

# Elasticsearch, Logstash & Kibana

Alexander Reelsen

@spinscale

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

# Agenda

- Introduction
- Elasticsearch + Ecosystem

Break: 10:30 - 11:00

- Logstash & Kibana
- Elasticsearch 1.0
- Q & A

# about

- Me

Interested in metrics, ops and the web

Likes the JVM

Working with elasticsearch since 2011

- Elasticsearch, founded in 2012

Products: Elasticsearch, Logstash, Kibana, Marvel

Professional services: Support & development subscriptions

Trainings

# Elasticsearch

# Agenda - Elasticsearch

- Introduction
- Installation, first steps
- Scaling features
- Ecosystem
- Use-cases
- Marvel
- Q & A

# Introduction

# Unstructured search

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

📁 Repositories	317
🔗 Code	17,981
🕒 Issues	2,008
👤 Users	2

Languages

Java	167
Ruby	167
JavaScript	139
Python	117
PHP	69
Shell	49
Puppet	40
Perl	38
Scala	16
C#	13

We've found 317 repository results

Sort: Best match ▾

-  **elasticsearch/elasticsearch** Java ★ 4,583 📄 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago
-  **richardwilly98/elasticsearch-river-mongodb** Java ★ 308 📄 48  
MongoDB River Plugin for **ElasticSearch**  
Last updated 2 minutes ago
-  **jprante/elasticsearch-river-jdbc** Java ★ 170 📄 70  
JDBC river for **Elasticsearch**  
Last updated 12 days ago
-  **elasticsearch/elasticsearch-hadoop** Java ★ 79 📄 28  
Read and write data to/from **ElasticSearch** within Hadoop  
Last updated 3 days ago

elasticsearch.

# Structured search

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

Repositories 317

Code 17,981

Issues 2,008

Users 2

Languages

Java 167

Ruby 139

JavaScript 117

Python 69

PHP 49

Shell 40

Puppet 38

Perl 16

Scala 13

C#

We've found 317 repository results

Sort: Best match

**elasticsearch/elasticsearch** Java ★ 4,583 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago

**richardwilly98/elasticsearch-river-mongodb** Java ★ 308 48  
MongoDB River Plugin for ElasticSearch  
Last updated 2 minutes ago

**jprante/elasticsearch-river-jdbc** Java ★ 170 70  
JDBC river for Elasticsearch  
Last updated 12 days ago

**elasticsearch/elasticsearch-hadoop** Java ★ 79 28  
Read and write data to/from ElasticSearch within Hadoop  
Last updated 3 days ago

# Enrichment

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

Repositories	317
Code	17,981
Issues	2,008
Users	2

Languages

Java	167
Ruby	167
JavaScript	139
Python	117
PHP	69
Shell	49
Puppet	40
Perl	38
Scala	16
C#	13

We've found 317 repository results

Sort: Best match

- elasticsearch/elasticsearch** Java ★ 4,583 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago
- richardwilly98/elasticsearch-river-mongodb** Java ★ 308 48  
MongoDB River Plugin for ElasticSearch  
Last updated 2 minutes ago
- jprante/elasticsearch-river-jdbc** Java ★ 170 70  
JDBC river for Elasticsearch  
Last updated 12 days ago
- elasticsearch/elasticsearch-hadoop** Java ★ 79 28  
Read and write data to/from ElasticSearch within Hadoop  
Last updated 3 days ago

# Sorting

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

Sort: Best match ▾

📁 Repositories	317
🔗 Code	17,981
🕒 Issues	2,008
👤 Users	2

Languages

Java	167
Ruby	167
JavaScript	139
Python	117
PHP	69
Shell	49
Puppet	40
Perl	38
Scala	16
C#	13

We've found 317 repository results

- elasticsearch/elasticsearch** Java ★ 4,583 📄 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago
- richardwilly98/elasticsearch-river-mongodb** Java ★ 308 📄 48  
MongoDB River Plugin for ElasticSearch  
Last updated 2 minutes ago
- jprante/elasticsearch-river-jdbc** Java ★ 170 📄 70  
JDBC river for Elasticsearch  
Last updated 12 days ago
- elasticsearch/elasticsearch-hadoop** Java ★ 79 📄 28  
Read and write data to/from ElasticSearch within Hadoop  
Last updated 3 days ago

elasticsearch.

# Pagination

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

📁	Repositories	317
<>	Code	17,981
🔔	Issues	2,008
👤	Users	2

We've found 317 repository results

Sort: Best match ▾

 **elasticsearch/elasticsearch** Java ★ 4,583 📄 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago

 **spinscale/elasticsearch-suggest-plugin** Java ★ 103 📄 23  
Plugin for **elasticsearch** which uses the lucene FST Suggester  
Last updated 4 days ago

◀ 1 2 3 4 5 6 7 8 9 ... 31 32 ▶

How are these search results? [Tell us!](#)

elasticsearch.

# Aggregation

GitHub

Explore Features Enterprise Blog

Sign up

Sign in

Search

elasticsearch

Search

Repositories 317

Code 7,981

Issues 1,008

Users 2

Languages

Java 167

Ruby 139

JavaScript 117

Python 69

PHP 49

Shell 40

Puppet 38

Perl 16

Scala 13

C#

We've found 317 repository results

Sort: Best match

**elasticsearch/elasticsearch** Java ★ 4,583 1,097  
Open Source, Distributed, RESTful Search Engine  
Last updated 2 hours ago

**richardwilly98/elasticsearch-river-mongodb** Java ★ 308 48  
MongoDB River Plugin for ElasticSearch  
Last updated 2 minutes ago

**jprante/elasticsearch-river-jdbc** Java ★ 170 70  
JDBC river for Elasticsearch  
Last updated 12 days ago

**elasticsearch/elasticsearch-hadoop** Java ★ 79 28  
Read and write data to/from ElasticSearch within Hadoop  
Last updated 3 days ago

# Suggestions



**GitHub** This repository:

**Sign up** **Sign in**

★ **Star** 4,683 **Fork** 1,097

**New Issue**

1 2 3 ... 19

**Labels**

- Lucene 4.5 Upgrade
- breaking
- bug
- enhancement
- feature
- non-issue

**Issues**

Issue Title	Number	Opened by	Time
elasticsearch/elasticsearch#1726 <b>debian</b> package violates naming convention	1	s1monw	14 hours ago
elasticsearch/elasticsearch#3571 <b>debian</b> package init-script: start-stop-daemon ne	11		
elasticsearch/elasticsearch#1681 <b>Debian</b> pkg	10		
elasticsearch/elasticsearch#3286 There is no official <b>debian</b> /ubuntu repository	9		
elasticsearch/elasticsearch#3500 Elasticsearch should include <b>debian</b> 's standard j	9		
elasticsearch/elasticsearch#1526 Moving <b>debian</b> package to maven	1		

Search elasticsearch/elasticsearch for 'debian'

Search GitHub for 'debian'

**Forms** #3702

**reproducible** #3701

**NoShardAvailableActionException** in ES 0.90.3 on startup #3700  
Opened by richardwilly98 a day ago

**Feature Request: Don't reindex the document when updating non-indexed fields** #3696  
Opened by dorian 2 days ago 4 comments

# Elasticsearch in 10 seconds

- Schema-free, REST & JSON based distributed document store
- Open Source: Apache License 2.0
- Zero configuration
- Written in Java, extensible

# Installation & first steps

# Zero configuration

```
$ wget https://download.elasticsearch.org/...  
$ tar -xf elasticsearch-1.0.0.RC2.tar.gz  
$ ./elasticsearch-1.0.0.RC2/bin/elasticsearch  
...  
[2014-01-19 14:53:11,508][INFO ][node] [Scanner] started  
...
```

# Is it alive?

```
» curl localhost:9200
```

```
{  
  "status" : 200,  
  "name" : "Scanner",  
  "version" : {  
    "number" : "1.0.0.RC2",  
    "build_hash" : "e018cda7e7a32643d59e0ac3cdb412ccc239af04",  
    "build_timestamp" : "2014-01-17T15:11:47Z",  
    "build_snapshot" : true,  
    "lucene_version" : "4.6.1"  
  },  
  "tagline" : "You Know, for Search"  
}
```

# Create...

```
» curl -XPUT localhost:9200/books/book/1 -d '{
  "title" : "Elasticsearch – The definitive guide",
  "authors" : "Clinton Gormley",
  "started" : "2013-02-04",
  "pages" : 230
}'
```

# Update...

```
» curl -XPUT localhost:9200/books/book/1 -d '{
  "title" : "Elasticsearch - The definitive guide",
  "authors" : [ "Clinton Gormley", "Zachary Tong" ],
  "started" : "2013-02-04",
  "pages" : 230
}'
```

# Delete...

```
» curl -X DELETE localhost:9200/books/book/1
```

# Realtime GET...

```
» curl -X GET localhost:9200/books/book/1
```

```
» curl -X GET localhost:9200/books/book/1/_source
```

# Search

```
» curl -XGET localhost:9200/books/_search?q=elasticsearch
```

```
{
  "took" : 2, "timed_out" : false,
  "_shards" : { "total" : 5, "successful" : 5, "failed" : 0 },
  "hits" : {
    "total" : 1, "max_score" : 0.076713204,
    "hits" : [ {
      "_index" : "books", "_type" : "book", "_id" : "1",
      "_score" : 0.076713204, "_source" : {
        "title" : "Elasticsearch - The definitive guide",
        "authors" : [ "Clinton Gormley", "Zachary Tong" ],
        "started" : "2013-02-04", "pages" : 230
      }
    } ]
  }
}
```

# Search - Query DSL

```
» curl -XGET 'localhost:9200/books/book/_search' -d '{
  "query": {
    "filtered": {
      "query": {
        "match": {
          "text": {
            "query": "To Be Or Not To Be",
            "cutoff_frequency": 0.01
          }
        }
      }
    },
    "filter": {
      "range": {
        "price": {
          "gte": 20.0
          "lte": 50.0
        }
      }
    }
  }
}'
```

# Scalability

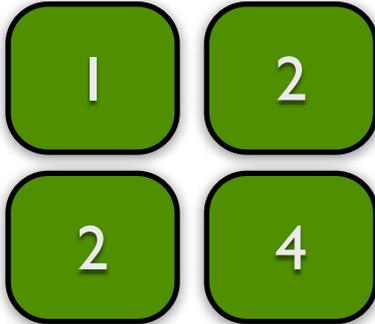
# Distributed & scalable

- Replication
  - Read scalability
  - Removing SPOF
- Sharding
  - Split logical data over several machines
  - Write scalability
  - Control data flows

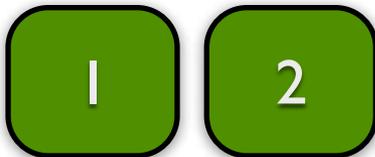
# Distributed & scalable

## node 1

### orders



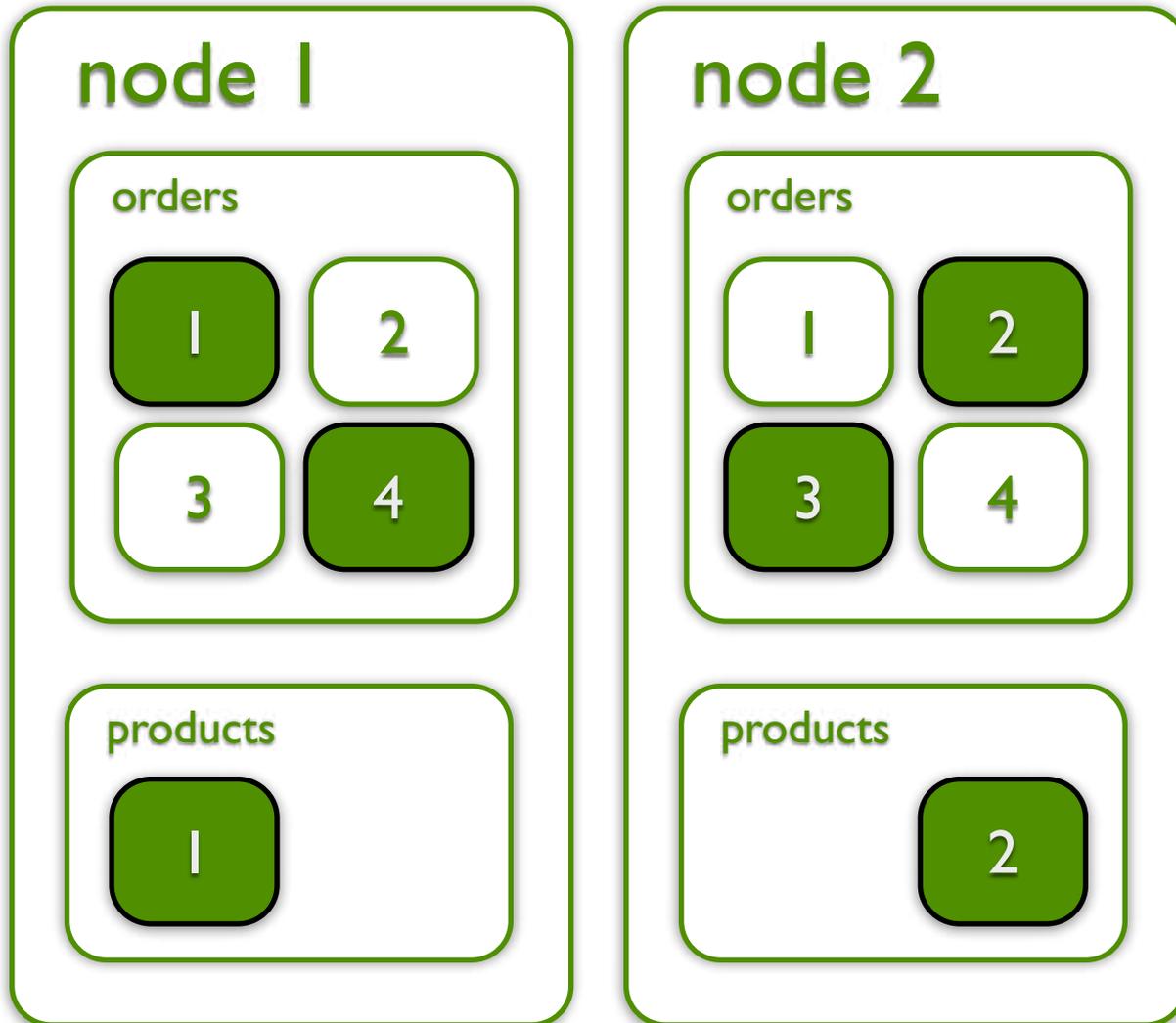
### products



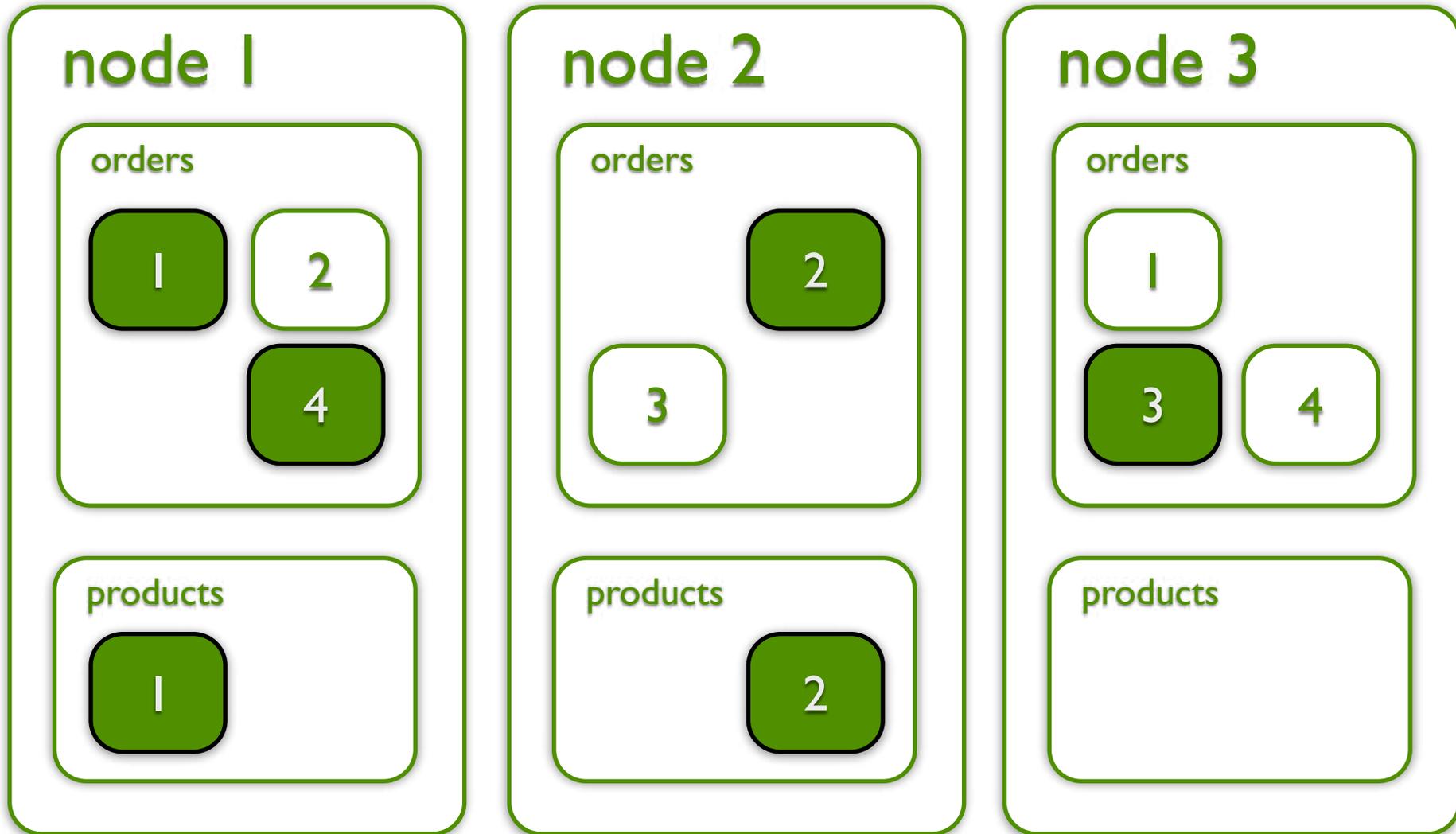
```
curl -X PUT localhost:9200/orders -d '{  
  "settings.index.number_of_shards" : 4  
  "settings.index.number_of_replicas" : 1  
}'
```

```
curl -X PUT localhost:9200/products -d '{  
  "settings.index.number_of_shards" : 2  
  "settings.index.number_of_replicas" : 0  
}'
```

# Distributed and scalable



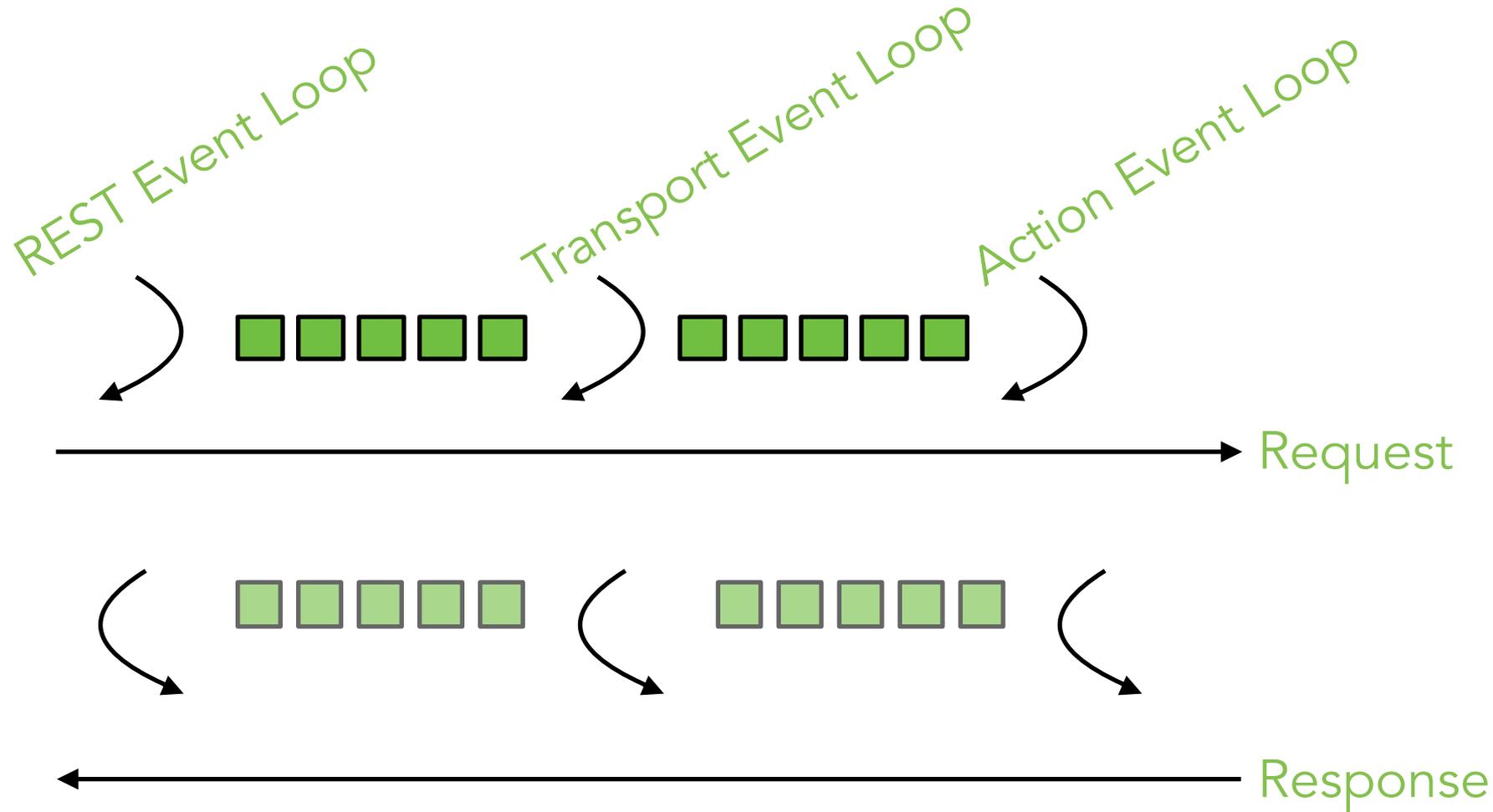
# Distributed & scalable



# Distributed & scalable

- JVM (high level & high performance if done right)
- Netty (async networking on top of the JVM)
- Lucene (fulltext search library)
- HPPC (high performance primitive collections)
- Google Guice (for extension & dependencies)

# A request under the hood



# Think async!

- Enforces event driven architecture
- Support for non-blocking model
- Enforce loose coupling
- Prefers push over pull
- Callback based concurrency
- Helps to avoid contention on resources / threads

# Ecosystem

# Ecosystem

- Plugins
- Clients for many languages  
Ruby, python, php, perl, javascript, (.NET coming)  
Scala, clojure, go
- Kibana
- Logstash
- Hadoop integration

# Elasticsearch use-cases

# What is data?

- Whatever provides value for your business
- Domain data
  - Internal: Orders, products
  - External: Social media streams, email
- Application data
  - Log files
  - Metrics

# Use case: Product search engine

# Product search engine

- Just index all your products and be happy?  
Search is not that easy
- Decomponding, Synonyms, Suggestions, Faceting, Custom scoring, Analytics, Price agents, Query optimization, beyond search

# Domain specific knowledge

- Search term: Topf  
What is expected? Blumentopf? Kochtopf?  
Or: Tuch (Handtuch, Halstuch, Geschirrtuch)  
Or: Decke (Tischdecke, Löschdecke, Mitteldecke)
- Decompounding (compound word token filter)  
Blumentopf also needs to match Leuchtblumentopf
- Synonyms  
Portmonee/Portemonnaie/Geldbörse

# Neutrality? Really?

- Is full-text search relevancy really your preferred scoring algorithm?
- Possible influential factors
  - Age of the product, been ordered in last 24h
  - On stock?
  - Provision
  - No shipping costs
  - Special offer
  - Rating (product or seller)

<http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/query-dsl-function-score-query.html>

# Faceting & Filtering

- Products grouped by  
Category  
Material  
Brand
- Allowing to filter  
All of the facets  
Price range  
Color  
Seller  
Ratings (hard!)

## Kategorien

Elektronik & Foto

Fernseher

+ Mehr...

+ Alle 35 Kategorien

## Versandoption (Was ist das?)

- Kostenlose Lieferung ab EUR 20 Bestellwert

## Displaygröße von Fernsehern

- 51 cm (20") & kleiner  
 53 - 59 cm (21-23")  
 61 - 76 cm (24-30")  
 79 - 99 cm (31-39")  
 102 - 114 cm (40-45")  
 116 - 120 cm (46-47")  
 121 - 140 cm (48-55")  
 142 cm (56") & mehr

## Fernseher-Funktionalität

- Smart / Internet  
 3D  
 HbbTV

## Farbe



## Fernseher-Seitenverhältnis

- 16:9 Wide screen  
 4:3 Standard

## Displaytechnologie von Fernsehern

- LED Backlight  
 LCD  
 Plasma

## Durchschn. Kundenrezension

- ★★★★☆ & mehr  
★★★★☆ & mehr  
★★★☆☆ & mehr  
★☆☆☆☆ & mehr

# Notification with Percolation

- Customer: If a product matches name  $X$  and costs below price  $Y$ , is color  $Z$ , then I want to get a mail  
More likely: Notify customer, when it is back on stock
- Enter percolation!  
Not: Index a document and fire a query  
But: Index a query and check a document against if it matches

<https://speakerdeck.com/javanna/whats-new-in-percolator>

# Other full-text search use cases

- News, Products, Cars, People, Auctions, Tickets
- Intranet document search
- Social media streams
- Emails
- Source code

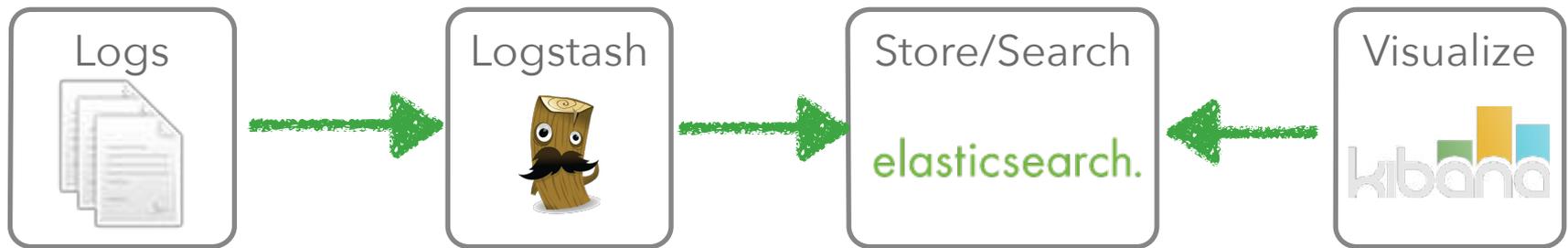


# Use-case: Log file analysis

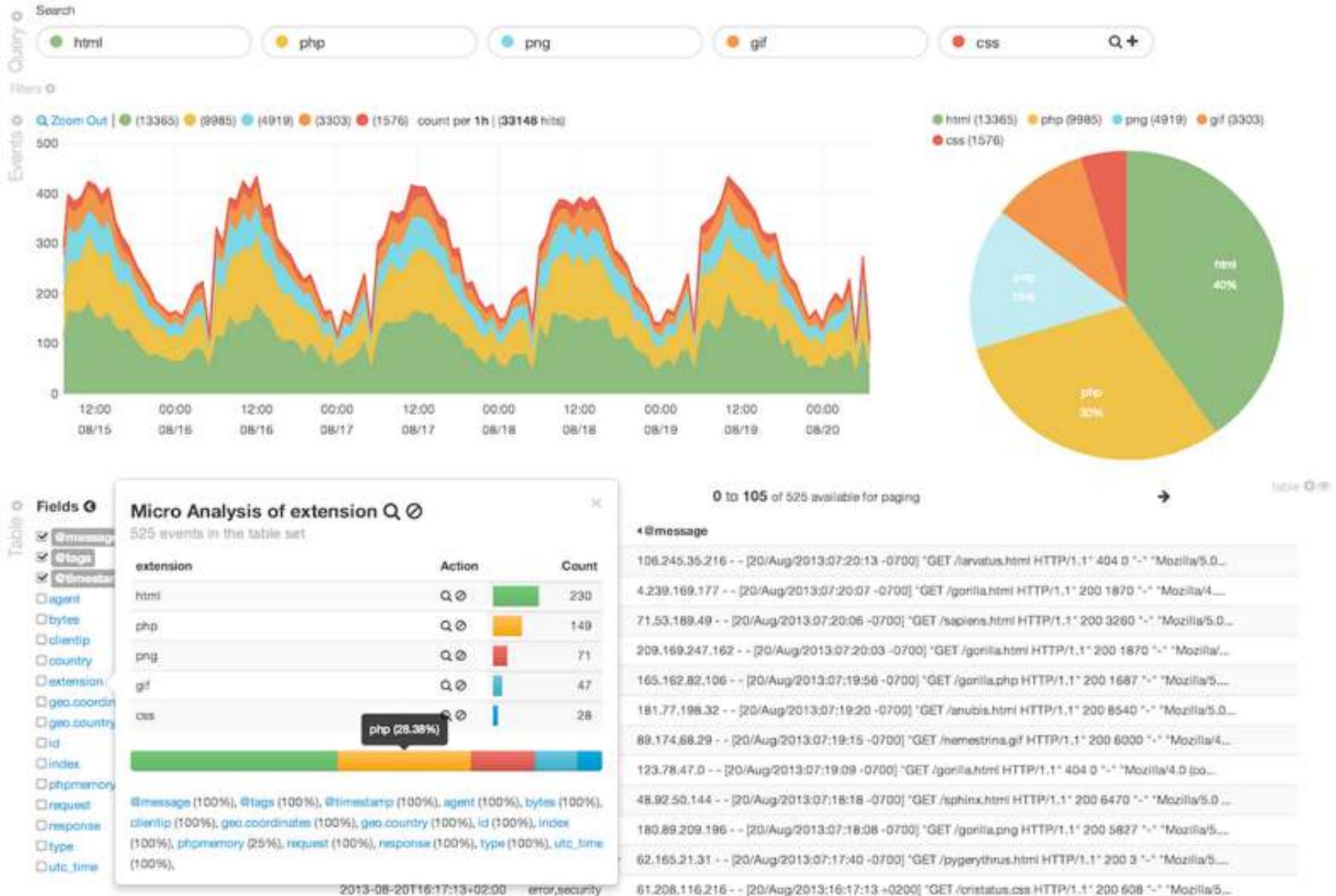
# logstash

- Managing events and logs
- Collect data
- Parse data
- Enrich data
- Store data (search and visualizing)

# Use case: Log files



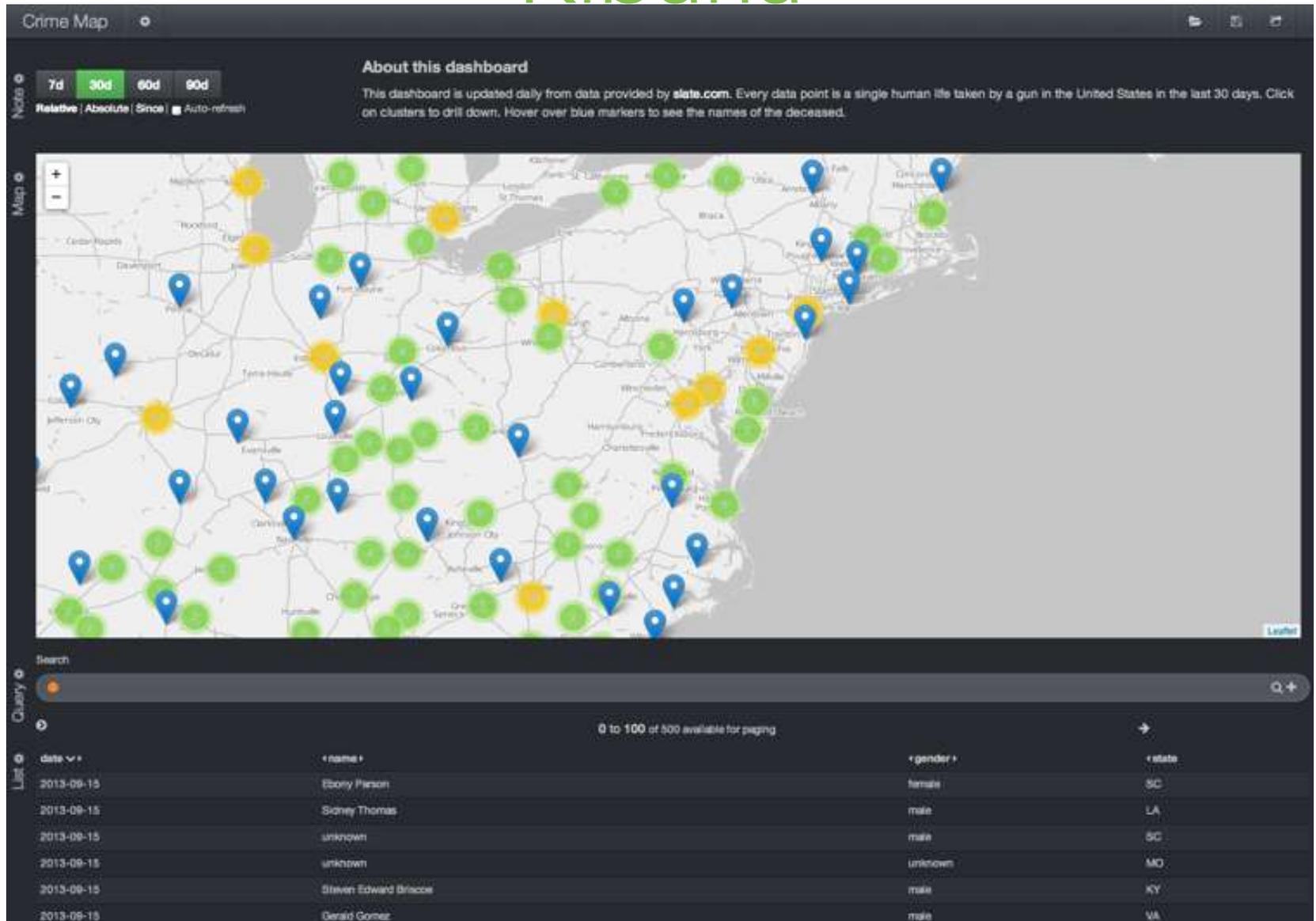
# Kibana



# Kibana



# Kibana



# Kibana



# Use-case: Analytics

# Analytics

- Aggregation of information
- Facets are one dimensional  
Categories/brands/material of all results of this query
- Questions are multidimensional  
Average revenue per category id per day
- Elasticsearch 1.0 will have aggregations

# Create knowledge from data

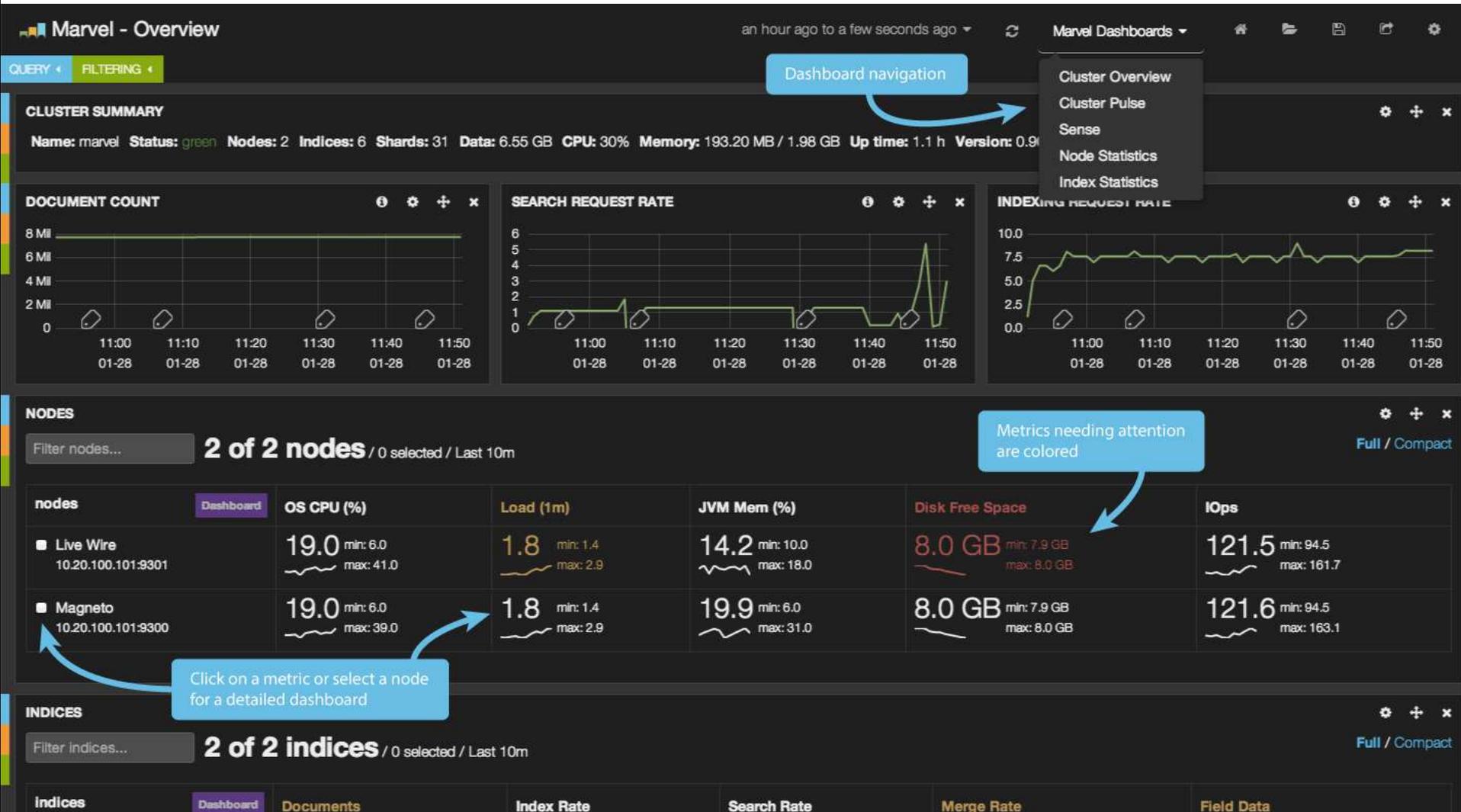
- Orders
  - How many orders were created every day in the last month?
  - How many orders were created per state in the last month?
- Money
  - What is the average revenue per shopping cart?
  - What is the average shopping cart size per order per hour?
- Product portfolio
  - Take the location of people into account for special offers?
  - Analyse page views: Premium or low budget ecommerce site?

# Marvel

# Monitor your cluster

- ... or have it monitored
- Point in time views are a start
- Visualize cluster behaviour, act before problems
  
- Free for development, 500\$/year for up to 5 nodes

# Overview



# Cluster Pulse

**Marvel - Cluster Pulse** an hour ago to a few seconds ago refreshed every 1m Marvel Dashboards

QUERY +

Pinned + Node events + Index events + Routing events + Q +

FILTERING + ★

### CLUSTER SUMMARY

Name: marvel Status: green Nodes: 2 Indices: 6 Shards: 31 Data: 6.54 GB CPU: 18% Memory: 373.04 MB / 1.98 GB Up time: 1.1 h Version: 0.90.10

### TIME LINE

View | ● Node events (6) ● Index events (40) ● Routing events (101) count per 30s | (147 hits)

Cluster events timeline

### CLUSTER EVENTS

0 to 100 of 147 available for paging

@timestamp	_type	event	message
2014-01-28T11:52:54.279+01:00	index_event	index_status	[.marvel-2014.01.28] status is GREEN
2014-01-28T11:52:54.279+01:00	routing_event	shard_started	[.marvel-2014.01.28][0][R] started on [Inferno][10.20.100.101:9301]
2014-01-28T11:52:46.016+01:00	index_event	index_status	[.marvel-kibana] status is GREEN
2014-01-28T11:52:46.016+01:00	routing_event	shard_started	[.marvel-kibana][0][R] started on [Inferno][10.20.100.101:9301]
2014-01-28T11:52:45.294+01:00	index_event	index_status	[stack] status is GREEN
2014-01-28T11:52:45.294+01:00	routing_event	shard_started	[stack][2][P] started on [Inferno][10.20.100.101:9301]
2014-01-28T11:52:45.294+01:00	routing_event	shard_initializing	[.marvel-kibana][0][R] initializing on [Inferno][10.20.100.101:9301]
2014-01-28T11:52:44.392+01:00	index_event	index_status	[.marvel-2013.11.23] status is GREEN
2014-01-28T11:52:44.392+01:00	routing_event	shard_initializing	[.marvel-2014.01.28][0][R] initializing on [Inferno][10.20.100.101:9301]
2014-01-28T11:52:44.392+01:00	routing_event	shard_started	[.marvel-2013.11.23][1][R] started on [Inferno][10.20.100.101:9301]

### EVENT TYPES

Term	Count	Action
routing_event	101	Q
index_event	40	Q
node_event	6	Q

Quickly filter by event type

# Node statistics



# Index statistics



# Sense

Server localhost:9200 Submit request to Elasticsearch Marvel Dashboards

```
1 # search for a super hero
2 GET marvel/superhero/_search
3 {
4   "query": {
5     "match": {
6       "name": "spiderman"
7     }
8   }
9   "name" string
10  "powers" string
11  "enemies" string
12  "rating" long
13 # index a doc
14 PUT marvel/superhero/spiderman
15 {
16   "name": "Spiderman",
17   "powers": ["webbing", "climbing", "night vision"],
18   "enemies": ["the green gobiln", "venom"]
19 }
20 # create an index
21 PUT marvel
22 {
23   "settings": {
24     "number_of_shards": 2,
25     "number_of_replicas": 1
26   },
27   "mappings": {
28     "superhero": {
29       "properties": {
30         "name": { "type": "string" },
31         "powers": {
32           "type": "string",
33           "index": "not_analyzed"
34         }
35       }
36     }
37   }
38 }
39 # index a doc
40 PUT marvel/superhero/venom
41 {
42   "name": "Venom",
43   "rating": 5
44 }
45 # index a doc
46 PUT marvel/superhero/green Goblin
```

**Suggestions as you type**

```
1 {
2   "took": 6,
3   "timed_out": false,
4   "_shards": {
5     "total": 2,
6     "successful": 2,
7     "failed": 0
8   },
9   "hits": {
10    "total": 1,
11    "max_score": 1,
12    "hits": [
13      {
14        "_index": "marvel",
15        "_type": "superhero",
16        "_id": "spiderman",
17        "_score": 1,
18        "_source": {
19          "name": "Spiderman",
20          "powers": [
21            "webbing",
22            "climbing",
23            "night vision"
24          ],
25          "enemies": [
26            "the green gobiln",
27            "venom"
28          ]
29        }
30      }
31    ]
32  }
33 }
```

**API response**

# Elasticsearch 1.0

# Elasticsearch 1.0

- Aggregations
- Snapshot/Restore
- Distributed/scalable percolator
- Cat API  
<http://www.elasticsearch.org/blog/introducing-cat-api/>
- Federated search: Tribe node

# Thanks for listening

P.S. We're hiring  
<http://elasticsearch.com/about/jobs>  
<http://elasticsearch.com/support>

Alexander Reelsen  
@spinscale  
alexander.reelsen@elasticsearch.com

elasticsearch.

# Logstash & Kibana



Alexander Reelsen

@spinscale

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

# Enter logstash

- Managing events and logs
- Collect data
- Parse data
- Enrich data
- Store data (search and visualizing)

# Why collect & centralise data?

- Access log files without system access
- Shell scripting: Too limited or slow
- Using unique ids for errors, aggregate it across your stack
- Reporting (everyone can create his/her own report)
- Bonus points: Unify your data to make it easily searchable

# Unify dates

- apache

```
[23/Jan/2014:17:11:55 +0000]
```

- unix timestamp

```
1390994740
```

- log4j

```
[2014-01-29 12:28:25,470]
```

- postfix.log

```
Feb 3 20:37:35
```

- ISO 8601

```
2009-01-01T12:00:00+01:00  
2014-01-01
```

# Enter logstash

- Managing events and logs
  - Collect data
  - Parse data
  - Enrich data
  - Store data (search and visualizing)
- } **Input**
- } **Filter**
- } **Output**

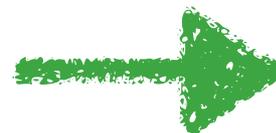
# Logstash architecture

Input

Filter

Output

?



?

# Inputs

- Monitoring: collectd, graphite, ganglia, snmptrap, zenoss
- Datastores: elasticsearch, redis, sqlite, s3
- Queues: rabbitmq, zeromq
- Logging: eventlog, lumberjack, gelf, log4j, relp, syslog, varnish log
- Platforms: drupal\_dblog, gemfire, heroku, sqs, s3, twitter
- Local: exec, generator, file, stdin, pipe, unix
- Protocol: imap, irc, stomp, tcp, udp, websocket, wmi, xmpp

# Outputs

- Store: elasticsearch, gemfire, mongodb, redis, riak, rabbitmq, solr
- Monitoring: ganglia, graphite, graphstastic, nagios, opentsdb, statsd, zabbix
- Notification: email, hipchat, irc, pagerduty, sns
- Protocol: gelf, http, lumberjack, metriccatcher, stomp, tcp, udp, websocket, xmpp
- External Monitoring: boundary, circonus, cloudwatch, datadog, librato
- External service: google big query, google cloud storage, jira, loggly, riemann, rabbitmq, s3, sqs, syslog, zeromq
- Local: csv, exec, file, pipe, stdout, null

# Installation

- ruby application, but Java required (JRuby)
- Download single jar, deb, RPM (also repositories)  
no gem/dependency hell!
- Puppet module

# Simple setup

- Download, create config and run

```
input {
  stdin {}
}

output {
  stdout { debug => true }
}
```

← simple.conf



```
echo foo | java -jar logstash-1.3.3-flatjar.jar agent -f simple.conf
{
  "message" => "foo",
  "@version" => "1",
  "@timestamp" => "2014-01-20T13:30:59.648Z",
  "host" => "kryptic.fritz.box"
}
```

# Analyze the output

```
{  
  "message" => "foo",  
  "@version" => "1",  
  "@timestamp" => "2014-01-20T13:30:59.648Z",  
  "host" => "kryptic.fritz.box"  
}
```

- message: Original content
- version: internal
- timestamp: Current timestamp
- host: Logstash hostname

# But what about filtering?

```
input {
  stdin {}
}

filter {
  grok {
    match => [ "message", "%{WORD:firstname} %{WORD:lastname} %{NUMBER:age}"
  ]
}

output {
  stdout { debug => true }
}
```

# But what about filtering?

```
echo "Alexander Reelsen 30" | java -jar
logstash-1.3.3-flatjar.jar agent -f sample-2.conf
{
    "message" => "Alexander Reelsen 30",
    "@version" => "1",
    "@timestamp" => "2014-01-21T16:56:02.502Z",
    "host" => "kryptic",
    "firstname" => "Alexander",
    "lastname" => "Reelsen",
    "age" => "30"
}
```

# Syslog example with grok

```
input { stdin {} }

filter {
  grok {
    match => { "message" => "%
{SYSLOGTIMESTAMP:syslog_timestamp} %
{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}(?:\[%
{POSINT:syslog_pid}\])?: %{GREEDYDATA:syslog_message}" }
  }
  date {
    match => [ "syslog_timestamp",
              "MMM d HH:mm:ss", "MMM dd HH:mm:ss" ]
  }
}

output { stdout { debug => true } }
```

# Syslog example with grok

```
cat sample-syslog.txt| java -jar logstash-1.3.3-  
flatjar.jar agent -f sample-syslog.conf  
{  
    "message" => "Jun 10 04:04:01  
lvps109-104-93-171 postfix/smtpd[11105]: connect from  
mail-we0-f196.google.com[74.125.82.196]",  
    "@version" => "1",  
    "@timestamp" => "2014-06-10T04:04:01.000+02:00",  
    "host" => "kryptic.local",  
    "syslog_timestamp" => "Jun 10 04:04:01",  
    "syslog_hostname" => "lvps109-104-93-171",  
    "syslog_program" => "postfix/smtpd",  
    "syslog_pid" => "11105",  
    "syslog_message" => "connect from mail-we0-  
f196.google.com[74.125.82.196]"  
}
```

# Syslog example with grok

```
Jun 10 04:04:01 lvps109-104-93-171 postfix/smtpd[11105]:
connect from mail-we0-f196.google.com[74.125.82.196]
:
{
    "message" => "Jun 10 04:04:01
lvps109-104-93-171 postfix/smtpd[11105]: connect from
mail-we0-f196.google.com[74.125.82.196]",
    "@version" => "1",
    "@timestamp" => "2014-06-10T04:04:01.000+02:00",
    "host" => "kryptic.local",
    "syslog_timestamp" => "Jun 10 04:04:01",
    "syslog_hostname" => "lvps109-104-93-171",
    "syslog_program" => "postfix/smtpd",
    "syslog_pid" => "11105",
    "syslog_message" => "connect from mail-we0-
f196.google.com[74.125.82.196]"
}
```

# Filters

- alter, anonymize, checksum, csv, drop, multiline
- dns, date, extractnumbers, geoip, i18n, kv, noop, ruby, range
- json, urldecode, useragent
- metrics, sleep
- ... many, many more ...

# Codecs

- Format conversion
- netflow, fluent, json\_lines, json, msgpack, collectd

# JSON codec

```
input {
  stdin {
    codec => json
  }
}

output {
  stdout { debug => true }
}
```

```
(echo -e '{"foo":"bar", "spam" : "eggs"\n} ' ) | java -jar
logstash-1.3.3-flatjar.jar agent -f sample-json-codec.conf
{
  "foo" => "bar",
  "spam" => "eggs",
  "@version" => "1",
  "@timestamp" => "2014-01-23T13:12:17.325Z",
  "host" => "kryptic.local"
}
```

# JSON multiline codec

```
input { stdin { codec => json_multi } }
output { stdout { debug => true } }
```

```
(echo -e '{"foo":"bar", "spam" : "eggs" }' ; echo '{ "c":"d", "e": "f"
}') | java -jar logstash-1.3.3-flatjar.jar agent -f sample-json-multi-
codec.conf
{
    "foo" => "bar",
    "spam" => "eggs",
    "@version" => "1",
    "@timestamp" => "2014-01-23T13:17:47.582Z",
    "host" => "kryptic.local"
}
{
    "c" => "d",
    "e" => "f",
    "@version" => "1",
    "@timestamp" => "2014-01-23T13:17:47.584Z",
    "host" => "kryptic.local"
}
```

# CLF log files

```
193.99.144.85 - - [23/Jan/2014:17:11:55 +0000] "GET / HTTP/1.1" 200 140  
"-" "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.19 (KHTML, like  
Gecko) Chrome/18.0.1025.5 Safari/535.19"
```

```
193.99.144.85 - - [23/Jan/2014:17:11:55 +0000] "GET /myimage.jpg HTTP/  
1.1" 200 140 "-" "Googlebot"
```

```
input { stdin {} }  
  
filter {  
  grok {  
    match => [ message, "%{COMBINEDAPACHELOG}" ]  
  }  
}  
  
output { stdout { debug => true } }
```

# CLF log files

```
{
  "message" => "193.99.144.85 - - [23/Jan/2014:17:11:55 +0000]
  \"GET / HTTP/1.1\" 200 140 \"-\" \"Mozilla/5.0 (Windows NT 6.1; WOW64)
  AppleWebKit/535.19 (KHTML, like Gecko) Chrome/18.0.1025.5 Safari/
  535.19\"",
  "@version" => "1",
  "@timestamp" => "2014-01-24T07:56:02.460Z",
  "host" => "kryptic.local",
  "clientip" => "193.99.144.85",
  "ident" => "-",
  "auth" => "-",
  "timestamp" => "23/Jan/2014:17:11:55 +0000",
  "verb" => "GET",
  "request" => "/",
  "httpversion" => "1.1",
  "response" => "200",
  "bytes" => "140",
  "referrer" => \"-\",
  "agent" => \"Mozilla/5.0 (Windows NT 6.1; WOW64)
  AppleWebKit/535.19 (KHTML, like Gecko) Chrome/18.0.1025.5 Safari/
  535.19\"
}
```

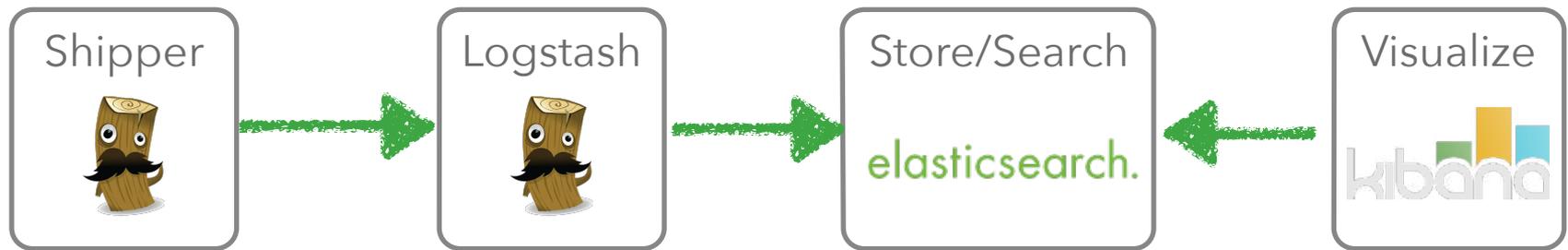
# Write to elasticsearch

```
input { stdin {} }

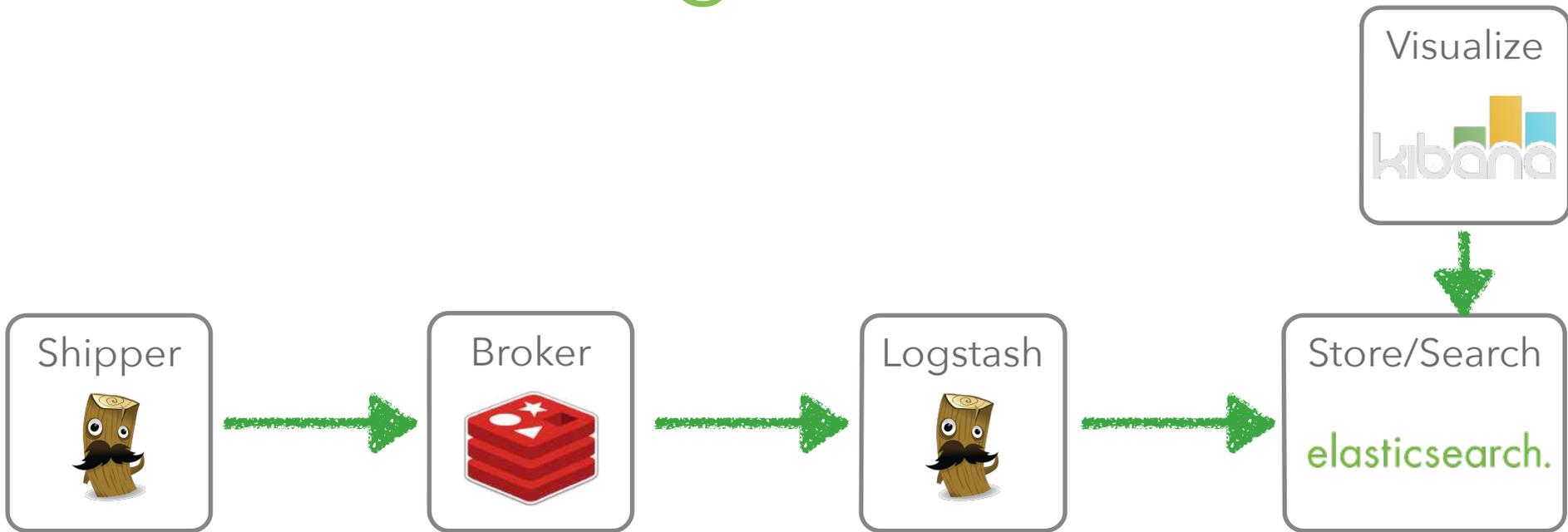
filter {
  grok {
    match => [ message, "%{COMBINEDAPACHELOG}" ]
  }
}

output {
  elasticsearch_http {}
}
```

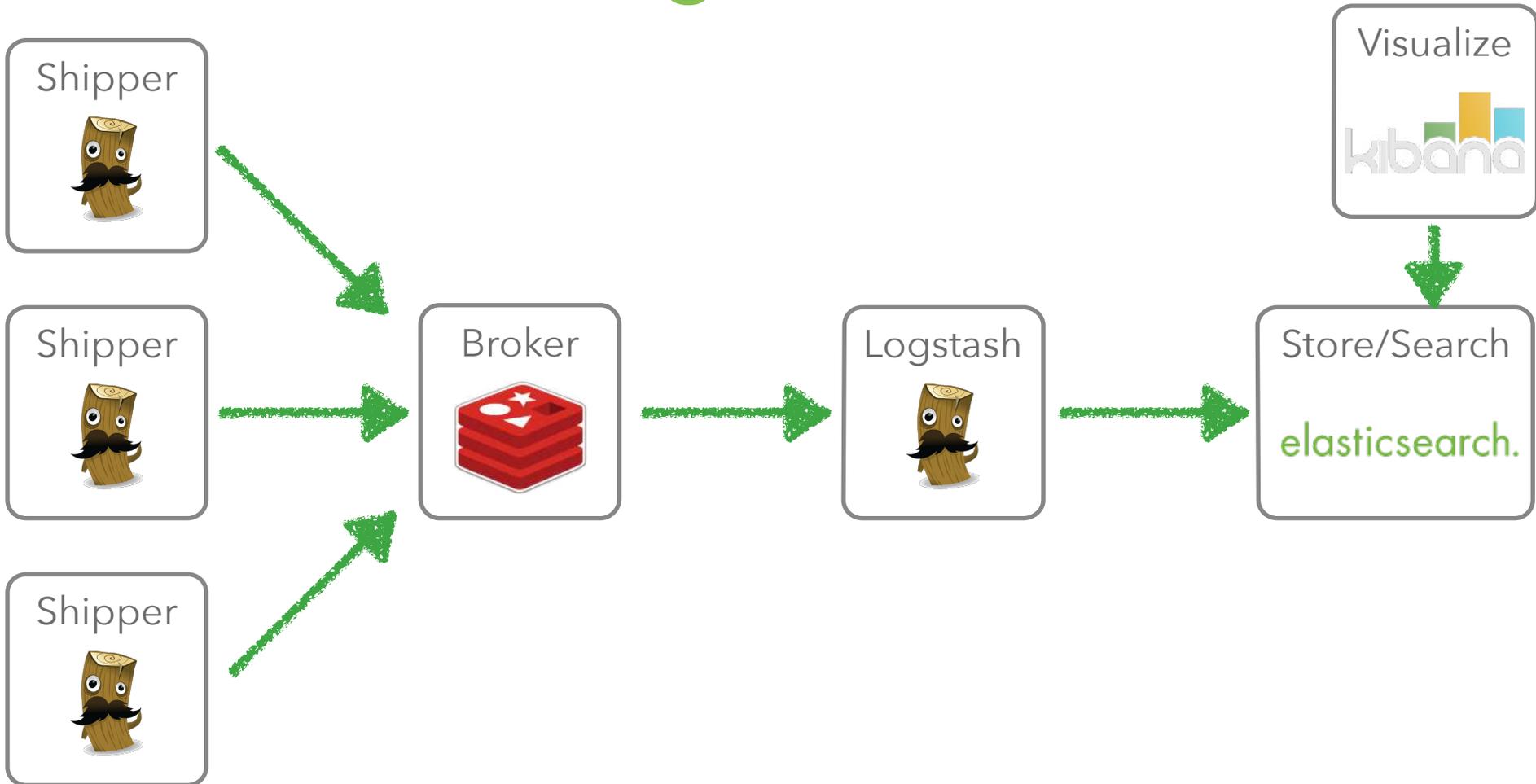
# Use case: Log files



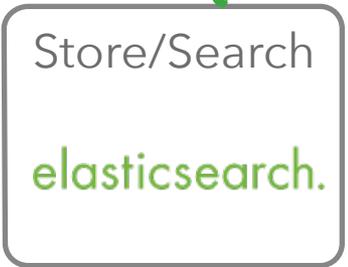
# Use case: Log files with broker



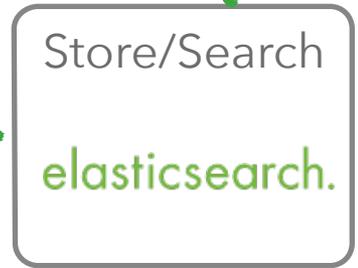
# Use case: Log files with broker



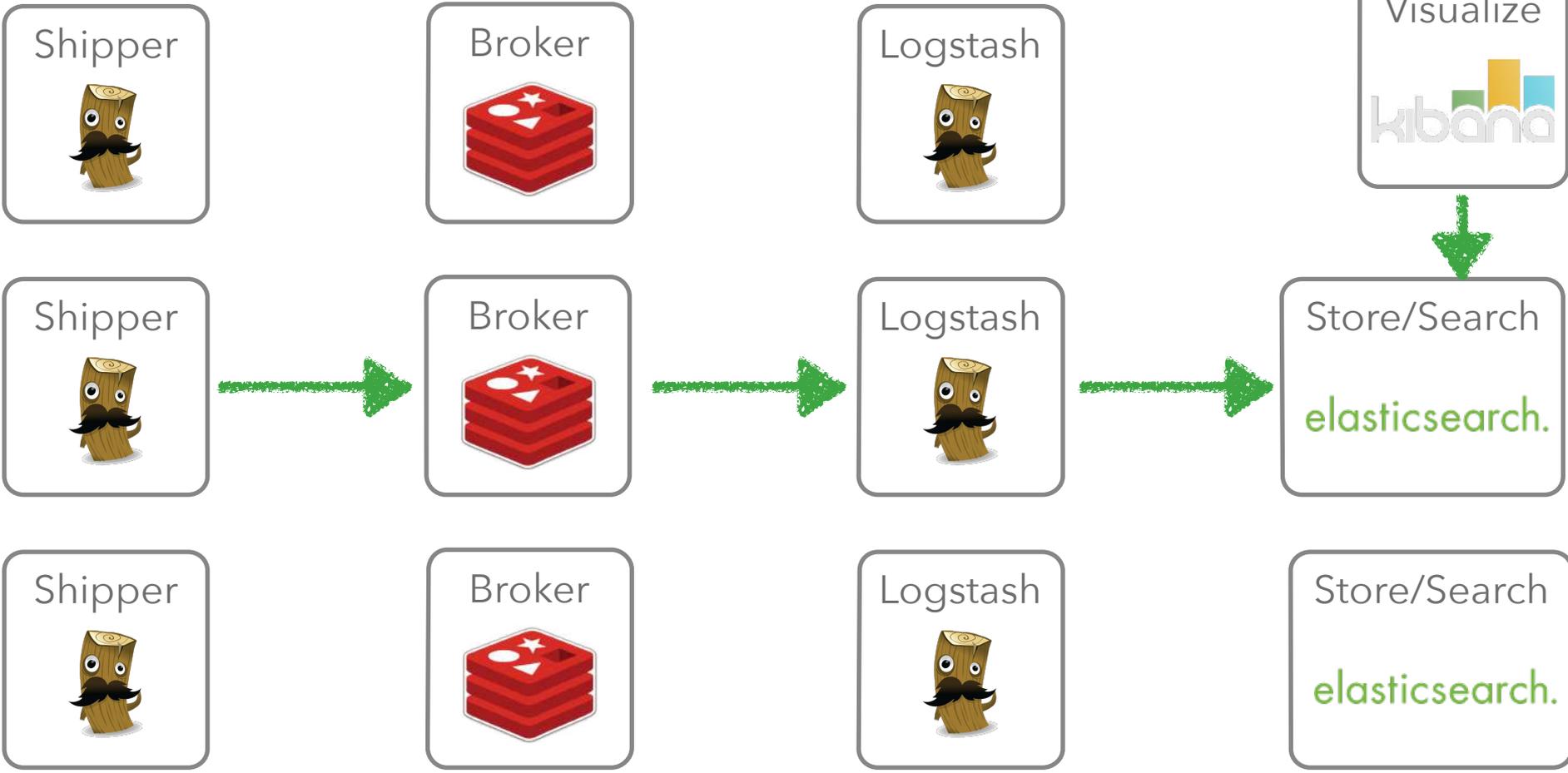
# Scale out any component



# Scale out any component



# Scale any component



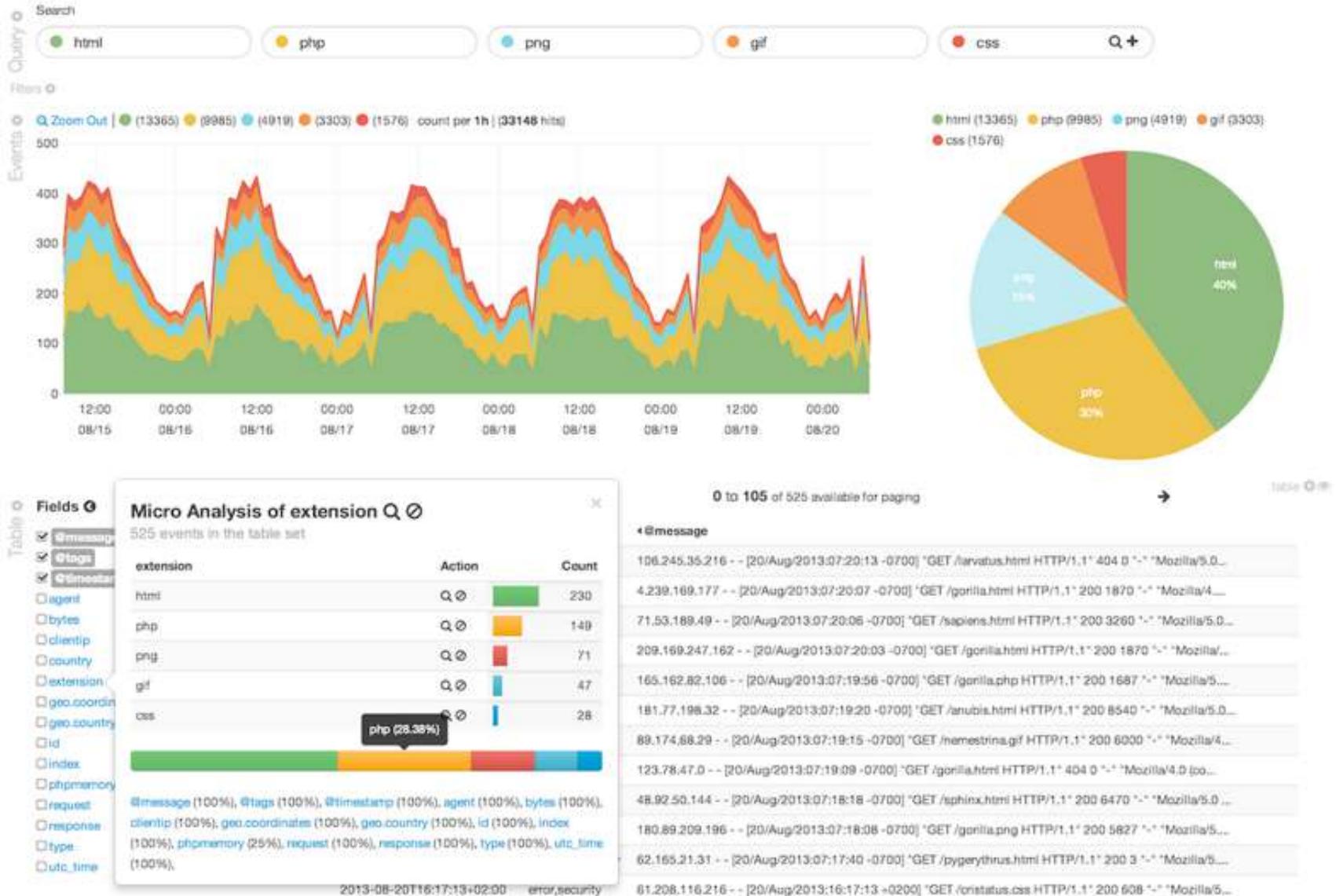
# Logstash scaling

- Events get passed via ruby SizedQueue
- input/worker/output threads, can be configured
- each input is one thread, unless explicitly configurable
- one worker thread by default, use -w to change
- output is a single thread (some outputs have their own queueing thread)

<http://logstash.net/docs/1.3.3/life-of-an-event>

# Kibana

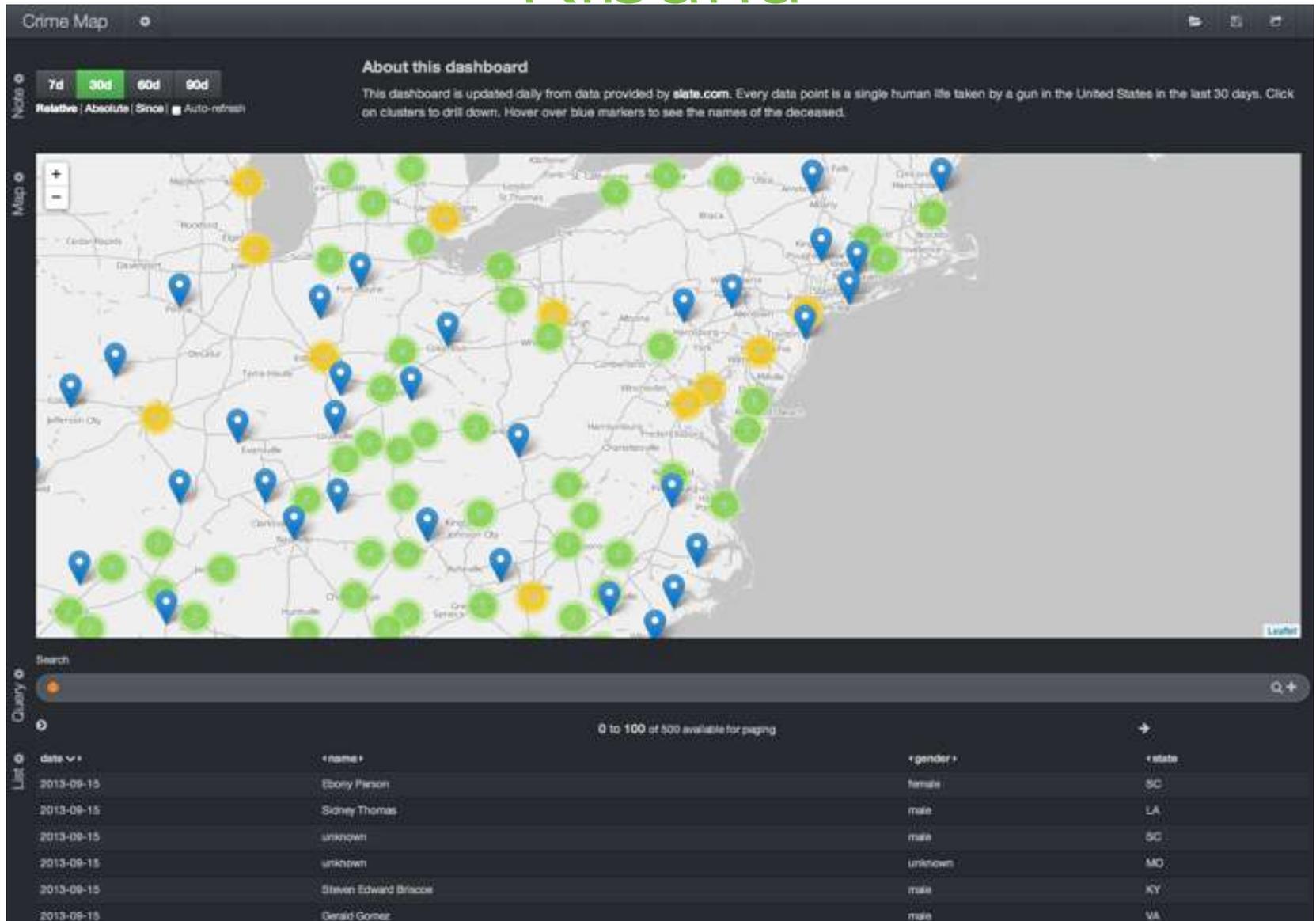
# Kibana



# Kibana



# Kibana



# Kibana



# Tools

# Useful helpers

- Curator

<http://www.elasticsearch.org/blog/curator-tending-your-time-series-indices/>

- Puppet module

<https://github.com/elasticsearch/puppet-logstash>

- logstash forwarder

<https://github.com/elasticsearch/logstash-forwarder>

- Logstash cookbook

<http://cookbook.logstash.net/>

# Demo - Meetup RSVP stream

# Demo - Wikipedia changes

# Elasticsearch 1.0

Alexander Reelsen

@spinscale

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

# Elasticsearch 1.0

- Aggregations
- Snapshot/Restore
- Distributed/scalable percolator
- Cat API
- ... and more

# Road to 1.0

- v0.4.0 - Feb 8, 2010
- v0.5.0 - Mar 5, 2010
- ...
- v0.19.0 - Mar 1, 2012
- v0.20.0 - Dec 7, 2012
- v0.90.0 - Apr 29, 2013
- v1.0 - Soon

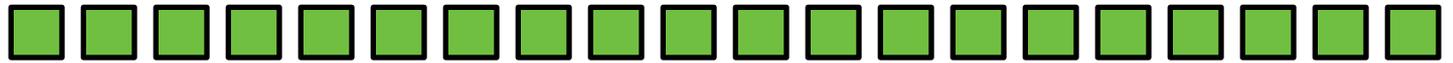
# Aggregations

# Aggregations

- Aggregation of information
- Facets are one dimensional  
Categories/brands/material of all results of this query
- Questions are multidimensional  
Average revenue per category id per day
- What is the average shopping cart size per order per hour?

# Aggregations

**Documents**



# Aggregations

Documents

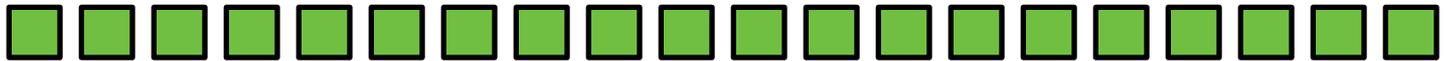


Query

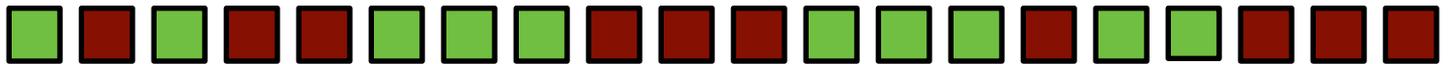


# Aggregations

Documents



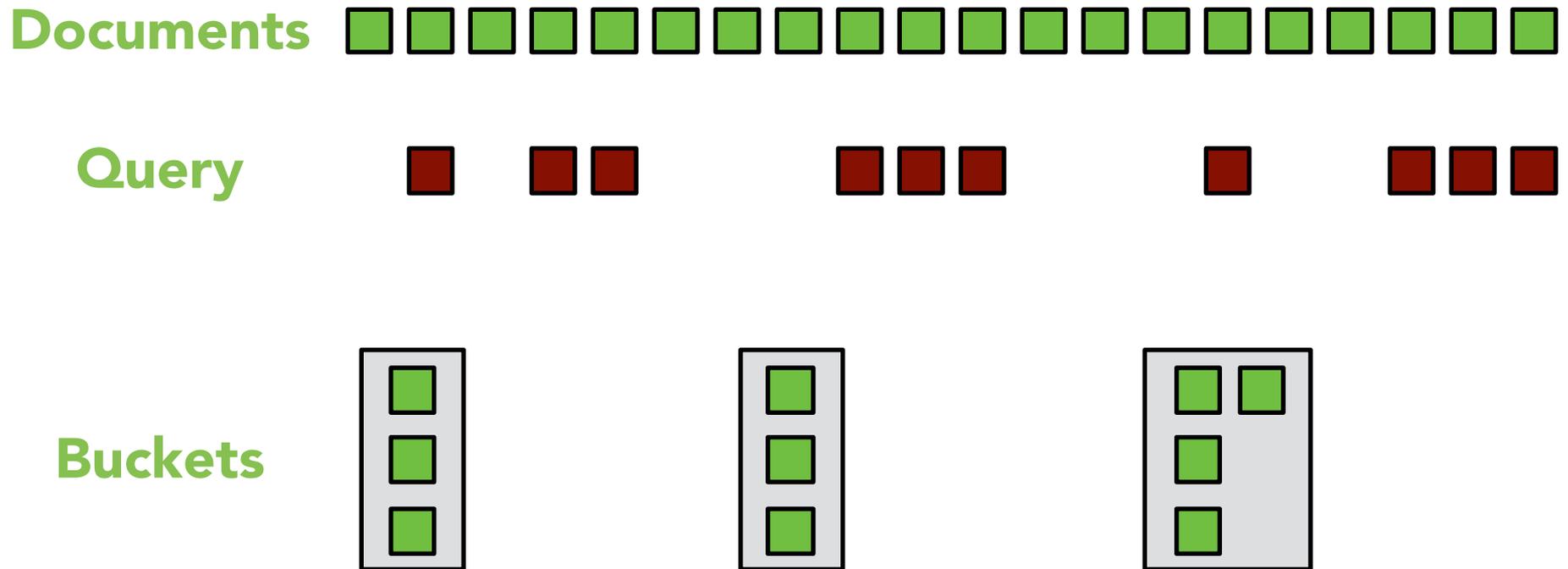
Query



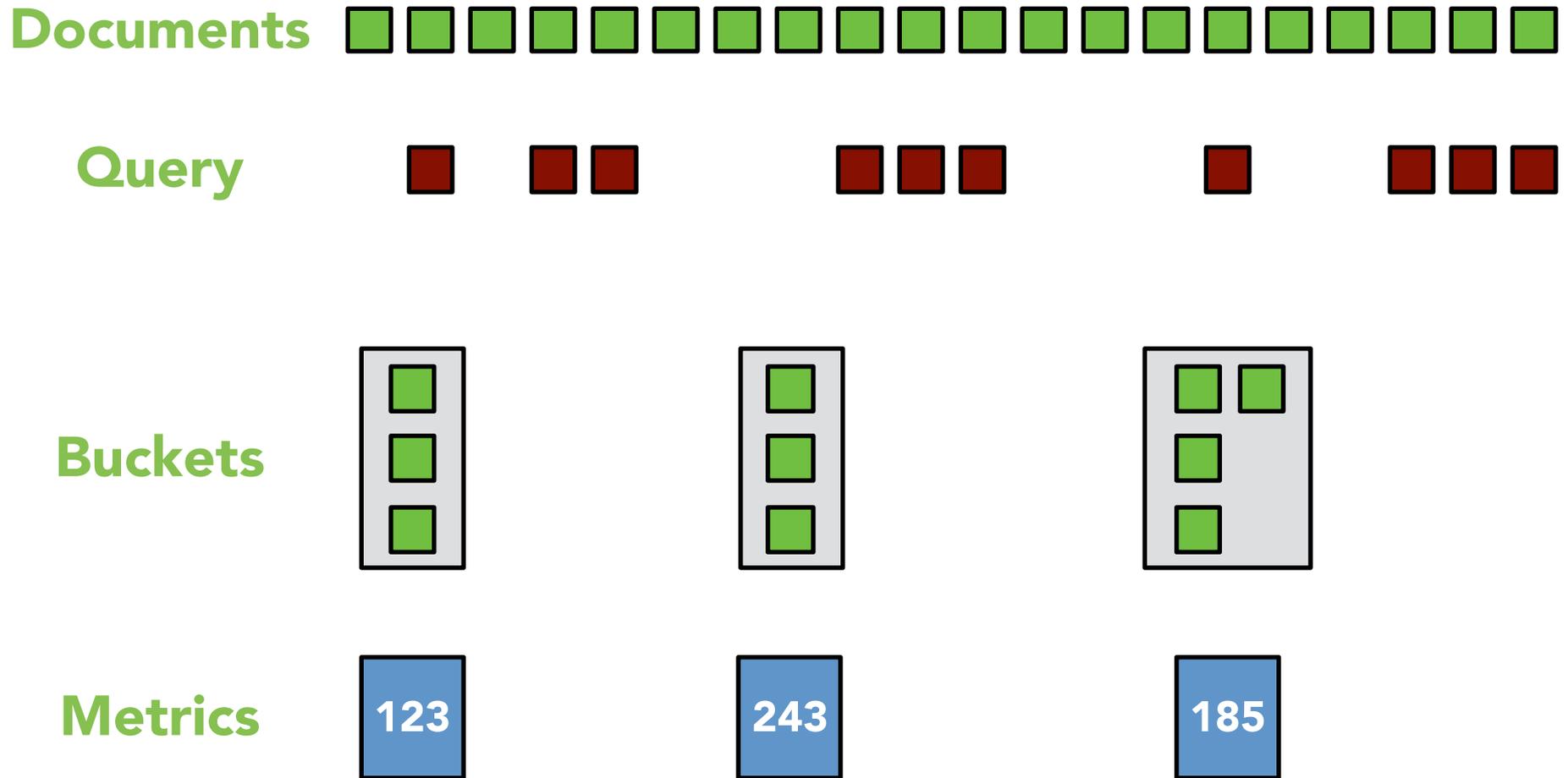
Buckets



# Aggregations



# Aggregations



# bucket aggregators

- global
- filter
- missing
- terms
- range
- date range
- ip range
- histogram
- date histogram
- geo distance
- nested

# metrics aggregators

- count
- stats
- extended stats
- avg
- max
- min
- sum

# Order average

```
» curl -XGET 'localhost:9200/orders/order/_search' -d '{
  "aggs": {
    "average_order_size" : {
      "avg" : { "field" : "total" }
    }
  }
}'
```

```
...
  "aggregations": {
    "average_order_size" : {
      "value" : 658.369
    }
  }
...

```

# Order average - filters

```
{
  "aggs": {
    "average_order_size_january": {
      "filter": {
        "range": { "created_at": { "gte": "2014-01-01", "lt": "2014-02-01" } } },
      "aggs": {
        "avg": { "field": "total" }
      }
    }
  }
}
```

```
...
  "aggregations": {
    "average_order_size_january": {
      "doc_count": 8,
      "value": 540.89754
    }
  }
...

```

# Order average - by day

```
{
  "aggs": {
    "by_day": {
      "filter": {
        "range": {
          "created_at": {
            "gte": "2014-01-01", "lt": "2014-02-01"
          }
        }
      },
      "aggs": {
        "daily_filter": {
          "date_histogram": {
            "field": "created_at",
            "interval": "day",
            "format": "yyyy-MM-dd"
          },
          "aggs": {
            "average_order_size": { "avg" : { "field" : "total" } }
          }
        }
      }
    }
  }
}
```

# Order average - by day

```
...
  "aggregations": {
    "by_day" : {
      "doc_count" : 32422,
      "daily_filter" : [ {
        "key_as_string" : "2014-01-01",
        "key" : 1388534400000
        "doc_count" : 423,
        "average_order_size" : {
          "value" : 380.0
        }
      }, {
        "key_as_string" : "2014-01-02",
        "key" : 1388534400000
        "doc_count" : 543,
        "average_order_size" : {
          "value" : 323.432
        }
      }, {
        ...
      ]
    }
  }
  ...
}
```

# Order average - by hour

```
{
  "aggs": {
    "by_day": {
      "filter": {
        "range": {
          "created_at": {
            "gte": "2014-01-01", "lt": "2014-02-01"
          }
        }
      },
      "aggs": {
        "hourly_filter": {
          "histogram": {
            "script": "doc[\u0027created_at\u0027].date.hourOfDay",
            "interval": 1
          },
          "aggs": {
            "average_order_size": { "avg" : { "field" : "total" } }
          }
        }
      }
    }
  }
}
```

# Order average - by hour

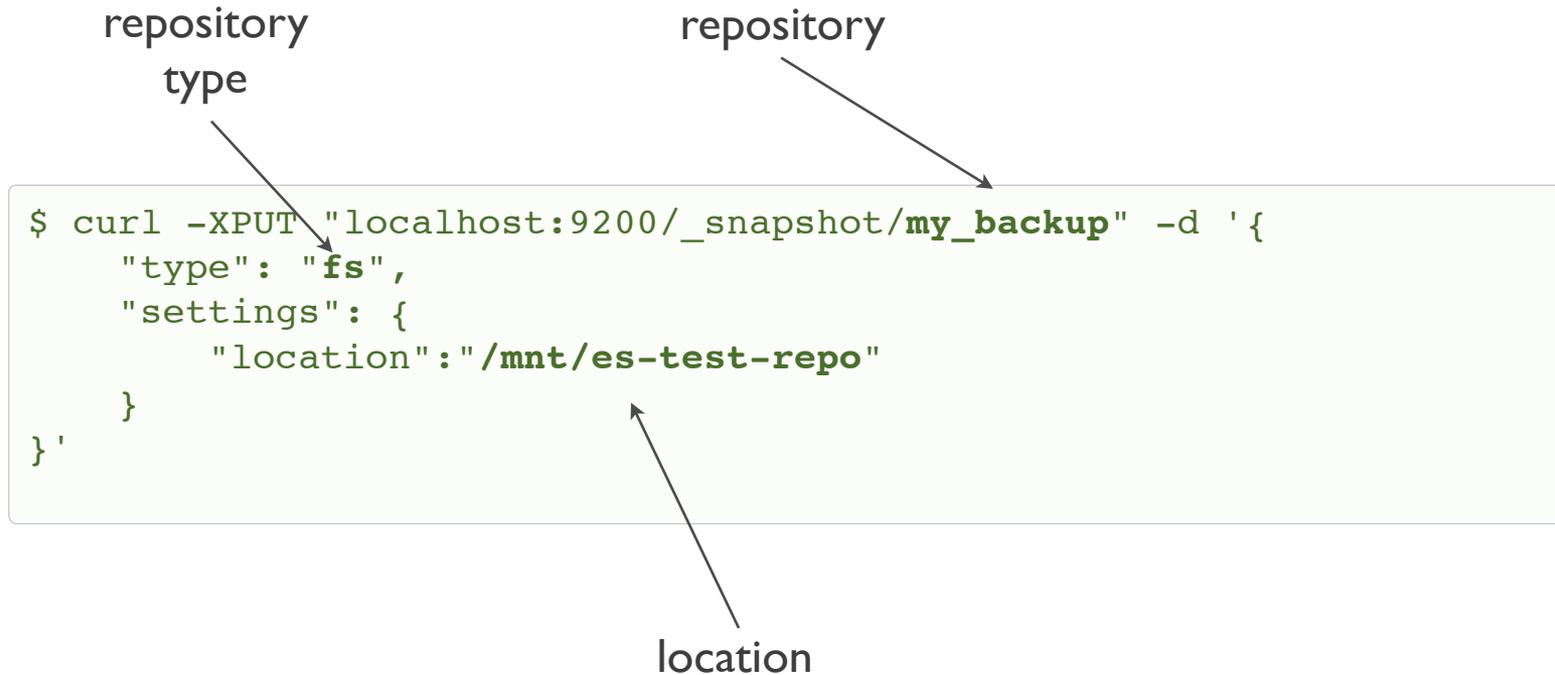
```
...
  "aggregations": {
    "by_day" : {
      "doc_count" : 32422,
      "daily_filter" : [ {
        "key" : "11",
        "doc_count" : 1534,
        "average_order_size" : {
          "value" : 380.0
        }
      }, {
        "key" : "18",
        "doc_count" : 8923,
        "average_order_size" : {
          "value" : 485.4323
        }
      }, {
        ...
      ]
    }
  }
  ...
}
```

# Snapshot/Restore

<http://www.elasticsearch.org/blog/introducing-snapshot-restore/>

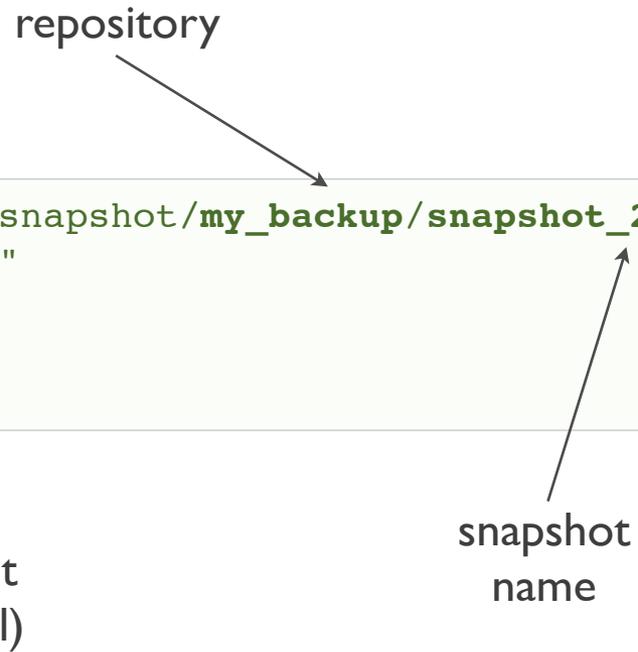
# Backup made easy

- Several shell commands + login were needed for pre 1.0 backups, but not via API



# Start snapshot

repository



```
$ curl -XPUT "localhost:9200/_snapshot/my_backup/snapshot_20131010" -d '{  
  "indices": "+test_*,-test_4"  
}'
```

index list  
(optional)

snapshot  
name

# Restore snapshot

close all indices  
that start with test\_

```
$ curl -XPOST "localhost:9200/test_*/_close"
```

repository  
name

snapshot  
name

```
$ curl -XPOST "localhost:9200/_snapshot/my_backup/snapshot_20131010" -d  
'{  
  "indices": "test_*"  
}'
```

index  
list

# Distributed & scalable Percolator

<http://www.elasticsearch.org/blog/percolator-redesign-blog-post/>

# percolator

- reverse search
- alerts
- updatable search results

# registering percolator in 0.90

target  
index

query id

```
$ curl -XPUT "localhost:9200/_percolator/tweeter/es-tweets" -d '{
  "query": {
    "match": { "text": "elasticsearch" }
  }
}'
```

# document percolation in 0.90

target  
index

percolation  
end point

```
$ curl -XGET "localhost:9200/twitter/tweet/_percolate" -d '{
  "doc": {
    "text": "#elasticsearch is awesome"
    "nick": "@imotov"
    "name": "Igor Motov"
    "date": "2013-11-03"
  }
}'
```

document  
to be percolated

```
{
  "ok": true
  "matches": ["es-tweets"]
}
```

matching  
queries

# how does it work in 0.90?

- all queries are stored in special `_percolate` index
- `_percolate` index has 1 primary shard which is replicated to every node
- each percolated document is indexed in memory
- all queries are executed against this document sequentially
- **execution time is linear to number of queries!**

# registering percolator in 1.0

reserved percolator  
type

query id

```
$ curl -XPUT "localhost:9200/some_index/.percolator/es-tweets" -d '{
  "query": {
    "match": { "body": "elasticsearch" }
  }
}'
```

any index with as  
many shards as you  
need

# multi index support

```
$ curl -XGET "localhost:9200/twitter,facebook/_percolate" -d '{
  "doc": {
    "body": "#elasticsearch is awesome"
    "nick": "@imotov"
    "name": "Igor Motov"
    "date": "2013-11-03"
  }
}'
```

document  
to be percolated



# other features

- percolation of existing document
- percolate count api
- filter support (in addition to queries in 0.90)
- highlighting, scoring
- multi-index, aliases support
- multi percolate (bulk percolation)

# Cat API

<http://www.elasticsearch.org/blog/introducing-cat-api/>

# Helping sysadmins

- Elasticsearch is full of monitoring APIs  
Everything is returned as JSON
- Humans are not the world's best JSON parsers
- What if elasticsearch had an easy to use interface from the commandline?

# Which one is the master?

```
$ curl "localhost:9200/_cluster/state?pretty&filter_metadata=true&filter_routing_table=true"
{
  "cluster_name" : "elasticsearch",
  "master_node" : "GNf0hEXlTfaBvQXKBF300A",
  "blocks" : { },
  "nodes" : {
    "ObdRqLHGQ6CMI5rOEstA5A" : {
      "name" : "Triton",
      "transport_address" : "inet[/10.0.1.11:9300]",
      "attributes" : { }
    },
    "4C7pKbfhTvu0slcSy_G4_w" : {
      "name" : "Kid Colt",
      "transport_address" : "inet[/10.0.1.12:9300]",
      "attributes" : { }
    },
    "GNf0hEXlTfaBvQXKBF300A" : {
      "name" : "Lang, Steven",
      "transport_address" : "inet[/10.0.1.13:9300]",
      "attributes" : { }
    }
  }
}
```

# Which one is the master? (v0.90)

```
$ curl "localhost:9200/_cluster/state?
pretty&filter_metadata=true&filter_routing_table=true"
{
  "cluster_name" : "elasticsearch",
  "master_node" : "GNf0hEXlTfaBvQXKBF300A",
  "blocks" : { },
  "nodes" : {
    "ObdRqLHGQ6CMI5rOEstA5A" : {
      "name" : "Triton",
      "transport_address" : "inet[/10.0.1.11:9300]",
      "attributes" : { }
    },
    "4C7pKbfhTvu0slcSy_G4_w" : {
      "name" : "Kid Colt",
      "transport_address" : "inet[/10.0.1.12:9300]",
      "attributes" : { }
    },
    "GNf0hEXlTfaBvQXKBF300A" : {
      "name" : "Lang, Steven",
      "transport_address" : "inet[/10.0.1.13:9300]",
      "attributes" : { }
    }
  }
}
```

# Which one is the master? (v1.0)

```
$ curl localhost:9200/_cat/master  
GNf0hEXlTfaBvQXKBF300A 10.0.1.13 Lang, Steven
```

# /cat/count

```
$ curl localhost:9200/_cat/count  
1383501234301 12:53:54 3344067
```

count



# `_cat/*` api

- `/_cat/allocation`
- `/_cat/count`
- `/_cat/health`
- `/_cat/master`
- `/_cat/aliases`
- `/_cat/nodes`
- `/_cat/recovery`
- `/_cat/shards`
- `/_cat/indices`
- `/_cat/thread_pool`

And more...

# And more...

- Disk-based fielddata

<http://www.elasticsearch.org/blog/disk-based-field-data-a-k-a-doc-values/>

- Fielddata circuit breaker
- Federated search

Thanks for listening

# Q & A

P.S. We're hiring  
<http://elasticsearch.com/about/jobs>  
<http://elasticsearch.com/support>

Alexander Reelsen  
@spinscale  
alexander.reelsen@elasticsearch.com