



Elastic Stack

Brief introduction and what's new

@medcl

About me

- Medcl, 曾勇 (Zeng Yong)
- Developer @ Elastic
 - Follow Elasticsearch since v0.5, 2010
 - Joined Elastic since September, 2015
 - Now in Beats team
- @medcl
- medcl@elastic.co
- <http://github.com/medcl>
- Based in Changsha, Hunan, China

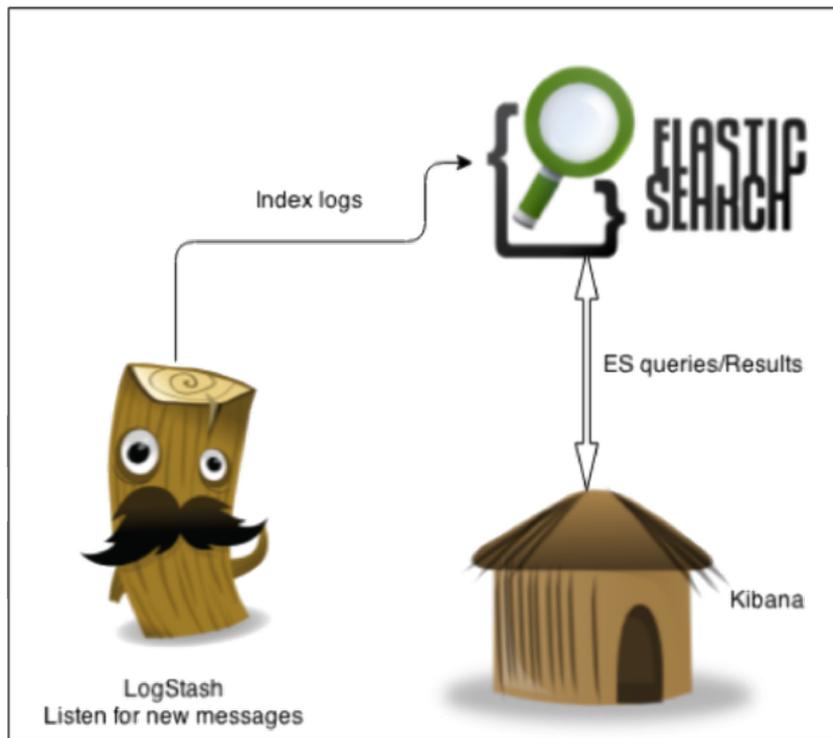


elastic

What's Elastic?

- A distributed startup company, since 2012
 - HQ: Mountain View, CA AND Amsterdam, Netherlands
 - With employees in 27 countries (and counting), spread across 18 time zones, speaking over 30 languages
- We are working on Open Source projects!
 - (Luckily some of them are popular, eg:elasticsearch)
- Offering support Subscription, X-pack, Cloud and Trainings
- Find us on: <https://github.com/elastic> and <https://www.elastic.co>

听说过“ELK”么？

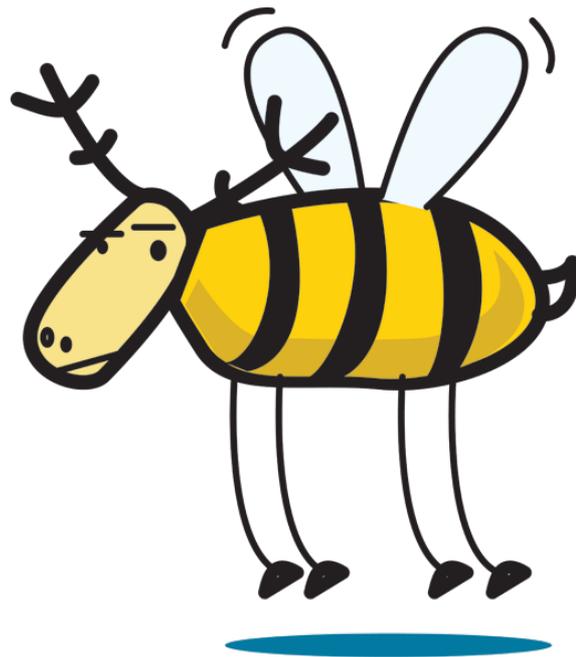


ELK is out!



Beats & Packetbeat

ELKB? BELK? LKBE? BKEL?



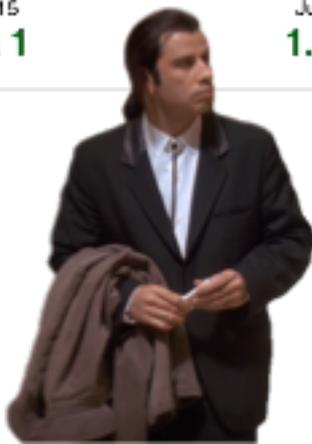
Logo



elasticsearch.



It's complicated



Release Bonanza

	Oct 28th	Nov 21st	Feb 2nd
es	2.0	2.1	2.2
kibana	4.2	4.3	4.4
ls	2.0	2.1	2.2
beats		1.0	1.1

It's time to unite!



The “Elastic Stack”

User
Interface



Store, Index,
& Analyze



Ingest



Extensions

Elastic Stack 能做什么？

Github: Enable Powerful Search For Both End-Users And Developers

GitHub Explore Features Enterprise Pricing Sign up Sign in

Search Search

Repositories 7,824
Code 744,618
Issues 51,842
Users 14

Languages

Java	1,308
JavaScript	987
Shell	804
Python	783
Ruby	778
PHP	397
Go	182
C#	168
Scala	159
HTML	92

We've found 7,824 repository results Sort: Best match

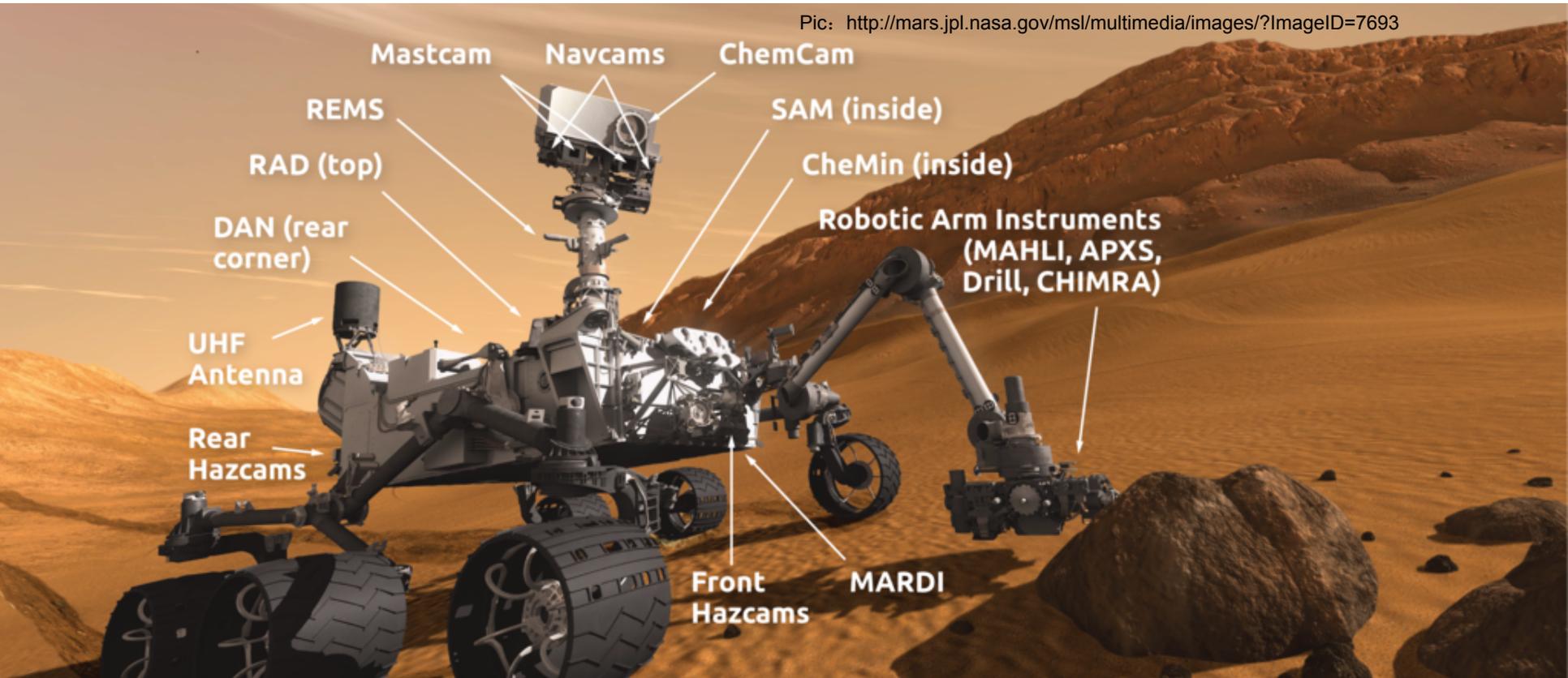
elastic/elasticsearch Java ★ 16,021 🍴 5,144
Open Source, Distributed, RESTful Search Engine
Updated 2 hours ago

dockerfile/elasticsearch ★ 306 🍴 272
ElasticSearch Dockerfile for trusted automated Docker builds.
Updated on Jan 8

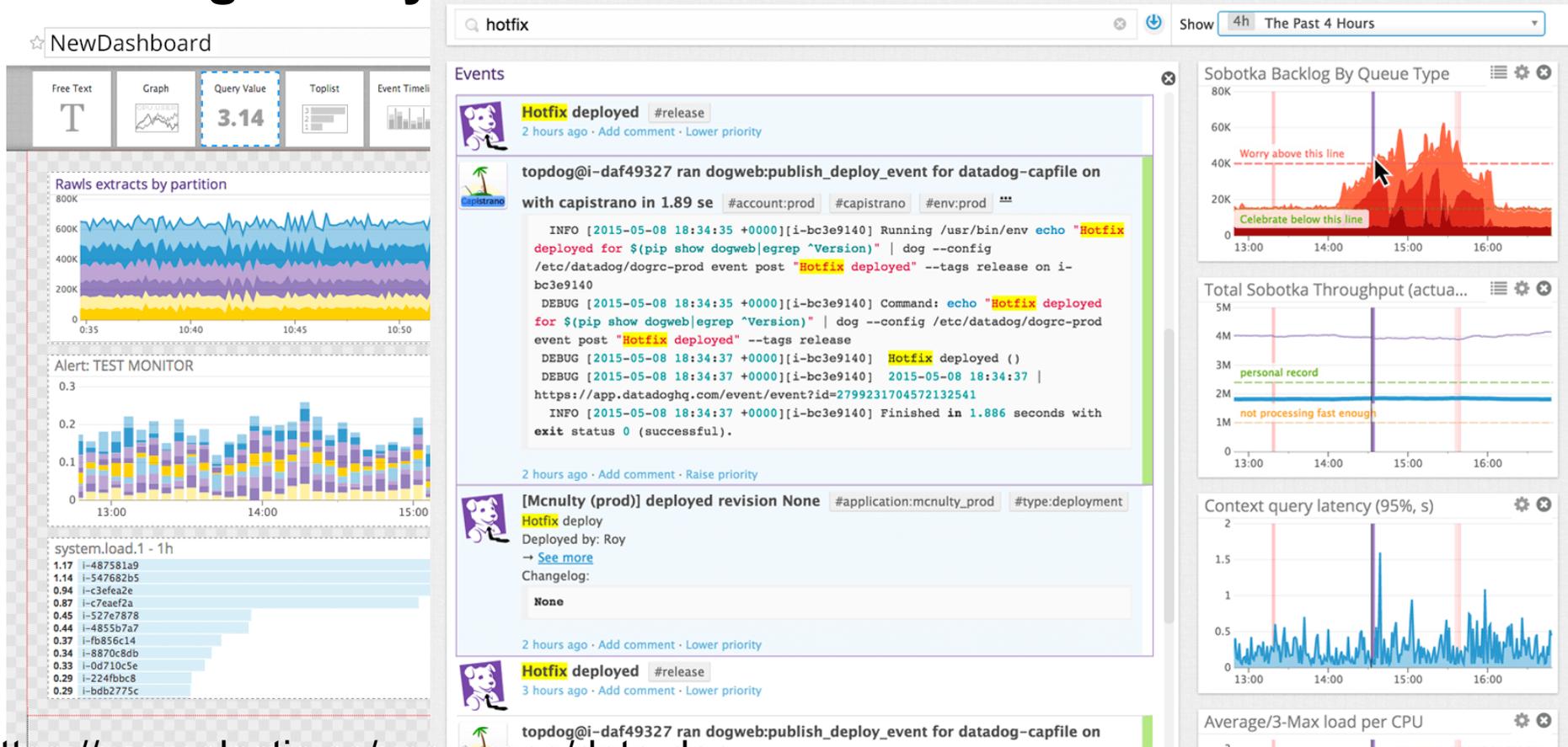
mesos/elasticsearch Java ★ 162 🍴 45
Elasticsearch on Mesos
Updated 23 hours ago

NASA: Unlocking Interplanetary Datasets with Real-Time Search

Pic: <http://mars.jpl.nasa.gov/msl/multimedia/images/?ImageID=7693>



Datadog: analysis metrics and time-series data



更多案例

<https://www.elastic.co/use-cases>

The “Elastic Stack”

User
Interface



Store, Index,
& Analyze



Ingest



Extensions

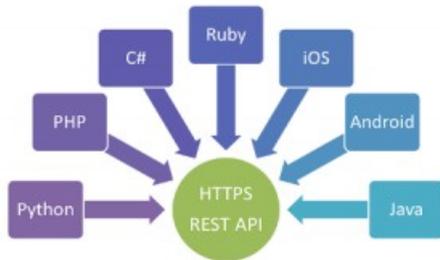
Elasticsearch is an open source, distributed, scalable, highly available, document-oriented, RESTful, full text search engine with real-time search and analytics capabilities

Thomson Reuters: “107 clusters ~1747 nodes” @Elastic{ON}16

<https://speakerdeck.com/elastic/thomson-reuters-research-journalism-finance-and-elastic>

Netflix: “~150 clusters totaling ~3,500 nodes hosting ~1.3 PB of data”

<http://techblog.netflix.com/2016/02/evolution-of-netflix-data-pipeline.html?m=1>

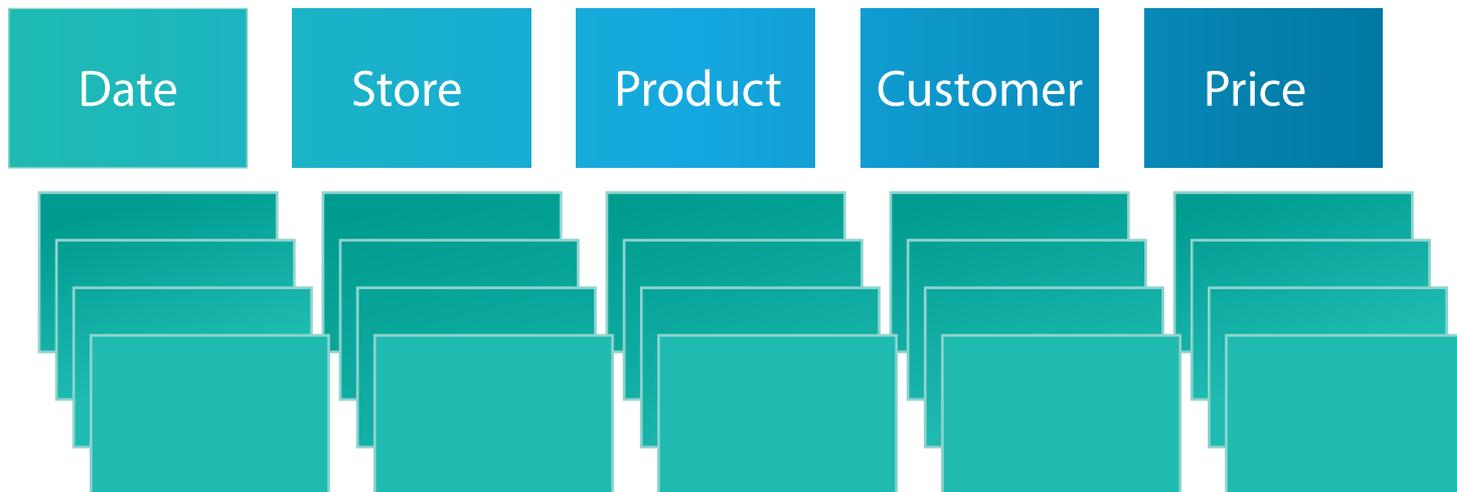


JSON

- Real-time analytics
- Time series data analytics
- Logging analytics
- Security analytics
- Fraud detection
- Prediction modeling
- Recommendations
- ...

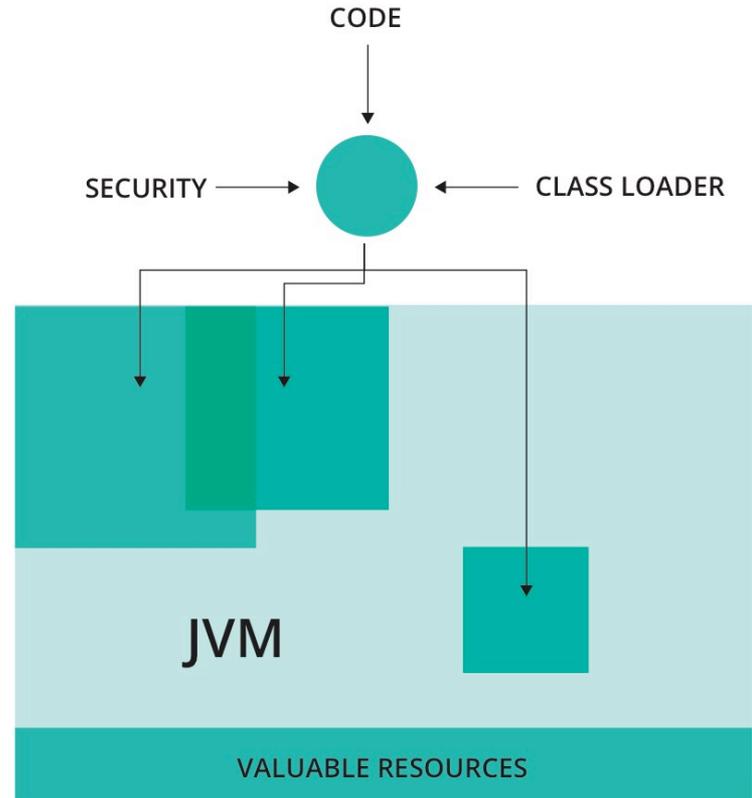
<http://github.com/elastic/elasticsearch>

Columnar Store



Java Security Manager

- One does not simply **fork a process**

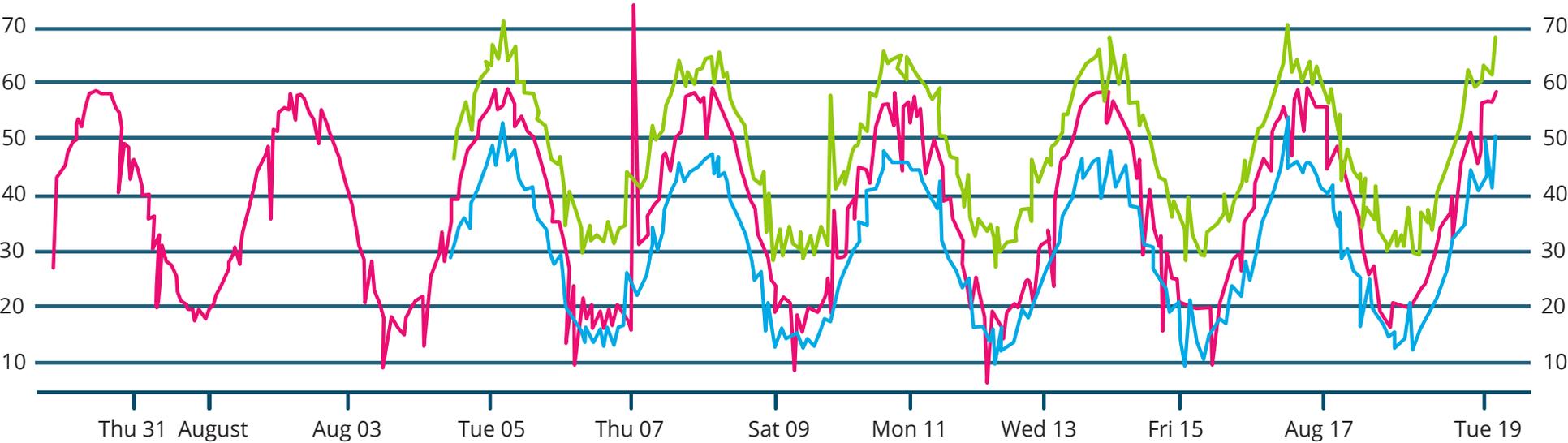


Cluster State Diffs

```
1. {  
2.   "nodes":  
3.   [  
4.     ["10,42.1.120",9200],  
5.     ["10,42.0.121",9200],  
6.     ["10,42.0.123",9200],  
7.     ["10,42.0.124",9200],  
8.     ["10,42.0.125",9200]  
9.   ]  
10. }
```

```
1. {  
2.   "nodes":  
3.   [  
4.     ["10,42.0.121",9200],  
5.     ["10,42.0.122",9200],  
6.     ["10,42.0.123",9200],  
7.     ["10,42.0.124",9200],  
8.     ["10,42.0.125",9200]  
9.   ]  
10. }
```

Pipeline Aggregations



Profile API

Original Query

```
{ "query": { "bool": { "must": [ { "match": { "journal": "biotech" } }, { "match": { "body": "dna" } } ], "should": [ { "match_phrase": { "body": "DNA extraction" } }, { "match_phrase": { "title": "DNA extraction" } } ] } } }
```

BooleanQuery 8.78ms | 100%

```
+ (+journal:biotech +body:dna body:"dna extraction" title:"dna extraction")  
#ConstantScore[_type:test]
```

BooleanQuery 3.567ms | 40.63%

```
+journal:biotech +body:dna body:"dna extraction" title:"dna extraction"
```

Slowest Branch

ConstantScoreQuery 2.261ms | 25.75%

```
ConstantScore[_type:test]
```

TermQuery 0.285ms | 3.25%

```
journal:biotech
```

TermQuery 0.406ms | 4.62%

```
body:dna
```

PhraseQuery 1.068ms | 12.16%

```
body:"dna extraction"
```

Phrase

PhraseQuery 0.01ms | 0.11%

```
title:"dna extraction"
```

Phrase

TermQuery 1.127ms | 12.84%

```
_type:test
```

Slowest Leaf

Painless Scripting

- Dynamic/ Static

```
def first = input.doc.first_name.0;  
def last  = input.doc.last_name.0;  
return first + " " + last;
```

```
String first = (String)((List)((Map)input.get("doc")).get("first_name")).get(0);  
String last  = (String)((List)((Map)input.get("doc")).get("last_name")).get(0);  
return first + " " + last;
```

it is ten times faster!

- Reindex API

- The Reindex API makes upgrading Elasticsearch easy
- Change problematic mappings & upgrade to the latest / greatest
- An important step towards 5.0 and there is a detailed blog post

- Task Management API

- Manage long running tasks in Elasticsearch
- A stepping stone towards future capabilities

What's more

- Plugincommand
 - bin/elasticsearch-plugin
- Lucene 6
 - DimensionalPoints/Multi-dimensionalpoints
 - numeric, date, and geospatial fields will be: 50% disk; 50% index time; 75% search time
- Ingest Node
 - grok, split, convert, and date etc.
- Text/Keyword to Replace Strings
- Instant aggregations
 - Date queries(aggregations) now cacheable
- Settings Validation
- Safety in production
- IndexName -> UUID
- Depreated logging

The “Elastic Stack”

User
Interface



Store, Index,
& Analyze

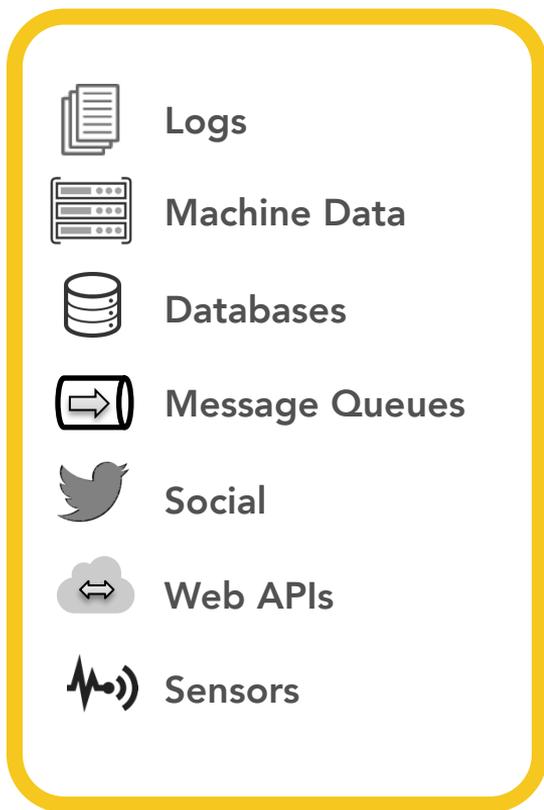


Ingest



Extensions

Logstash: Collect from diverse inputs

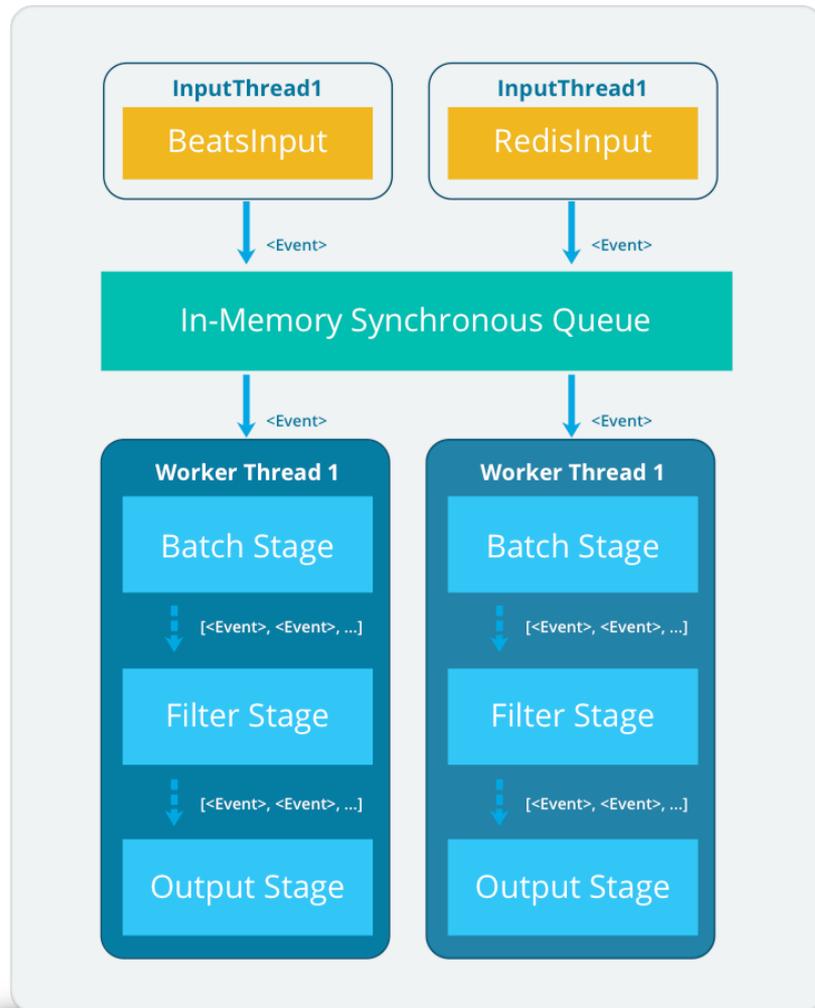


- Collects diverse sources
 - Logs + many others
 - Over 200 plugins
- Connects with live streams
 - Real-Time data
 - Wire / Transaction data
 - Full-Packet Network Capture

<http://github.com/elastic/logstash>

New pipeline Architecture

Faster, more reliable pipeline



Config Reload

```
1 input {
2   file {
3     path => "/Users/Elastic/Logstash/configs/webapp_logs/app.log"
4   }
5 }
6
7 filter {
8   if [metadata][logstash_plugin] == "true" {
9     mutate {
10      add_field => { plugin_type => "%{[metadata][plugin_type]}" }
11    }
12  }
13  else {
14    drop {}
15  }
16 }
17
18 output {
19   stdout { codec => rubydebug }
20
21   elasticsearch {
22     hosts => "localhost"
23     user => "es_admin"
24     password => "logstash+love"
25   }
26 }
```

Plugins



Log and metrics data



Sensor and device data



Web and social data



Data stores and streams

Kafka, HDFS,
Salesforce, HTTP,
Oh my!



Analysis
Elasticsearch + ANY data store

Alerting
Watcher + ANY notifications tools

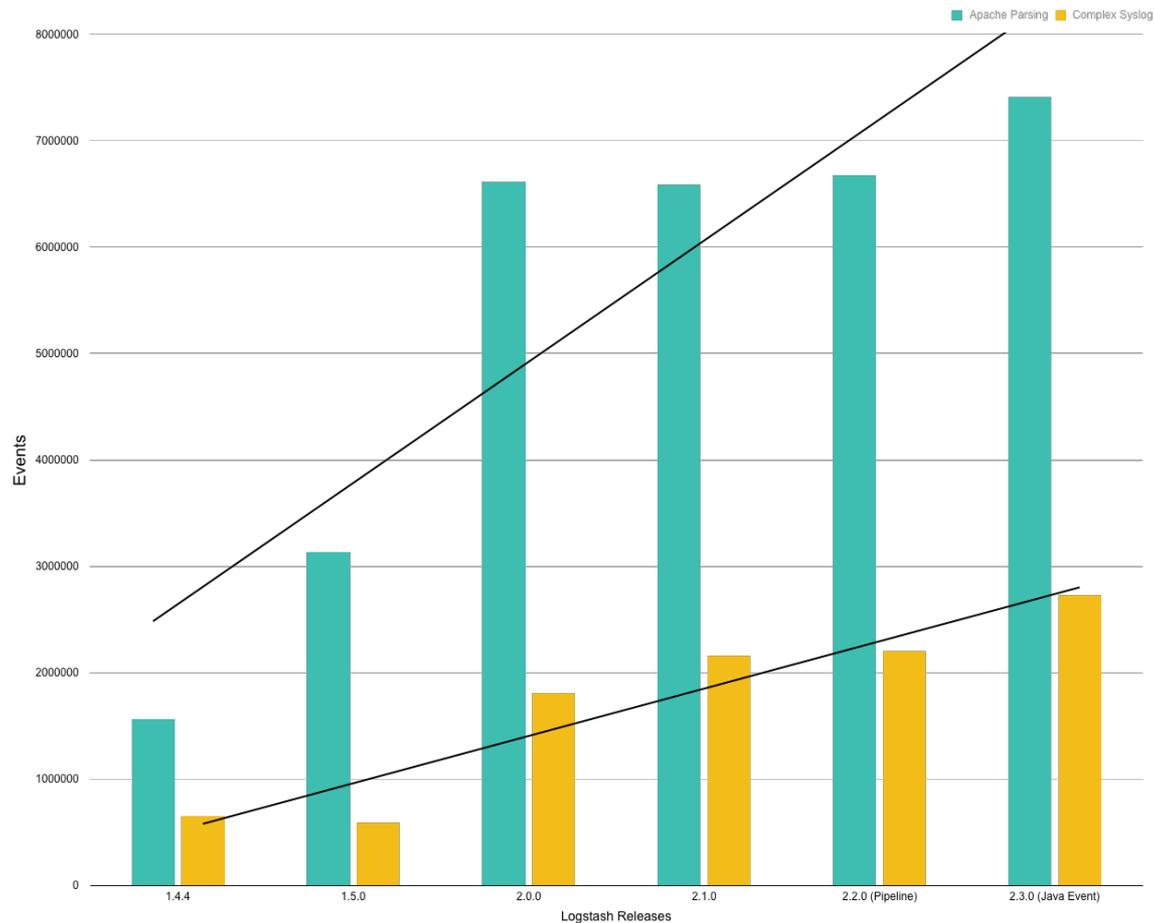


Monitoring
Marvel + ANY monitoring tools

Archiving
Hadoop + ANY cloud storage platform

Performance

Now you can
grok faster



Monitoring API

```
curl localhost
```

```
curl localhost:9600/_node/hot_threads?human
```

```
{  
  "events" :  
    "in" : 1  
    "filtere  
    "out" :  
  },  
  "jvm" : {  
    "timesta  
    "uptime_  
    "mem" :  
      "heap_  
      "heap_  
      ....
```

```
Hot threads at 2016-03-30T20:08:22-07:00, busiestThreads=3:  
5.22 % of of cpu usage by waiting thread named '[main]>worker3'  
  java.lang.Object.wait(Native Method)  
  java.lang.Object.wait(Object.java:460)  
  org.jruby.RubyThread$SleepTask.run(RubyThread.java:1050)  
  org.jruby.RubyThread.executeBlockingTask(RubyThread.java:1066)  
  org.jruby.RubyThread.wait_timeout(RubyThread.java:1414)  
  org.jruby.ext.thread.Queue.pop(Queue.java:152)  
  org.jruby.ext.thread.Queue.pop(Queue.java:127)  
  org.jruby.ext.thread.SizedQueue.pop(SizedQueue.java:111)  
  org.jruby.ext.thread.SizedQueue$INVOKER$i$pop.call(SizedQueue$INVOKER  
  org.jruby.runtime.callsite.CachingCallSite.call(CachingCallSite.java  
2.44 % of of cpu usage by timed_waiting thread named '[main]-pipeline-  
  java.lang.Object.wait(Native Method)  
  ....
```

What's more

- Plugin command
 - bin/logstash-plugin
- Kafka0.9 support
 - Support SSLencryptionandclientauth

The “Elastic Stack”

User
Interface



Store, Index,
& Analyze



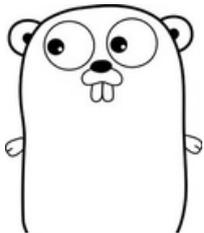
Ingest



Extensions

beats

- Beats are lightweight shippers that collect and ship all kinds of **operational data** to Elasticsearch
 - Small application
 - Install as agent on your servers
 - Written in **Golang**
 - No runtime dependencies
 - Single purpose

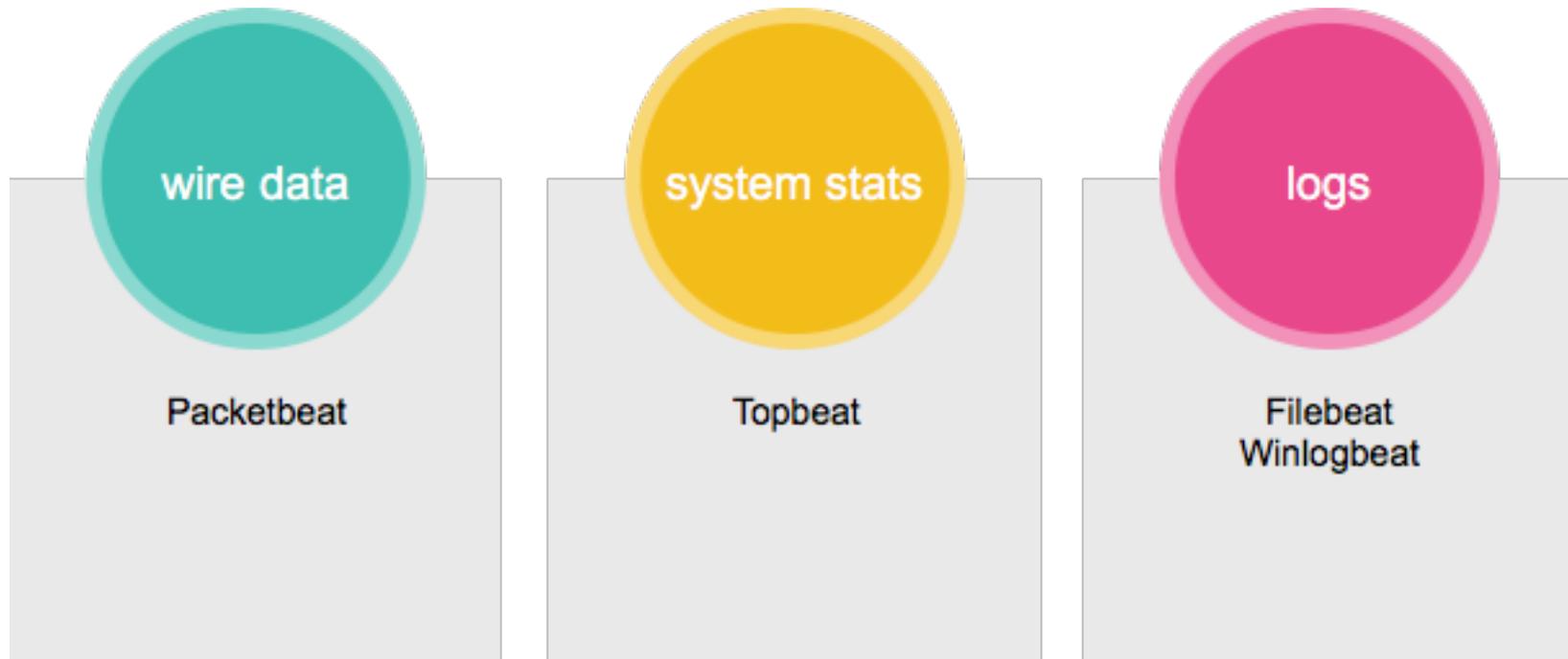


<http://github.com/elastic/beats>



<https://www.flickr.com/photos/8barbikes/17256970434/>

Examples of operational data



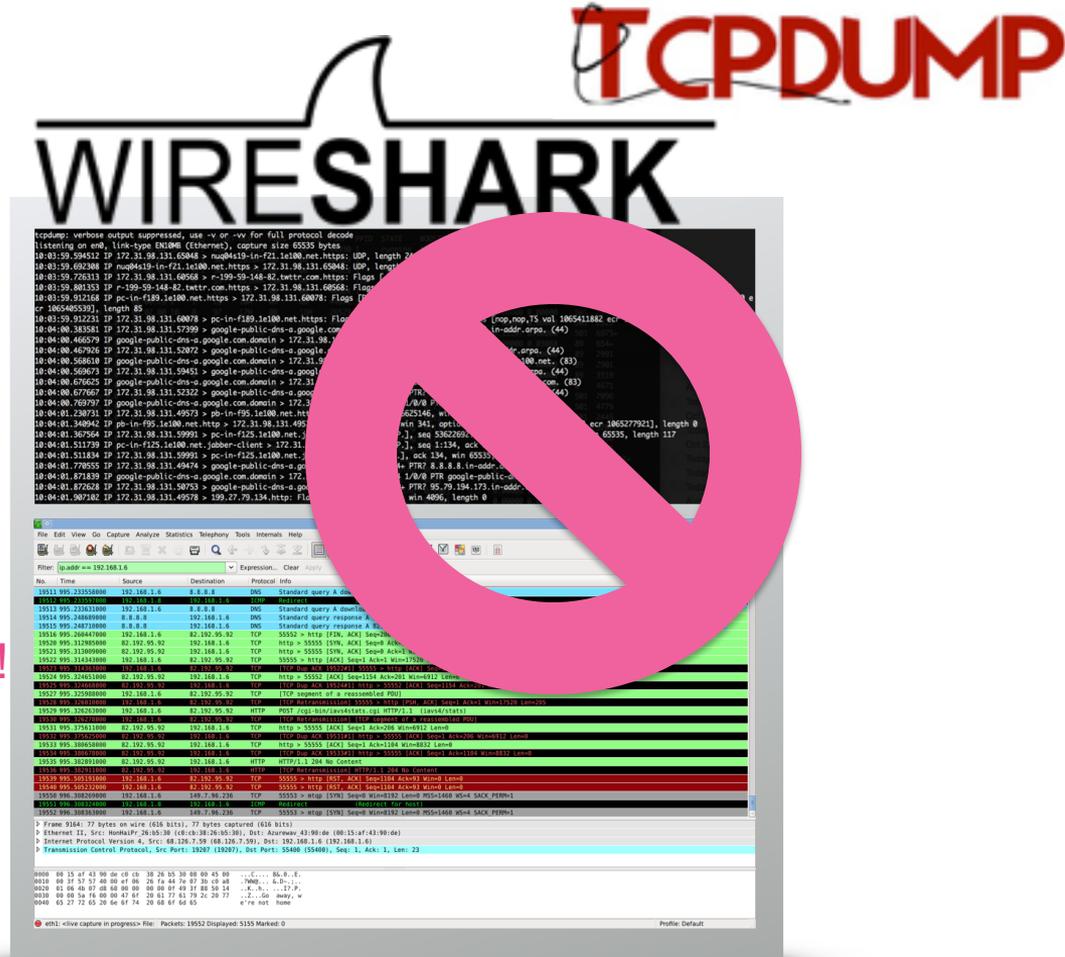
Packetbeat

Sniffs the traffic between your servers, parses the **application-level** protocols on the fly.

Built-in protocols:

- HTTP
- MySQL
- PostgreSQL
- Redis
- Thrift-RPC
- MongoDB
- DNS
- Memcache

• ...



Let's go realtime!

winlogbeat!

System Number of events: 728

Level	Date and Time	Source	Event...	Task C
Information	1/13/2015 9:26:35 AM	Service Control Manager	7036	None
Information	1/13/2015 9:26:35 AM	Service Control Manager	7036	None
Information	1/13/2015 9:26:35 AM	Service Control Manager	7036	None
Information	1/13/2015 9:26:35 AM	Service Control Manager	7036	None
Information	1/13/2015 9:26:35 AM	Service Control Manager	7036	None
Information	1/13/2015 9:26:33 AM	Ntfs (Microsoft-Windows-N...	98	None
Information	1/13/2015 9:26:33 AM	Kernel-Processor-Power (Mi...	55	(47)
Information	1/13/2015 9:26:33 AM	Kernel-Processor-Power (Mi...	55	(47)
Information	1/13/2015 9:26:32 AM	Kernel-Power	508	(159)
Information	1/13/2015 9:26:32 AM	FilterManager	6	None

Event 7036, Service Control Manager

General Details

The Plug and Play service entered the running state.

Log Name: System
Source: Service Control Manager
Event ID: 7036
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online Help](#)

Logged: 1/13/2015 9:26:35 AM
Task Category: None
Keywords: Classic
Computer: vagrant-2012-r2

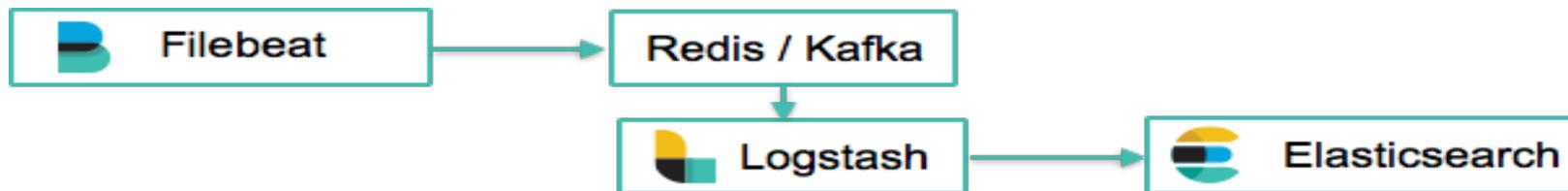
Forwards Windows Event logs to Elasticsearch

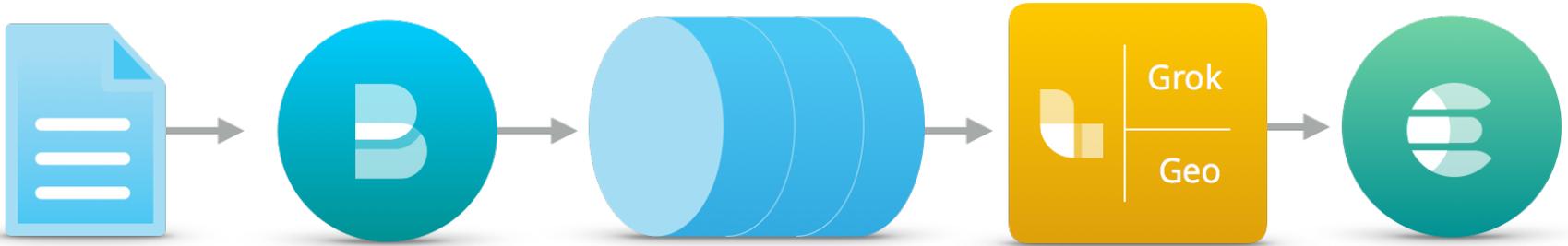
Filebeat

A more lightweight log shipper

- **Generic filtering**

Flexibly reduce the amount of data sent of the wire and stored





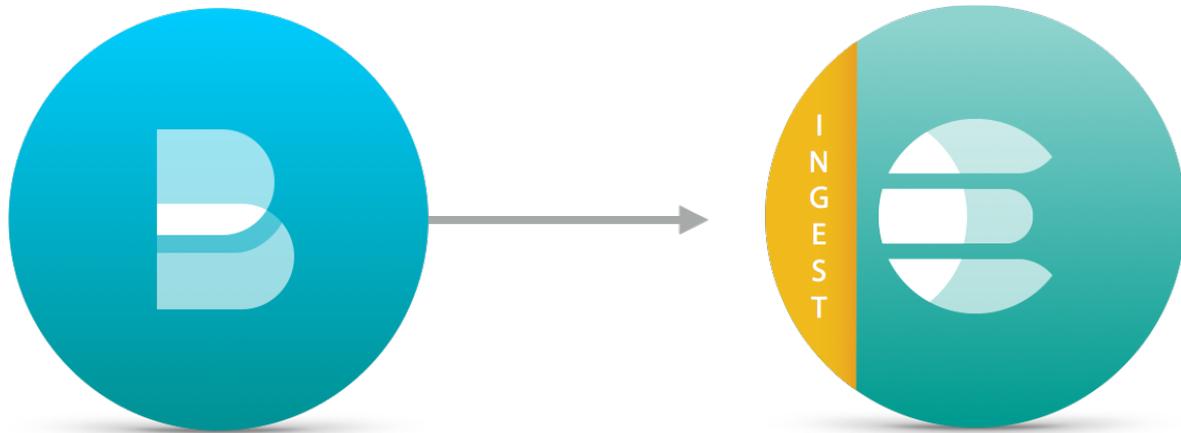
Simple things should be simple

*Not
like this*



Like this





```

Processes: 367 total, 3 running, 6 stuck, 358 sleeping, 1833 threads
Load Avg: 2.79, 2.69, 2.66 CPU usage: 35.77% user, 8.25% sys, 55.96% idle
SharedLibs: 161M resident, 22M data, 10M linkedit. MemRegions: 132196 total, 5764M resident, 90M private
PhysMem: 13G used (3762M wired), 2581M unused.
VM: 2084G vszize, 527M framework vszize, 56231732(0) swappins, 59291827(0) swapouts.
Networks: packets: 66492613/486 in, 57364574/316 out. Disks: 6744547/309G read, 34720568/883G written.

```

PID	COMMAND	%CPU	TIME	#TH	#WQ	#PORTS	MEM	PURG	OPRS	PGRP	PPID	STATE	BOOSTS
64667	burn	89.3	04:00.32	5/1	0	15	580K	00	00	64667	63666	running	*0[1]
29185	java	21.0	92:05.82	74	0	186	369M	00	65M	29185	98883	sleeping	*0[2]
15112	topbeat	18.7	02:56:57	12	0	61	9000K	00	1100K	15111	15111	sleeping	*0[1]
325	iTerm	12.8	02:18:11	12	4	366	90M	40968	40M	325	1	sleeping	*0[19744]
65237	top	7.4	00:08.34	1/1	0	24	7012K	00	00	65237	18734	running	*0[1]
0	kernel_task	4.6	23:39:40	228/4	0	2	1715M+	00	00	0	0	running	*0[0]
61	mds	3.3	01:48:47	10	6	329+	14M	00	34M	61	1	stuck	*0[1]
194	mds_stores	3.1	02:03:27	10	7	76	20M+	1184K	41M	194	1	stuck	*0[1]
186	WindowServer	1.8	14:39:27	5	2	1224	79M	5324K	267M	186	1	sleeping	*0[1]
637-	Dropbox	1.5	93:04.38	84	0	385	80M	00	44M	637	1	sleeping	*0[53781]
97	hidd	1.4	93:44.84	6	2	98-	3300K-	00	1400K	97	1	sleeping	*0[1]
53155	mdworker	1.1	01:05.39	4	0	66	14M	00	1268K	53155	1	sleeping	*0[1]
53158	mdworker	1.0	01:06.41	4	0	62	12M	00	1000K	53158	1	sleeping	*0[1]
3431	Slack	0.9	03:36:09	19	1	456	611M-	27M	331M	3431	1	sleeping	*0[22918]
75966	python2.7	0.7	28:17.99	3	1	33	8240K	00	13M	75966	75959	stuck	*0[1]
53159	mdworker	0.7	01:04.20	4	0	62	17M	00	1144K	53159	1	sleeping	*0[1]
53157	mdworker	0.6	01:06.09	4	0	62	15M	00	948K	53157	1	sleeping	*0[1]
324-	zoom.us	0.5	13:07:07	13	0	44995	124M	00	179M	324	1	sleeping	*48[18]
23794-	dbusseventsd	0.4	01:45.43	1	0	7	4168K	00	148K	637	23793	sleeping	*0[1]
75965	python2.7	0.4	22:28.90	2	0	15	6860K	00	11M	75965	75959	sleeping	*0[1]
58424	Google Chrom	0.3	01:23.17	12	0	111	117M-	00	37M	316	316	sleeping	*0[2]
46	fsseventsd	0.3	30:30.91	13	0	306	4796K	00	4104K	46	1	sleeping	*0[1]
23795-	dbusseventsd	0.2	01:07.97	1	0	7	32K	00	152K	637	23794	sleeping	*0[1]
48135	Google Chrom	0.2	06:38.73	15	0	60	43M	00	18M	316	316	sleeping	*0[1]
65300	screencaptur	0.2	00:00.14	2	0	52	2220K	20K	00	336	336	sleeping	*0[1]
89	mDNSResponde	0.2	46:50.49	7	2	88	2096K	00	1124K	89	1	sleeping	*0[1]
316	Google Chrom	0.1	20:08:22	48	1	1998	943M	736K	1245M	316	1	sleeping	*0[10133]
39271	inkscape-bin	0.1	03:05.45	7	0	39	7844K	00	168M	39268	39268	sleeping	*0[1]
60240	Google Chrom	0.1	01:33.59	9	0	110	39M+	00	34M	316	316	sleeping	*0[2]
212	symptomsd	0.1	16:32.42	4	2	182	2184K	00	2112K	212	1	sleeping	*0[66876]
23793-	dbusseventsd	0.0	00:23.07	1	0	12	36K	00	180K	637	637	sleeping	*0[1]
61533	Google Chrom	0.0	00:07.55	13	0	123	90M	00	00	316	316	sleeping	*0[2]
16578	Google Chrome	0.0	06:12.31	13	0	155	150M+	00	50M	316	316	sleeping	*0[2]
639-	Google Photo	0.0	15:39.98	14	0	262	4648K	00	34M	639	1	sleeping	*0[27266]
13660	Finder	0.0	16:11.28	9	2	423	115M	00	239M	13660	1	sleeping	*0[2627]
58571	com.apple.op	0.0	00:13.14	4	1	263	20M	512K	19M	58571	1	sleeping	*0[297]
54793	Google Chrome	0.0	00:18.73	13	0	121	80M	00	16M	316	316	sleeping	*0[2]
54832	com.apple.op	0.0	01:02.49	4	1	263	9248K	00	26M	54832	1	sleeping	*0[1856]
62861	Google Chrome	0.0	00:01.85	12	0	112	76M	00	00	316	316	sleeping	*0[2]
3391	com.apple.op	0.0	02:43.68	4	1	256	4992K	00	16M	3391	1	sleeping	*0[13618]

Topbeat

Like the Unix **top** command but sends the output periodically to Elasticsearch. Also works on Windows.

System wide

- system load
- total CPU usage

...

Per process

- state
- name
- command line

...

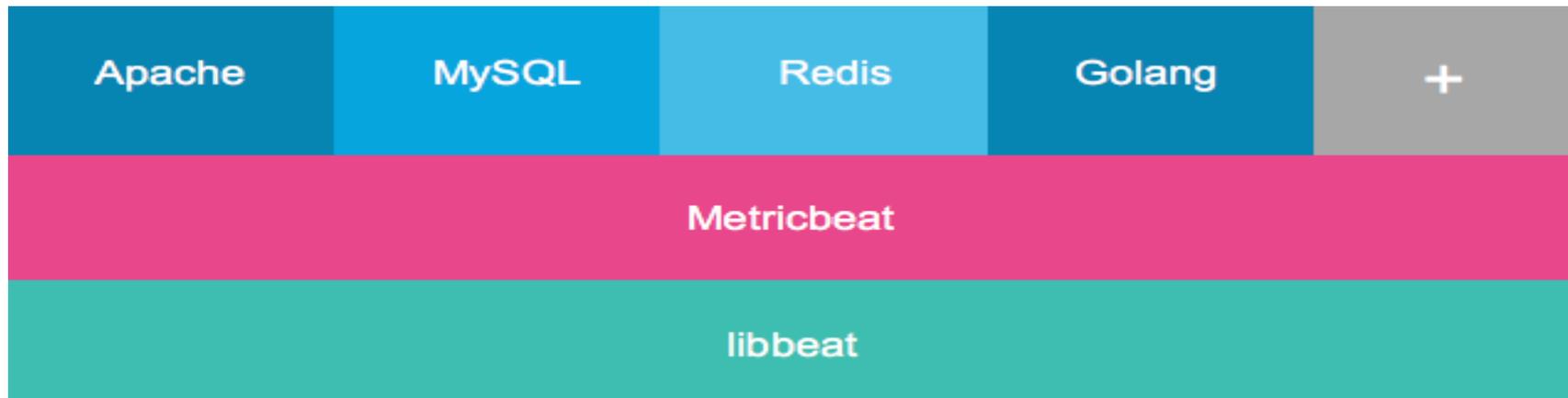
Disk usage

- available disks
- used, free space ...

That's More!

Metricbeat: Connecting Numb3rs

- Listens to the internal “beat” of systems via APIs.



<http://github.com/elastic/beat-generator/>

Custom Fields and generic filtering

- Filtering the exported data

```
filter:
  - include_fields:
    fields:
      - bytes_in
      - bytes_out
      - ip
      - client_ip
      - dns.question.name
      - dns.question.etld_plus_one
      - dns.response_code
```

What's more

- Decode JSON from log lines
- Kafka output
 - Output to Kafka directly
- Integration with IngestNode
 - set“pipelineparameter” in the Elasticsearch output config
- Support IP/TCP flows
 - Report statistics like packet/byte counts

The “Elastic Stack”

User
Interface



Store, Index,
& Analyze



Ingest



Extensions

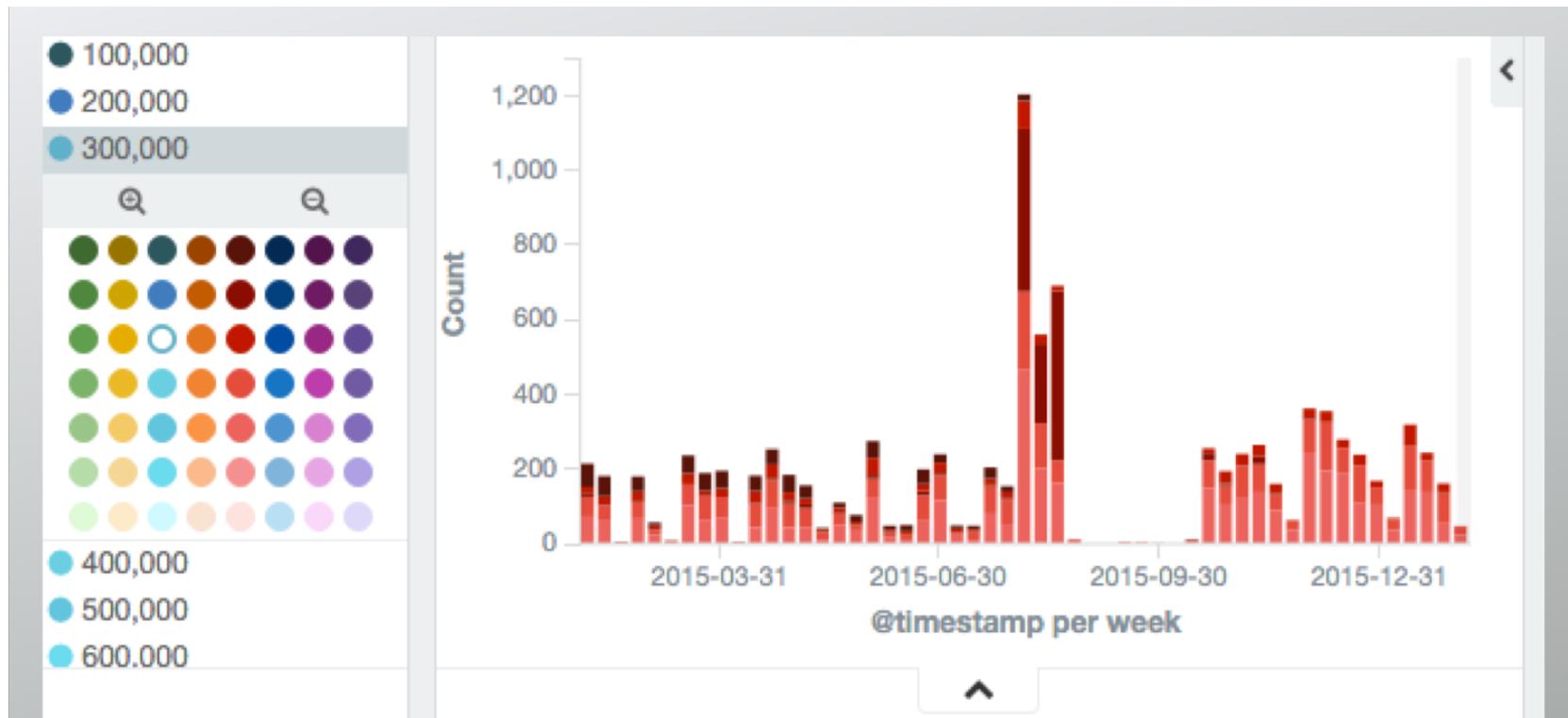
What's Kibana?

Kibana is an open source analytics and **visualization platform** designed to work with Elasticsearch.

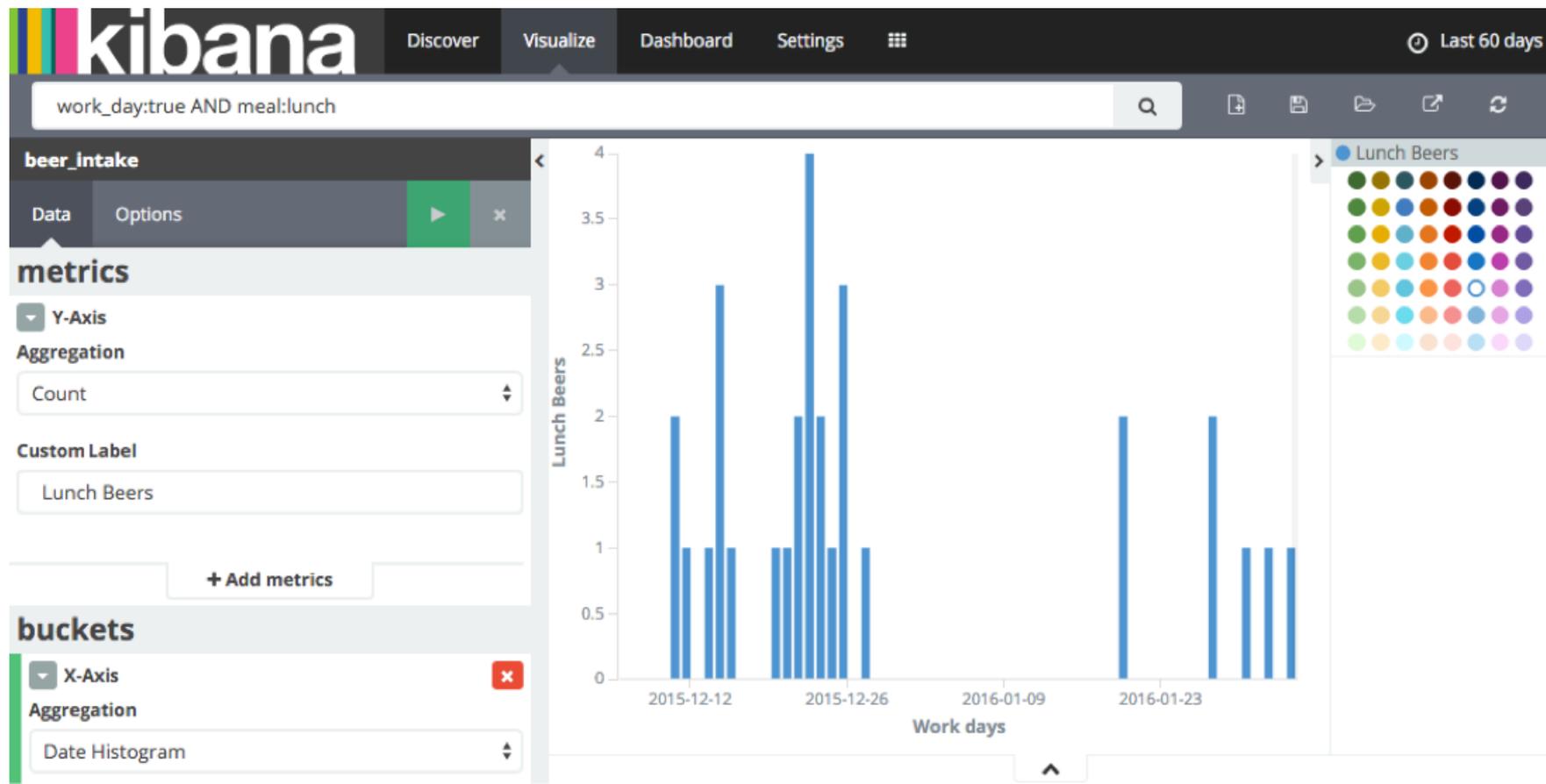


<http://github.com/elastic/kibana>

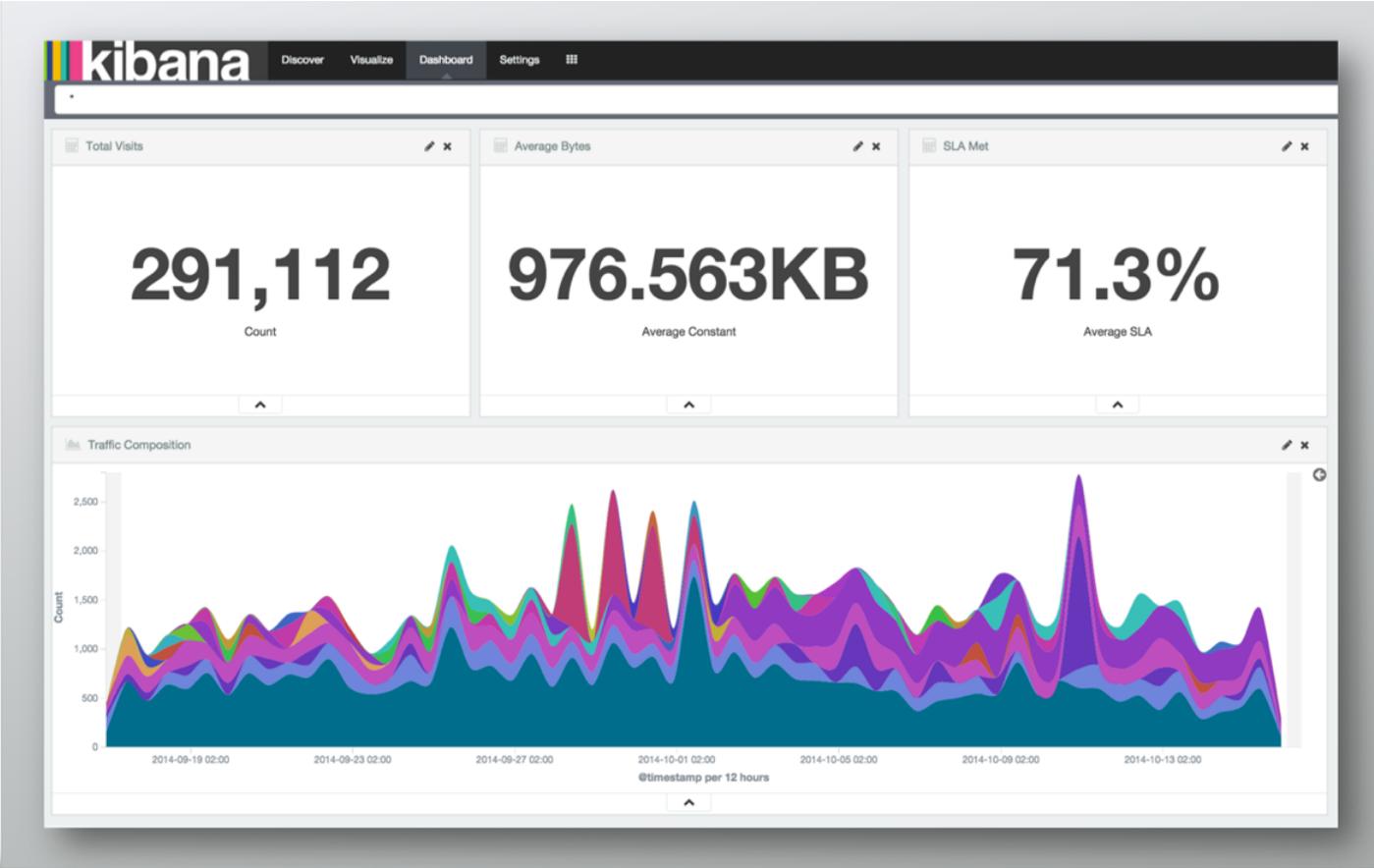
Colour picker



Custom Legends



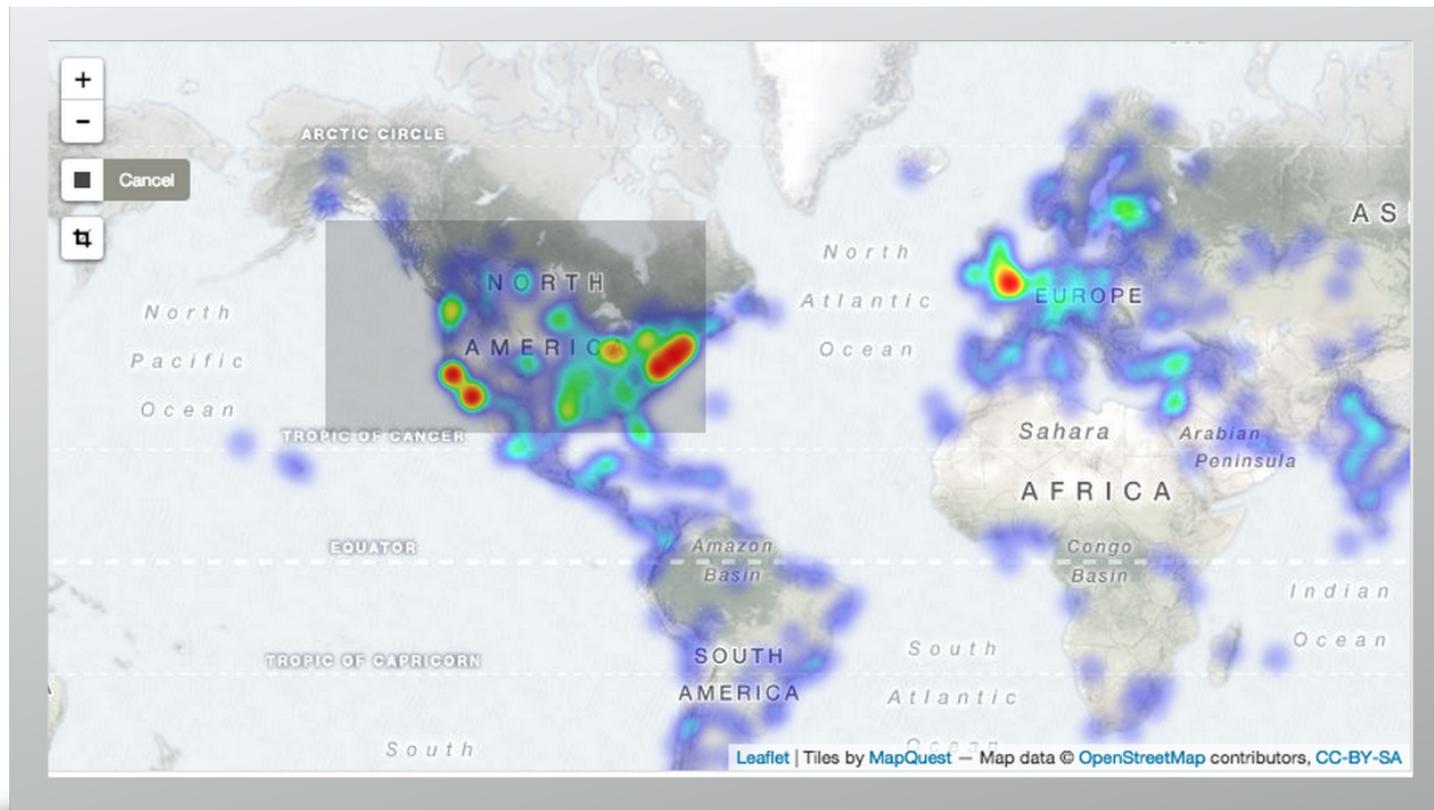
Field formatters



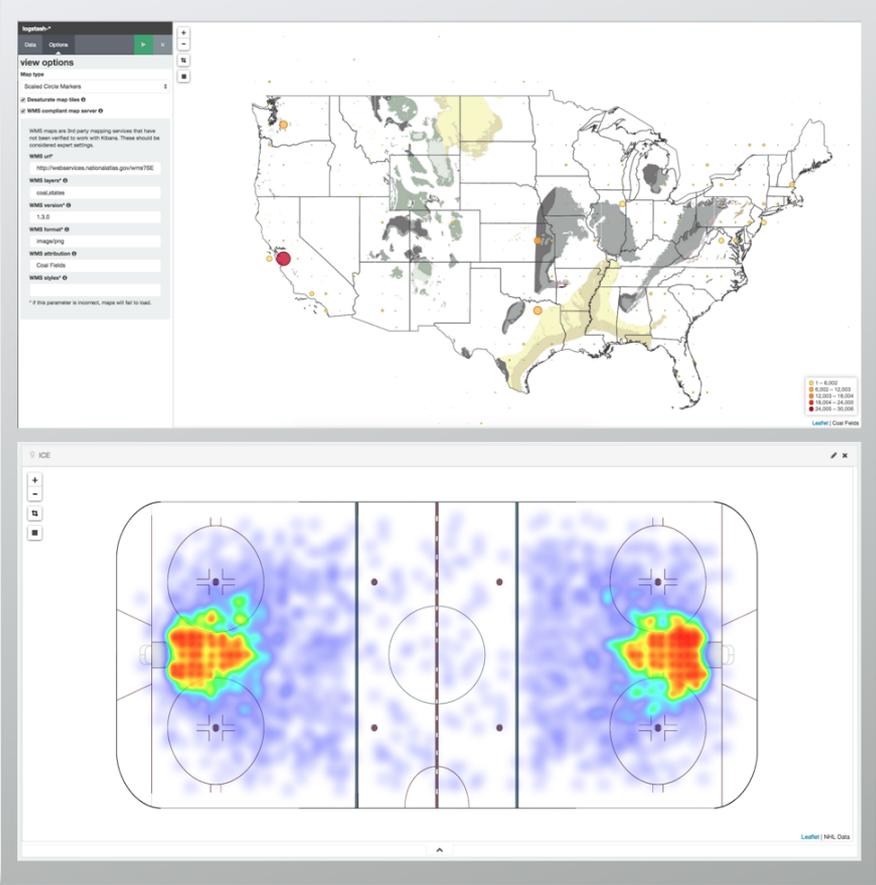
Black theme



Heat map



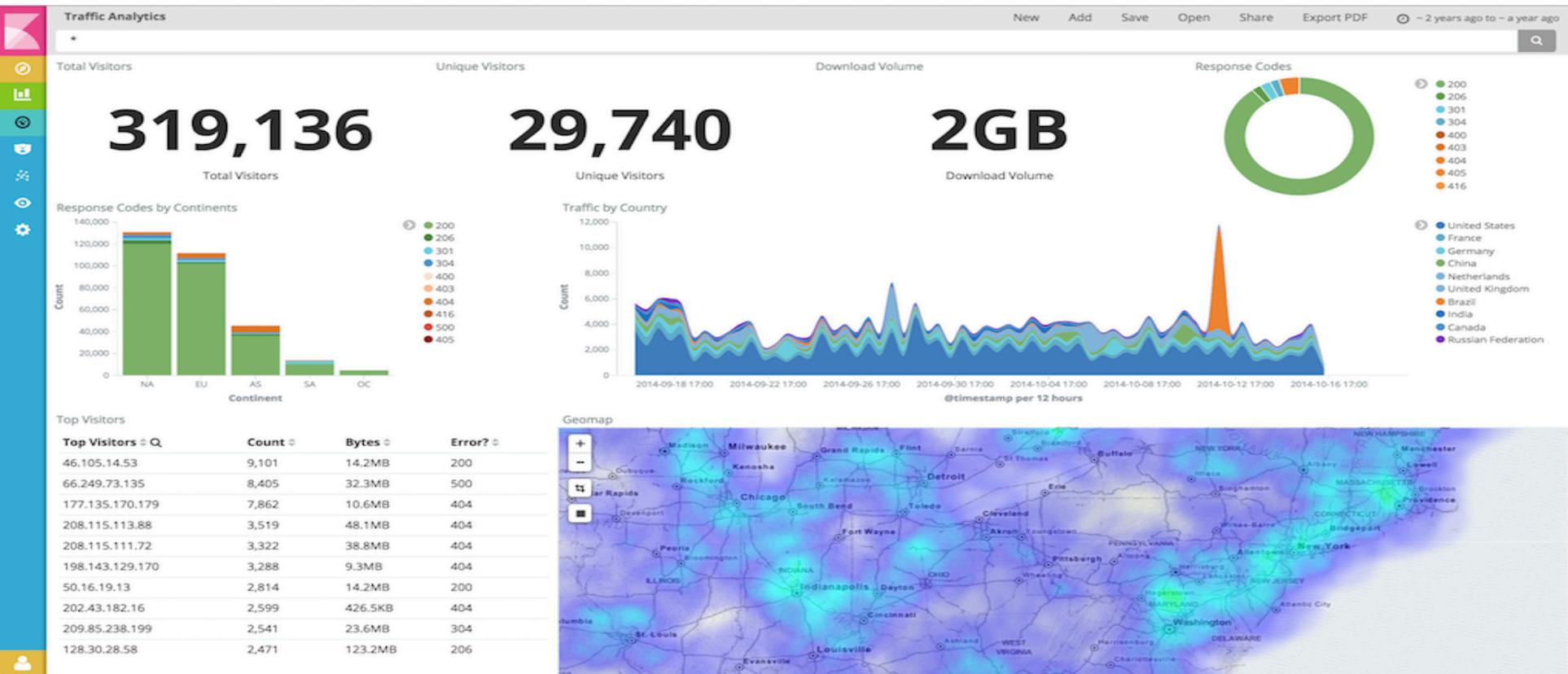
Pluggable Tile Servers



Global timezone



Brings a new focus on your data



Application Framework

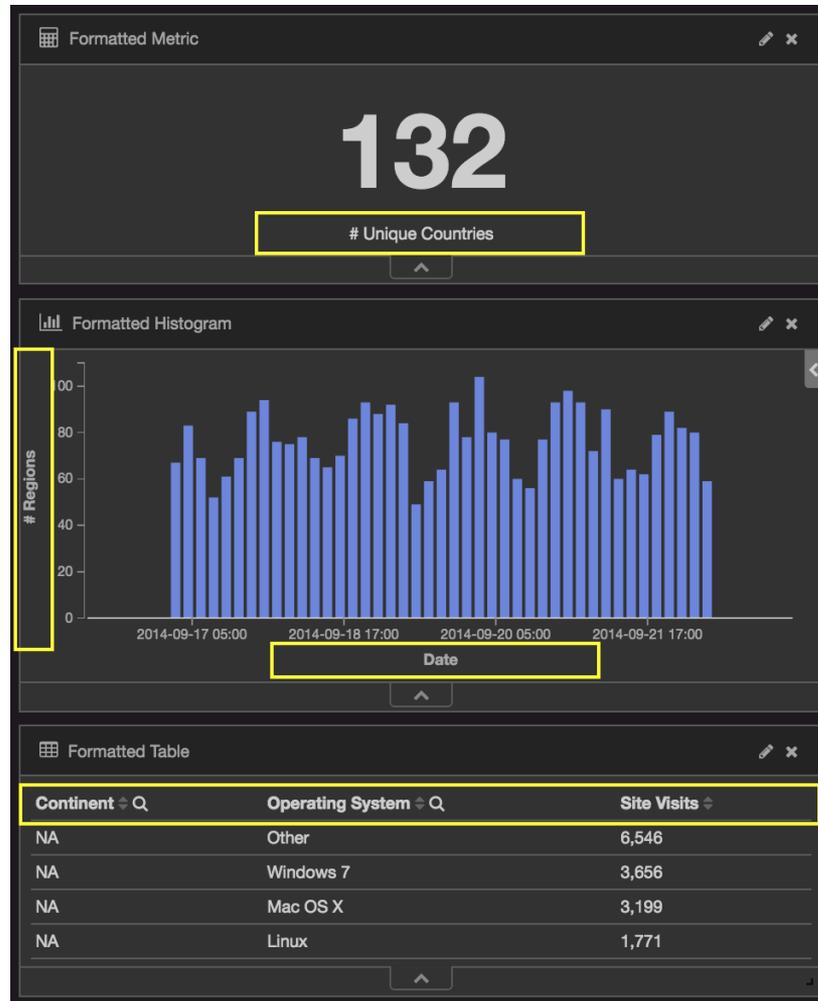
Appear in the main navigation



System-Generated Labels

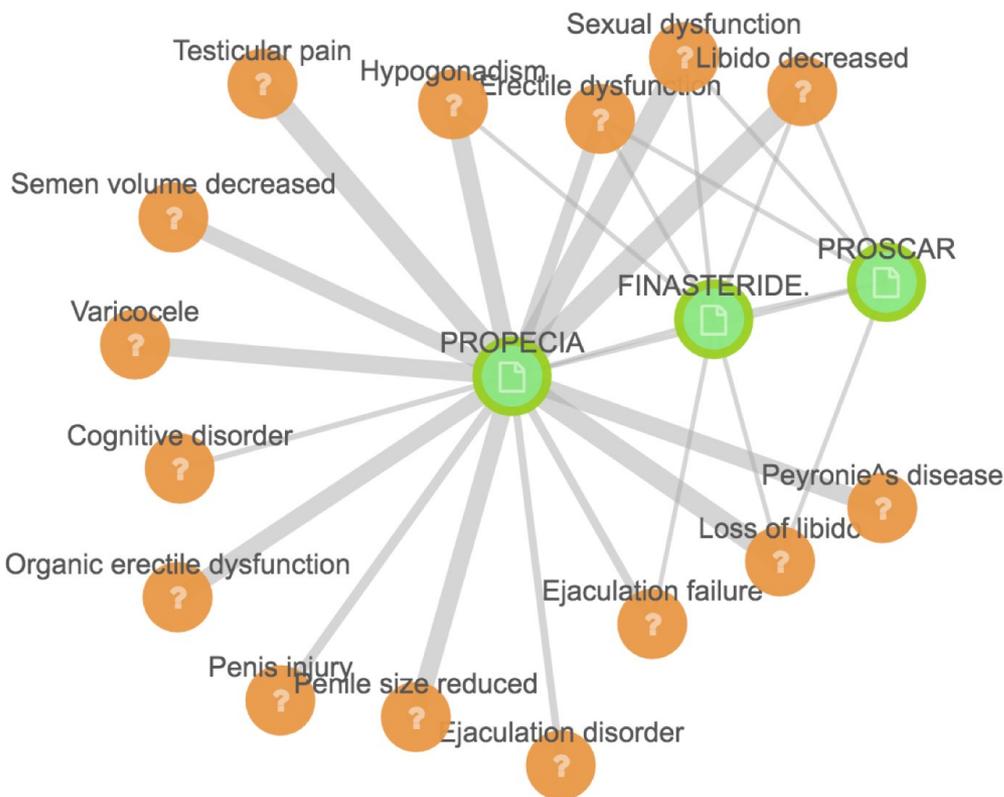


Custom Labels



What's more

- Plugin command
 - bin/kibana-plugin
- “Sense ”willbe“ Console” (not yet available)
 - Sense plugin will be built into Kibana
- Graph



Selections

all none invert linked

- PROPECIA
- FINASTERIDE.
- PROSCAR

Link summary

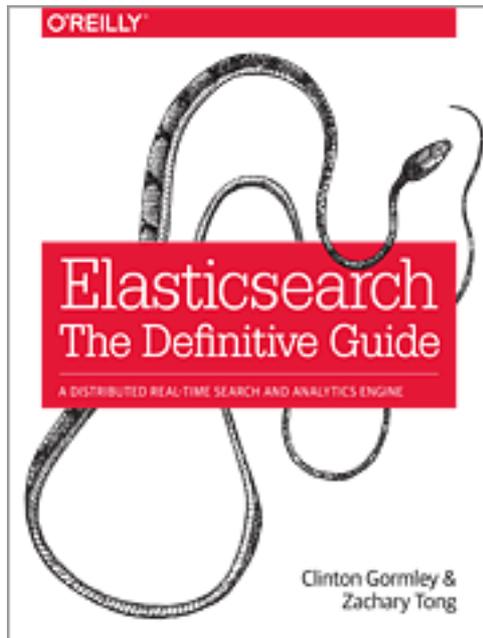
Penile size reduced PROPECIA

23 (17) 299

Community

- 源码 & Issue: <http://github.com/elastic/>
- 中文社区: <http://elasticsearch.cn>
- 官方 QQ 群: 190605846

ES权威指南翻译中，欢迎志愿者加入！
<https://github.com/elasticsearch-cn/elasticsearch-definitive-guide>



Thanks!