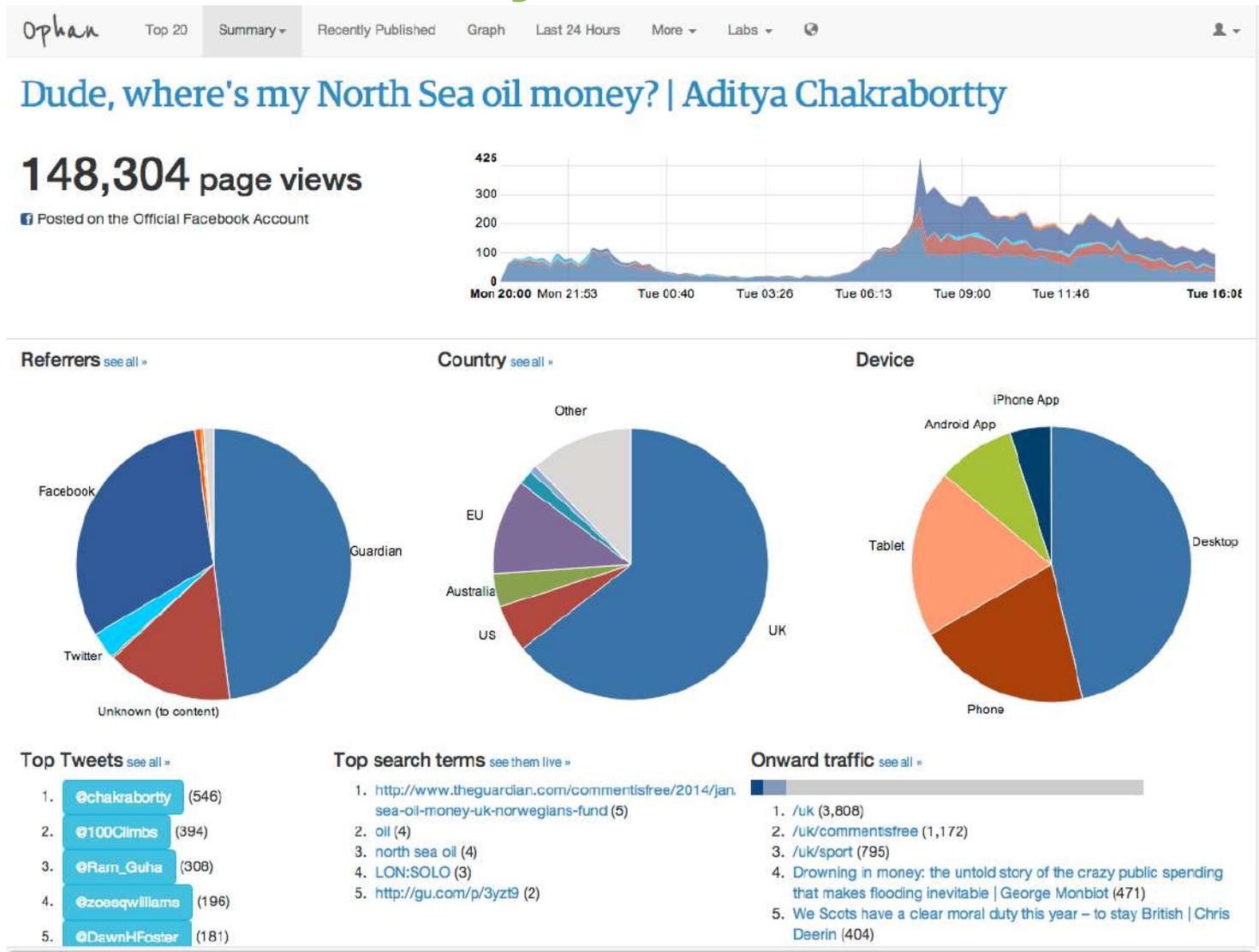
# Case Study: The Guardian



# Case Study: The Guardian



# Case Study: The Guardian



# Case Study: The Guardian



# Example: Web server log files



# Example: Web server log files

```
input { stdin {} }

filter {
  grok { match => { "message" => "%{COMBINEDAPACHELOG}" } }

  date { match => [ "timestamp", "dd/MMM/YYYY:HH:mm:ss Z" ] }

  geoip { source => "clientip" }

  useragent {
    source => "agent"
    target => "useragent"
  }
}

output {
  elasticsearch {
    protocol => "http"
    host => "localhost"
  }
}
```

# Example: Web server log files

```
input { stdin {} }

filter {
  grok { match => { "message" => "%{COMBINEDAPACHELOG}" } }

  date { match => [ "timestamp", "dd/MMM/YYYY:HH:mm:ss Z" ] }

  geoip { source => "clientip" }
}
```

```
cat access.log | logstash agent -f logstash-logs.conf
```

```
  source => "agent"
  target => "useragent"
}

output {
  elasticsearch {
    protocol => "http"
    host => "localhost"
  }
}
```

# Example: Web server log files

```
{
  "message" => "83.149.9.216 - - [28/May/2014:16:13:42 -0500] \"GET /presentations/logstash-monitorama-2013/images/kibana-search.png HTTP/1.1\" 200 203023
  \"http://semicomplete.com/presentations/logstash-monitorama-2013/\" \"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/32.0.1700.77 Safari/537.36\"",
  "@version" => "1",
  "@timestamp" => "2014-05-28T21:13:42.000Z",
  "host" => "kryptic.local",
  "clientip" => "83.149.9.216",
  "ident" => "-",
  "auth" => "-",
  "timestamp" => "28/May/2014:16:13:42 -0500",
  "verb" => "GET",
  "request" => "/presentations/logstash-monitorama-2013/images/kibana-search.png",
  "httpversion" => "1.1",
  "response" => "200",
  "bytes" => "203023",
  "referrer" => "\"http://semicomplete.com/presentations/logstash-monitorama-2013/\"",
  "agent" => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/537.36\"",
  "geopip" => {
    "ip" => "83.149.9.216",
    "country_code2" => "RU",
    "country_code3" => "RUS",
    "country_name" => "Russian Federation",
    "continent_code" => "EU",
    "region_name" => "48",
    "city_name" => "Moscow",
    "latitude" => 55.75219999999999,
    "longitude" => 37.6156,
    "timezone" => "Europe/Moscow",
    "real_region_name" => "Moscow City",
    "location" => [
      [0] 37.6156,
      [1] 55.75219999999999
    ]
  },
  "useragent" => {
    "name" => "Chrome",
    "os" => "Mac OS X 10.9.1",
    "os_name" => "Mac OS X",
    "os_major" => "10",
    "os_minor" => "9",
    "device" => "Other",
    "major" => "32",
    "minor" => "0",
    "patch" => "1700"
  }
}
```

# Example: Web server log files

```
"message" => "83.149.9.216 - - [28/May/2014:16:13:42 -0500] \"GET /
presentations/logstash-monitorama-2013/images/kibana-search.png HTTP/1.1\"
200 203023 \"http://semicomplete.com/presentations/logstash-
monitorama-2013/\" \"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/
537.36\"",
```

**grok**

```
"@version" => "1",
"@timestamp" => "2014-05-28T21:13:42.000Z",
  "host" => "kryptic.local",
  "clientip" => "83.149.9.216",
  "ident" => "-",
  "auth" => "-",
  "timestamp" => "28/May/2014:16:13:42 -0500",
  "verb" => "GET",
  "request" => "/presentations/logstash-monitorama-2013/images/
kibana-search.png",
  "httpversion" => "1.1",
  "response" => "200",
  "bytes" => "203023",
  "referrer" => "\"http://semicomplete.com/presentations/logstash-
monitorama-2013/\"",
  "agent" => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/
537.36\""
```

# Example: Web server log files

```
"message" => "83.149.9.216 - - [28/May/2014:16:13:42 -0500] \"GET /
presentations/logstash-monitorama-2013/images/kibana-search.png HTTP/1.1\"
200 203023 \"http://semicomplete.com/presentations/logstash-
monitorama-2013/\" \"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/
537.36\"",
```

**grok**

**date**

```
"@version" => "1",
"@timestamp" => "2014-05-28T21:13:42.000Z",
"host" => "kryptic.local",
"clientip" => "83.149.9.216",
"ident" => "-",
"auth" => "-",
"timestamp" => "28/May/2014:16:13:42 -0500",
"verb" => "GET",
"request" => "/presentations/logstash-monitorama-2013/images/
kibana-search.png",
"httpversion" => "1.1",
"response" => "200",
"bytes" => "203023",
"referrer" => "\"http://semicomplete.com/presentations/logstash-
monitorama-2013/\"",
"agent" => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/
537.36\""
```

# Example: Web server log files

```
"geoip" => {
  "ip" => "83.149.9.216",
  "country_code2" => "RU",
  "country_code3" => "RUS",
  "country_name" => "Russian Federation",
  "continent_code" => "EU",
  "region_name" => "48",
  "city_name" => "Moscow",
  "latitude" => 55.75219999999999,
  "longitude" => 37.6156,
  "timezone" => "Europe/Moscow",
  "real_region_name" => "Moscow City",
  "location" => [
    [0] 37.6156,
    [1] 55.75219999999999
  ]
},
"useragent" => {
  "name" => "Chrome",
  "os" => "Mac OS X 10.9.1",
  "os_name" => "Mac OS X",
  "os_major" => "10",
  "os_minor" => "9",
  "device" => "Other",
  "major" => "32",
  "minor" => "0",
  "patch" => "1700"
}
```

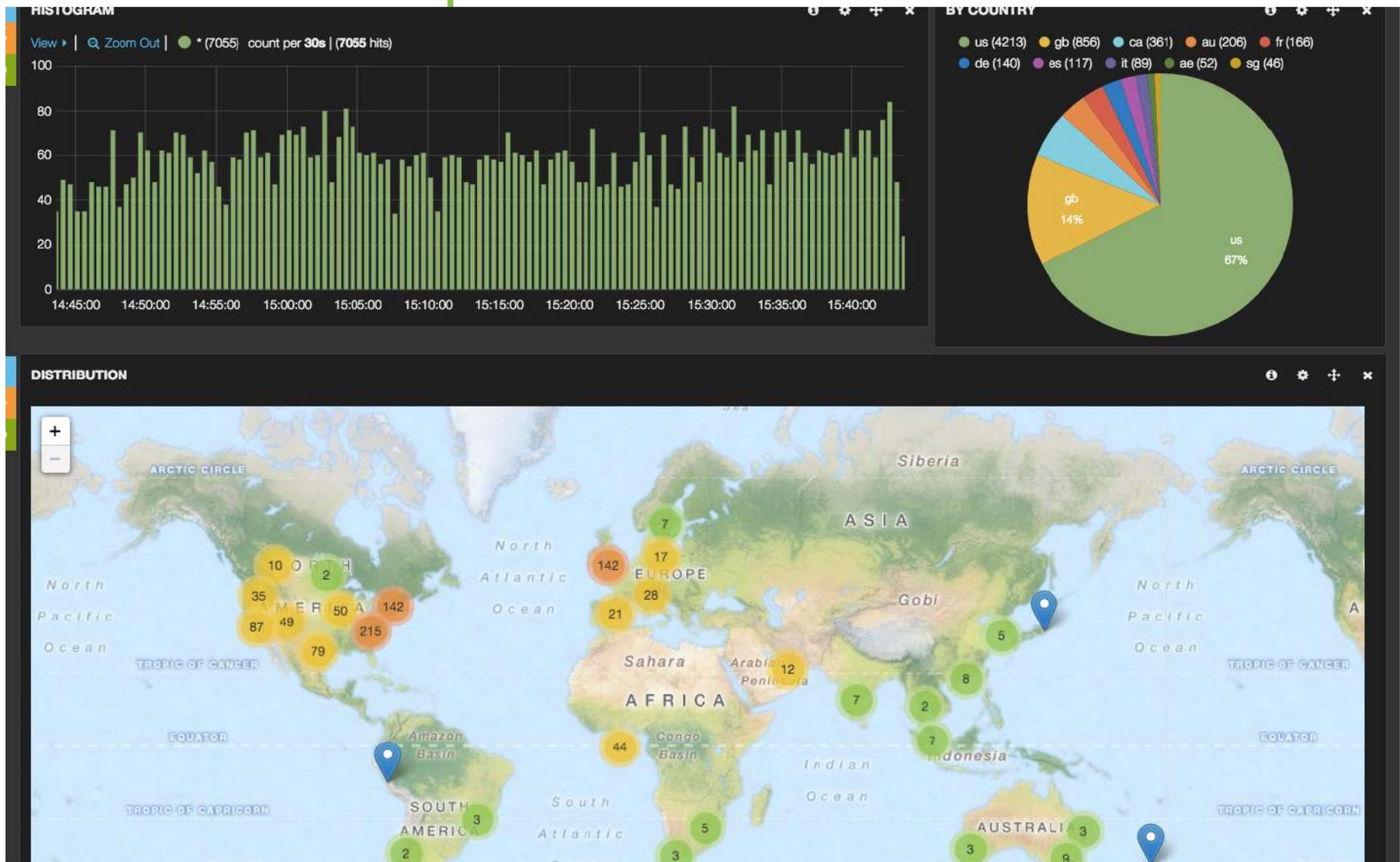
**geoip**

**useragent**

# meetup.com RSVP stream

- All RSVPs are written out to a HTTP stream
- Each line is a JSON document
- Available at *<http://stream.meetup.com/2/rsvps>*

# meetup.com RSVP stream



# meetup.com RSVP stream

```
{
  response: "yes",
  member: { member_name: "Charlie ", member_id: 176530582 },
  visibility: "public",
  event: {
    time: 1413270000000,
    event_url: "http://www.meetup.com/2EuroBootCamp/events/212054422/",
    event_id: "qsvrtkysnbsb", event_name: "Tuesday Morning Boot Camp"
  },
  guests: 0,
  mtime: 1412774717000,
  rsvp_id: 1477279032,
  group: {
    group_name: "2 Euro Boot Camp!!",
    group_city: "Barcelona",
    group_lat: 41.4, group_lon: 2.17,
    group_urlname: "2EuroBootCamp",
    group_id: 17456462,
    group_country: "es",
    group_topics: [ { urlkey: "fitness", topic_name: "Fitness" } ]
  },
  venue: {
    lon: 1.58728,
    venue_name: "Paque de la Espana Industrial",
    venue_id: 22845382,
    lat: 41.462646
  }
}
```

# meetup.com RSVP stream

```
# curl -s http://stream.meetup.com/2/rsvps |  
logstash agent -f logstash-meetup.conf
```

```
input {  
  stdin {  
    codec => json_lines  
    type => 'meetup'  
  }  
}
```

# meetup.com RSVP stream

```
filter {
  if [venue][lat] and [venue][lon] {
    mutate {
      add_field => [ "[venue][lonlat]", "%{[venue][lon]}",
                    "tmplat", "%{[venue][lat]}" ]
    }
    mutate { merge => [ "[venue][lonlat]", "tmplat" ] }
    mutate {
      convert => [ "[venue][lonlat]", "float" ]
      remove => [ "tmplat" ]
    }
  }
}

metrics {
  meter => "meetup.country.%{[group][group_country]}"
  meter => "meetup.country.total"
  add_tag => "metric"
  flush_interval => 60
}
}
```

# meetup.com RSVP stream

```
output {
  if "metric" in [tags] {
    stdout {
      codec => rubydebug
    }
    elasticsearch {
      host => 'localhost'
      index => 'metrics'
      protocol => 'http'
    }
  }
  if [type] == "meetup" {
    elasticsearch {
      host => 'localhost'
      index => 'meetups'
      protocol => 'http'
    }
  }
}
```

# wikipedia edits

- wikipedia has a changes stream
- constantly posted in an IRC channel

# wikipedia edits

```
input {
  irc {
    type => 'wikipedia'
    host => 'irc.wikimedia.org'
    nick => 'logstash-wikipedia'
    channels => ['#de.wikipedia']
  }
}
```

# wikipedia edits

```
filter {
  # remove some weird encoding stuff from IRC
  mutate {
    gsub => [
      "message", "\u000302", "",
      "message", "\u000303", "",
      "message", "\u000307", "",
      "message", "\u000310", "",
      "message", "\u000314", "",
      "message", "\u00034", "",
      "message", "\u00035", "",
      "message", "\u0003", ""
    ]
  }
  # extract page and user
  grok {
    match => [ "message", "[\[%{GREEDYDATA:page}\]\]\%{GREEDYDATA} \*
%{GREEDYDATA:user} \* %{GREEDYDATA}" ]
  }
}
```

# wikipedia edits

```
output {
  stdout {
    codec => line {
      format => 'Page: %{page}'
    }
  }
  elasticsearch {
    host => 'localhost'
    index => 'wikipedia-edits'
    protocol => 'http'
  }
}
```

# wikipedia edits

```
» logstash agent -f logstash-wikipedia.conf
```

```
Page: Yamaha Aerox
```

```
Page: Neues Beginnen - Blätter internationaler Sozialisten
```

```
Page: Portal Diskussion:Fußball
```

```
Page: Saputo
```

```
Page: Portal:Phantastik/Mitarbeiten
```

```
Page: Gesetz über den Einsatz der Informations- und  
Kommunikationstechnik in der öffentlichen Verwaltung
```

```
Page: Spvg Plettenberg
```

```
Page: Pflanzen gegen Zombies: Garden Warfare
```

```
Page: Wasserstandsanzeiger Bremerhaven
```

# sysdig

- sysdig is a system call tracer (tcpdump for syscalls)
- powerful query language
- very useful for system tracing (intrusions, performance tracing, weird behaviour)
- See <http://www.sysdig.org/>

# sysdig

- Easy to find things

```
# sysdig -r dumpfile.scap "evt.type = open and evt.arg.name  
contains /usr/sbin"  
  
2122 13:54:01.755117599 0 bash (1633) < open fd=3(<f>/usr/sbin/  
hacked) name=/usr/sbin/hacked flags=262(O_TRUNC|O_CREAT|O_WRONLY)  
mode=0
```

- Now do this for all machines...

# sysdig

```
input { stdin { } }

filter {

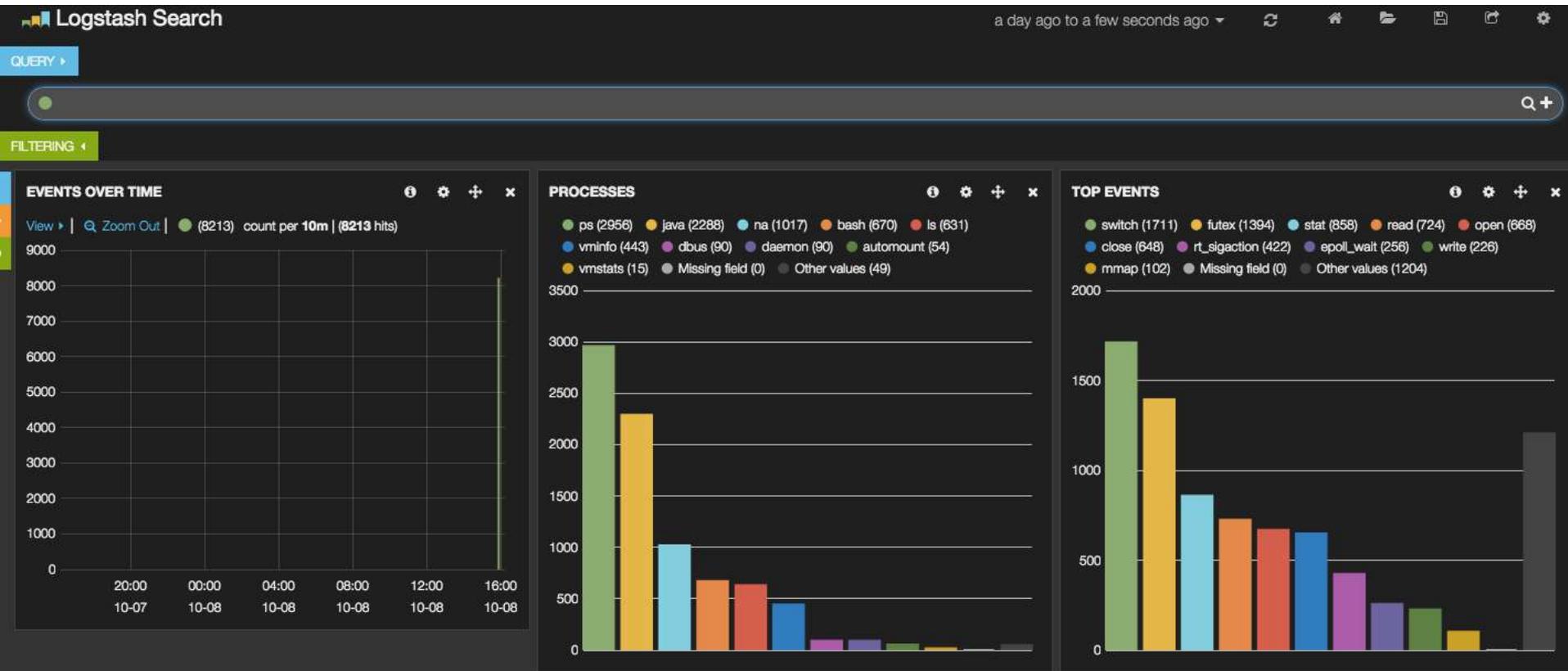
  grok {
    pattern => "^{%NUMBER:num:int} {%NUMBER:time:float} {%INT:cpu:int} %
{NOTSPACE:procname} {%NOTSPACE:tid} (?<direction>[<>]) {%WORD:event} %
{DATA:args}$"
  }

  date { match => [ "time", "UNIX" ] }

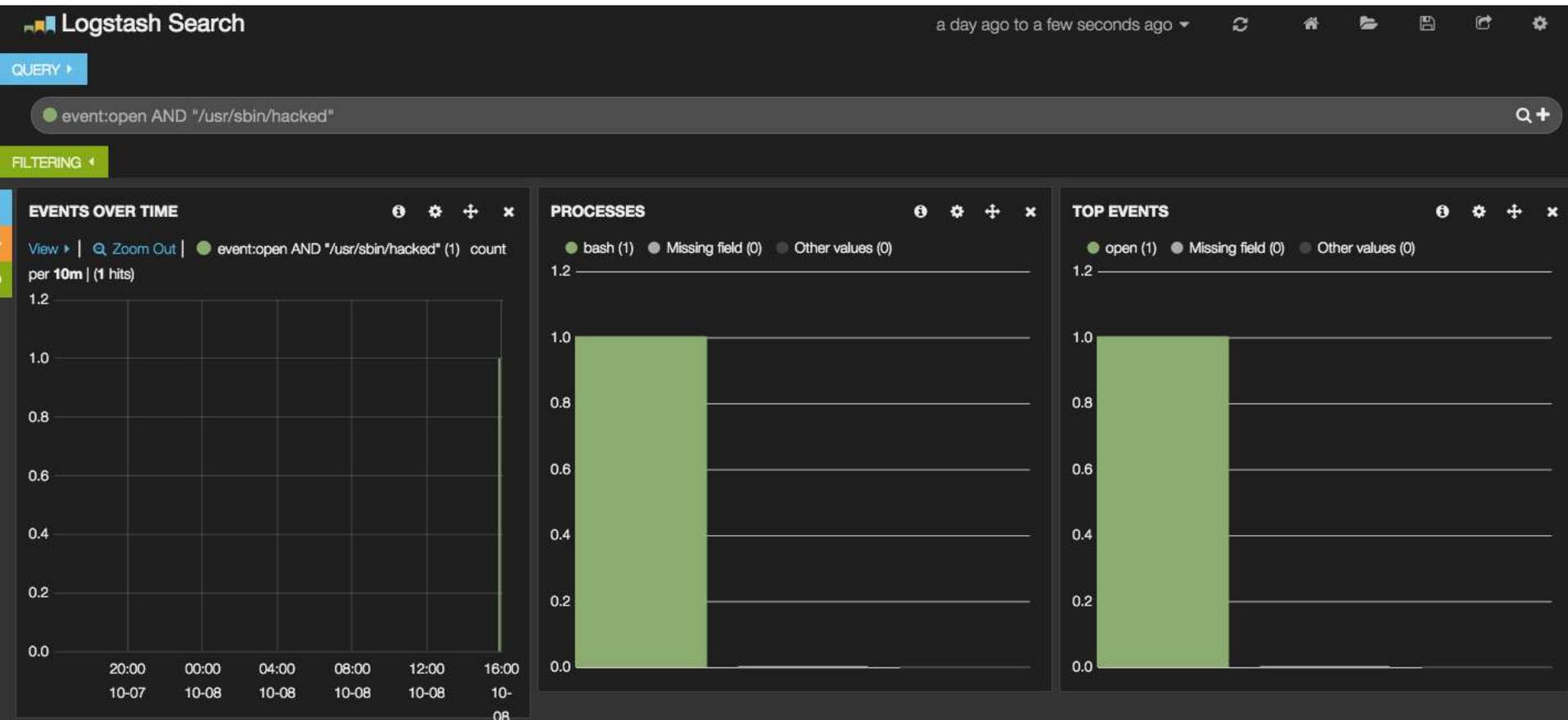
  if [args] {
    kv {
      source => "args"
      remove_field => "args"
    }
  }
}
```

```
output {
  elasticsearch {
    protocol => http
    index => "sysdig-%{+YYYY.MM.dd}"
  }
}
```

# sysdig



# sysdig



# sysdig

● event:open AND "/usr/sbin/hacked"

FILTERING +

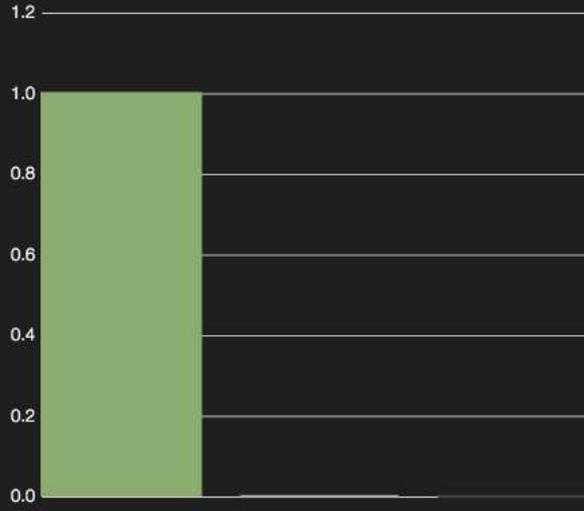
## EVENTS OVER TIME

View | Zoom Out | ● event:open AND "/usr/sbin/hacked" (1) count per 10m | (1 hits)



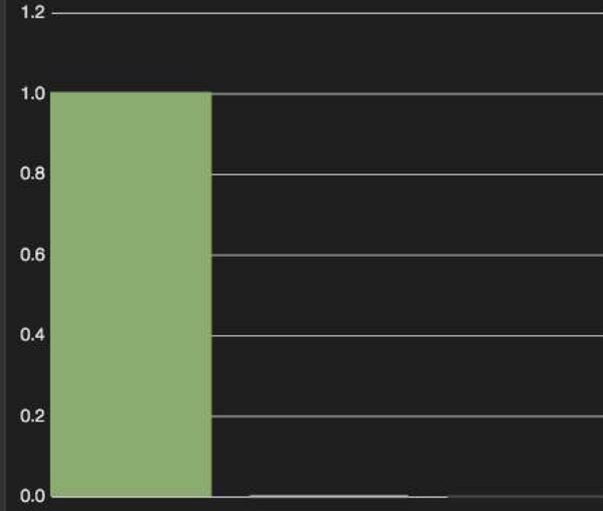
## PROCESSES

● bash (1) ● Missing field (0) ● Other values (0)



## TOP EVENTS

● open (1) ● Missing field (0) ● Other values (0)



# Summary

# Summary

- Do not create data silos. Free your data!
- Make sure data is easy to query, not to store
- Visualize
- Find your use-case: Business, system administration, your app... it's versatile!



elasticsearch.



elasticsearch.

# Soon...

- Kibana 4... is going to be **huge**
- Elasticsearch 1.4.0.Beta1 has been released
- Logstash going towards 1.5.0

# Kibana 4

The screenshot displays the Kibana 4 interface. At the top, there are navigation tabs: Discover, Visualize, Dashboard, and Settings. The 'Discover' tab is active. On the right, there is a search bar and a 'Last 7 days' filter. Below the search bar, the 'Selected Fields' list includes @tags, auth, extension, and g.coordinates. The 'Fields' section is expanded to show popular fields like @message, @timestamp, \_source, \_type, agent, bytes, clientip, longterm, \_id, \_index, g.dst, g.src, and g.srcdst.

The main visualization is a bar chart titled 'September 23rd 2014, 16:06:29.508 - September 30th 2014, 16:06:29.508'. The y-axis is labeled 'Count of documents' and ranges from 0 to 3k. The x-axis is labeled 'timestamp per 3 hours'. A tooltip is visible over the chart, showing: '@timestamp: September 24th 2014, 08:00:00.000' and 'Count of documents: 2005'. The total number of hits is 49606.

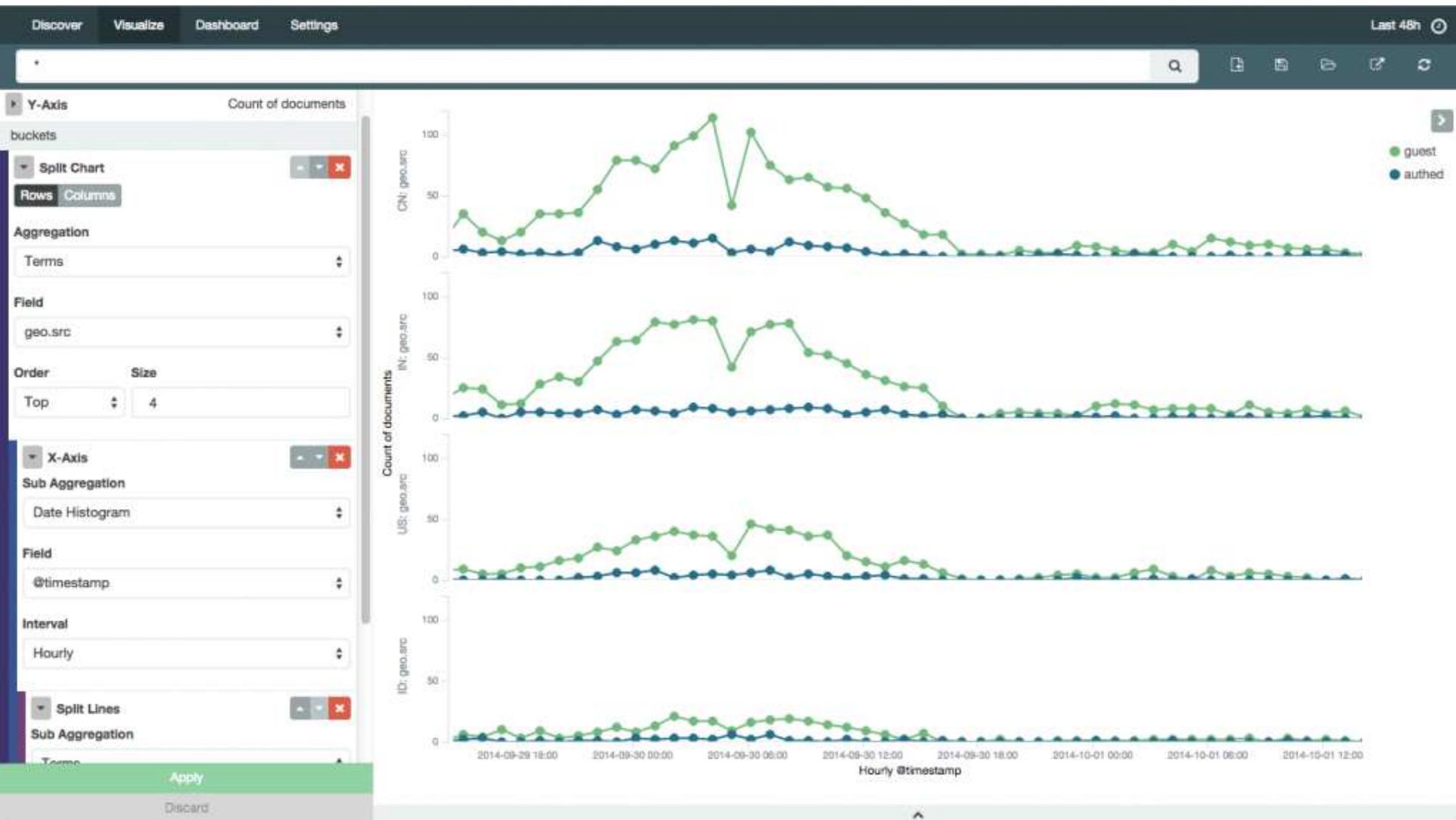
Below the chart, a table displays document details. The table has columns: Time, @tags, auth, extension, and geo.coordinates. Two documents are shown:

Time	@tags	auth	extension	geo.coordinates
September 30th 2014, 16:06:14.468	["success","info"]	guest	png	{"lat":21.31869111,"lon":-157.9224072}
September 30th 2014, 16:06:07.698	["success","info"]	authed	png	{"lat":33.17183583,"lon":-86.30553778}

Below the table, there are two tabs: 'Table' (selected) and 'JSON'. The 'Table' view shows a list of fields with their values:

@message	89.206.207.237 -- [30/Sep/2014:23:06:07 +0000] "GET /paniscus.png HTTP/1.1" 403 0 "-" Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24"
@tags	["success","info"]
@timestamp	September 30th 2014, 16:06:07.698
_id	116028
_index	logstash-2014.09.30
_source	{"id":"116028","index":"logstash-2014.09.30","type":"nginx","geo.coordinates":{"lat":33.17183583,"lon":-86.30553778},"geo.srcdst":"IN:CN","g

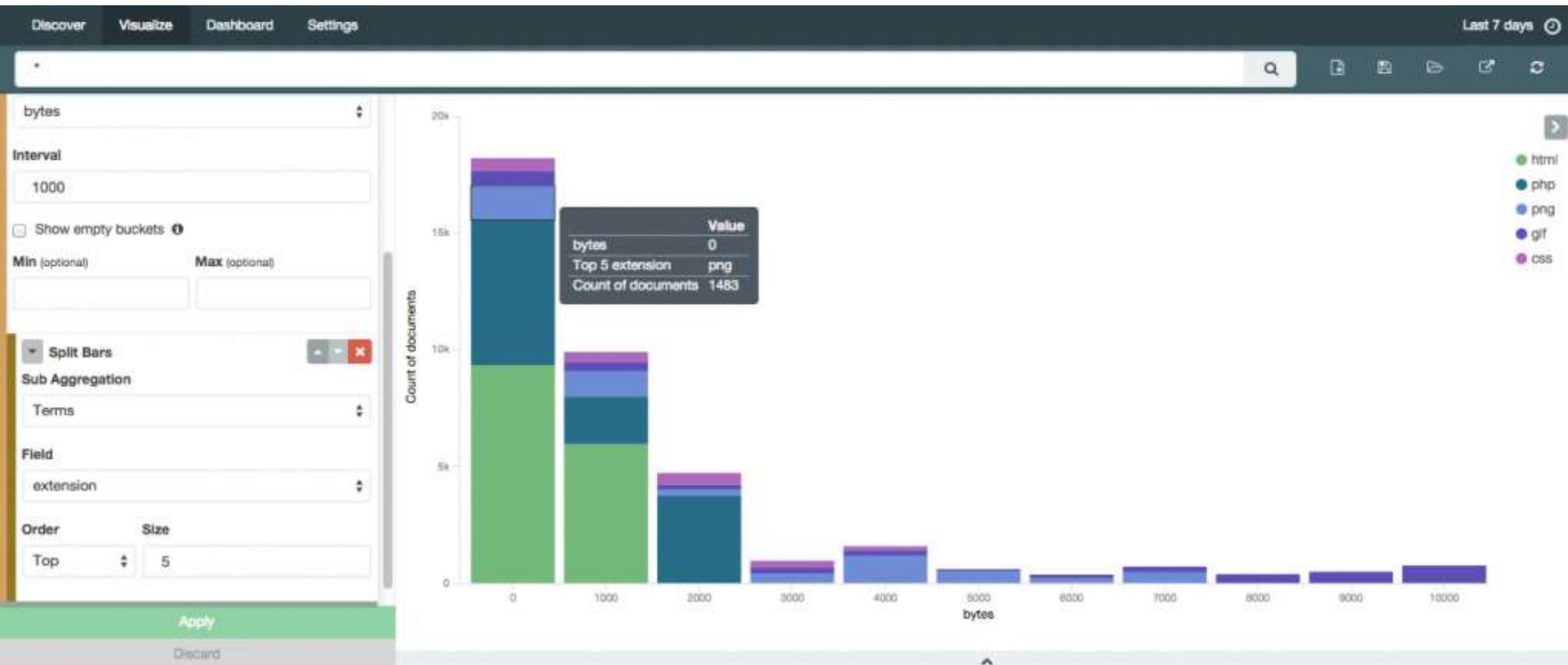
# Kibana 4



# Kibana 4



# Kibana 4



# Kibana 4



# Getting up & running is easy

- Download Elasticsearch, logstash & Kibana archives

```
# elasticsearch-1.4.0.Beta1/bin/elasticsearch  
  
# kibana-4.0.0-BETA1/bin/kibana  
  
# logstash-1.4.2/bin/logstash agent -f logstash.conf  
  
# open localhost:5601
```

# Thanks for listening!

## Q & A

P.S. We're hiring

<http://elasticsearch.com/about/jobs>

P.P.S. We're helping

<http://elasticsearch.com/support>

<http://elasticsearch.com/training>

Alexander Reelsen

@spinscale

[alexander.reelsen@elasticsearch.com](mailto:alexander.reelsen@elasticsearch.com)

**elasticsearch.**