



大数据平台架构

# 基于 **Elastic Stack** 的数据探索与分析

@medcl

# QCon

2016.10.20~22

上海·宝华万豪酒店

## 全球软件开发大会 2016

### [上海站]



购票热线: 010-64738142

会务咨询: [qcon@cn.infoq.com](mailto:qcon@cn.infoq.com)

赞助咨询: [sponsor@cn.infoq.com](mailto:sponsor@cn.infoq.com)

议题提交: [speakers@cn.infoq.com](mailto:speakers@cn.infoq.com)

在线咨询 (QQ): 1173834688

团 · 购 · 享 · 受 · 更 · 多 · 优 · 惠

# 7折

优惠 (截至06月21日)  
现在报名, 立省2040元/张

# About me

- 曾勇 (Medcl)
- 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



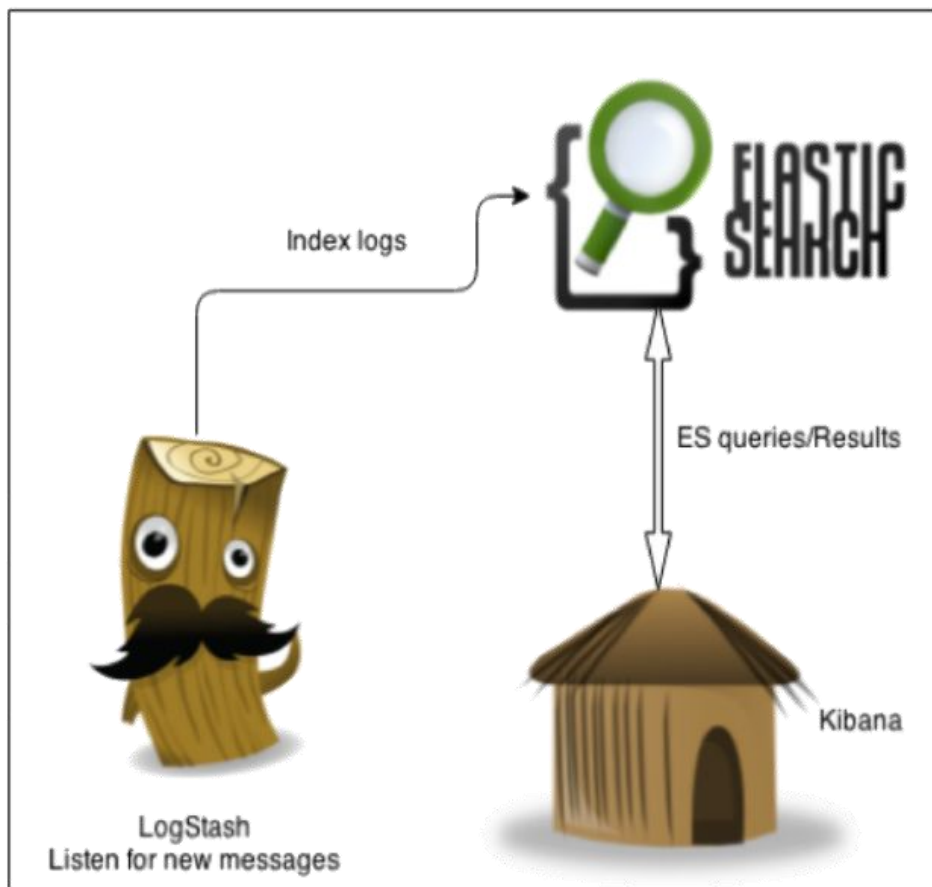
# elastic

# 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？

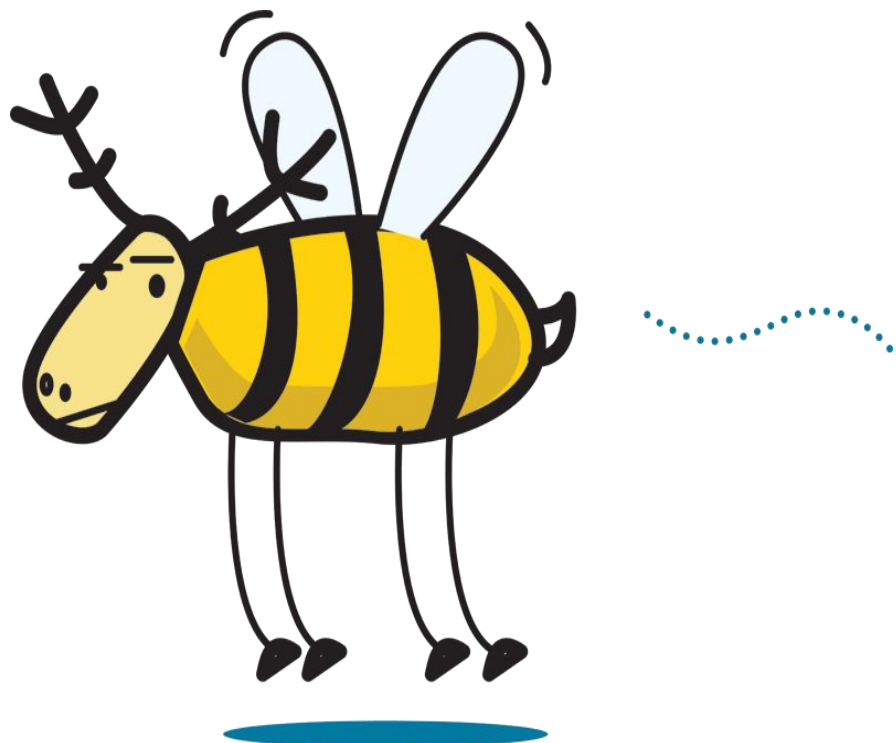


# But ELK is out!



Beats & Packetbeat

## ELKB? BELK? LKBE? BKEL?



# Logo



elasticsearch.



# Release Bonanza

|        | Oct 28th   | Nov 21st   | Feb 2nd    |
|--------|------------|------------|------------|
| es     | <b>2.0</b> | <b>2.1</b> | <b>2.2</b> |
| kibana | <b>4.2</b> | <b>4.3</b> | <b>4.4</b> |
| ls     | <b>2.0</b> | <b>2.1</b> | <b>2.2</b> |
| beats  |            | <b>1.0</b> | <b>1.1</b> |
|        |            |            |            |

**It's time to unite!**

# The “Elastic Stack”, stay together from v 5.0

User  
Interface



kibana

Store, Index,  
& Analyze



elasticsearch

Ingest



logstash



beats



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

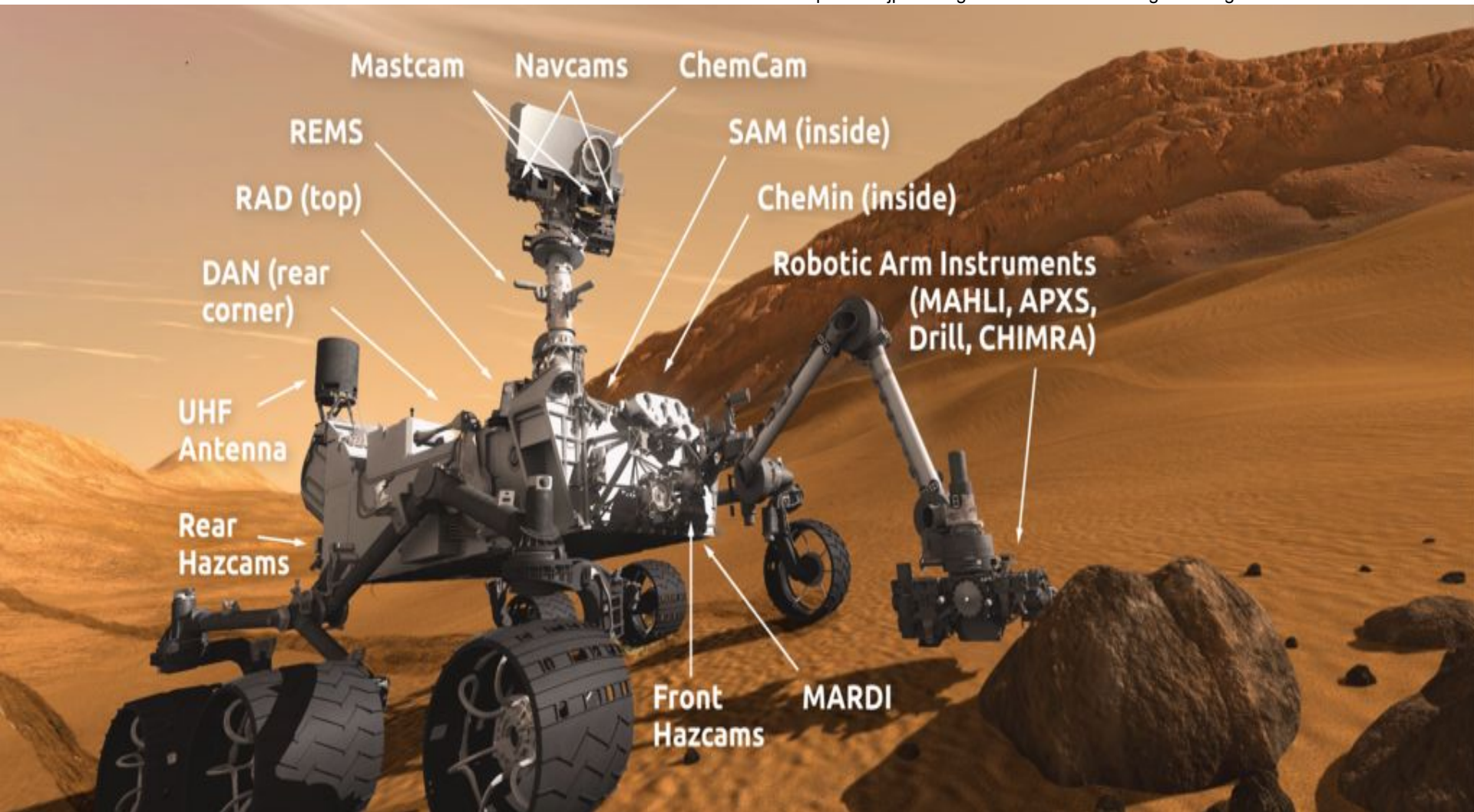
12

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



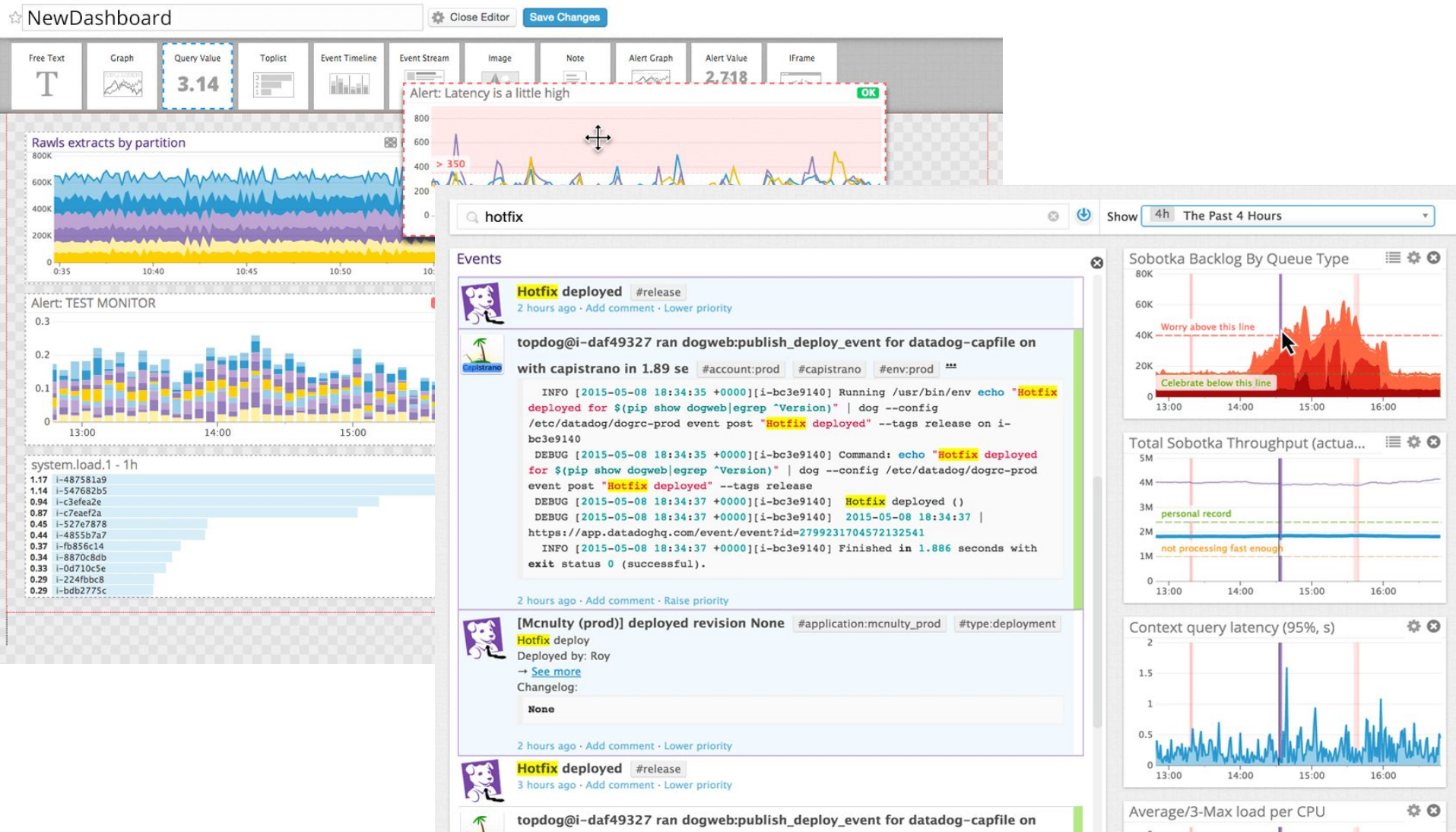
# NASA: Unlocking Interplanetary Datasets with Real-Time Search

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



<https://www.elastic.co/elasticon/2015/sf/unlocking-interplanetary-datasets-with-real-time-search>

# Datadog: analysis metrics and time-series data



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

更多:

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

# The “Elastic Stack”

User  
Interface



Store, Index,  
& Analyze

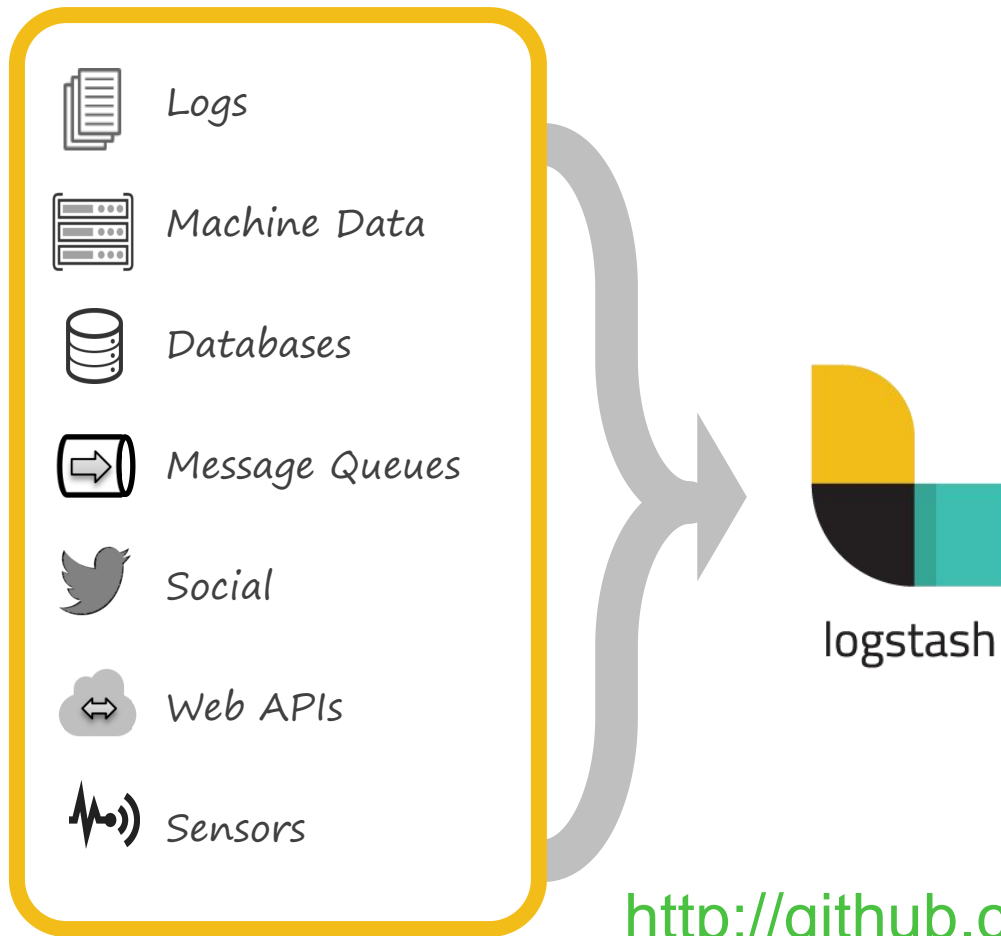


Ingest



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>

# The “Elastic Stack”

User  
Interface



Store, Index,  
& Analyze



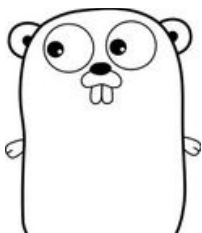
Ingest



Extensions



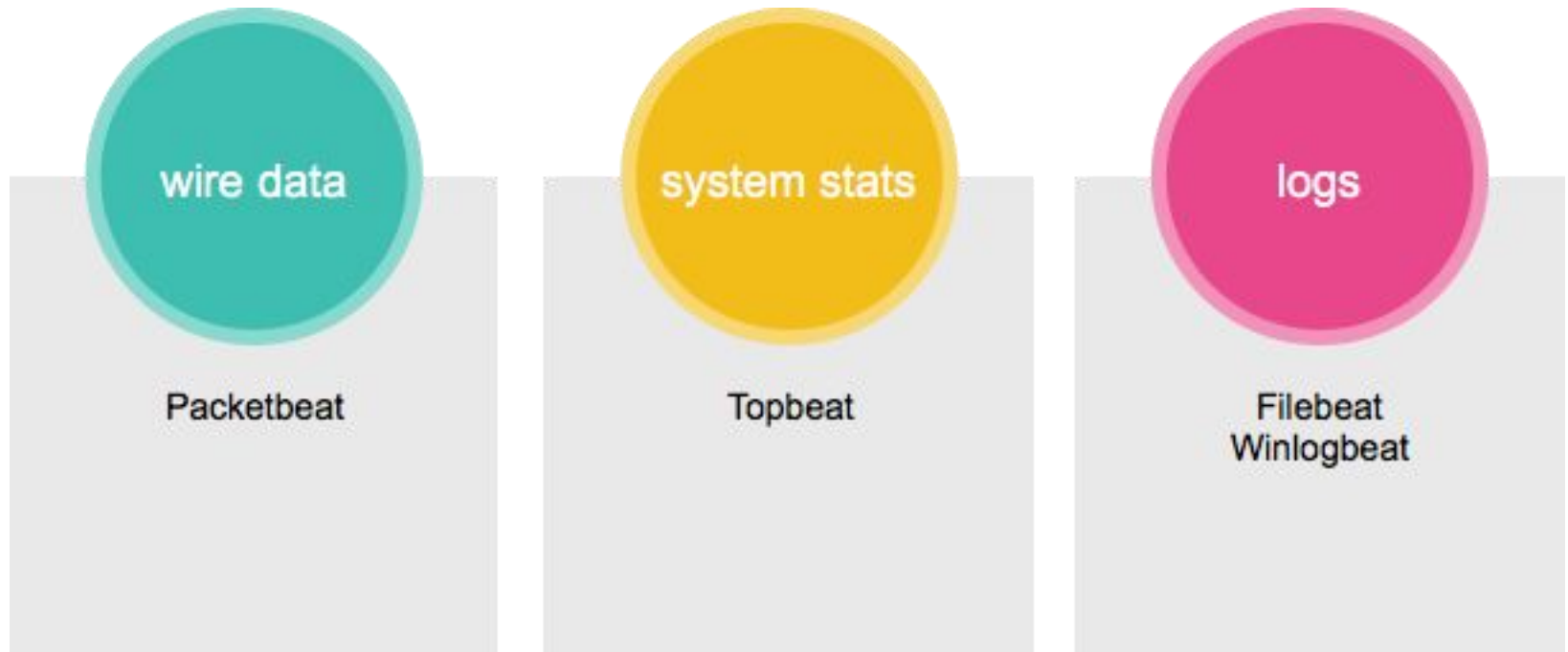
- Beats are lightweight shippers that collect and ship all kinds of **operational data** to Elasticsearch
  - Small application
  - Install as agent on your servers
  - Written in **Golang**
  - No runtime dependencies
  - Single purpose



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

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

# Examples of operational data



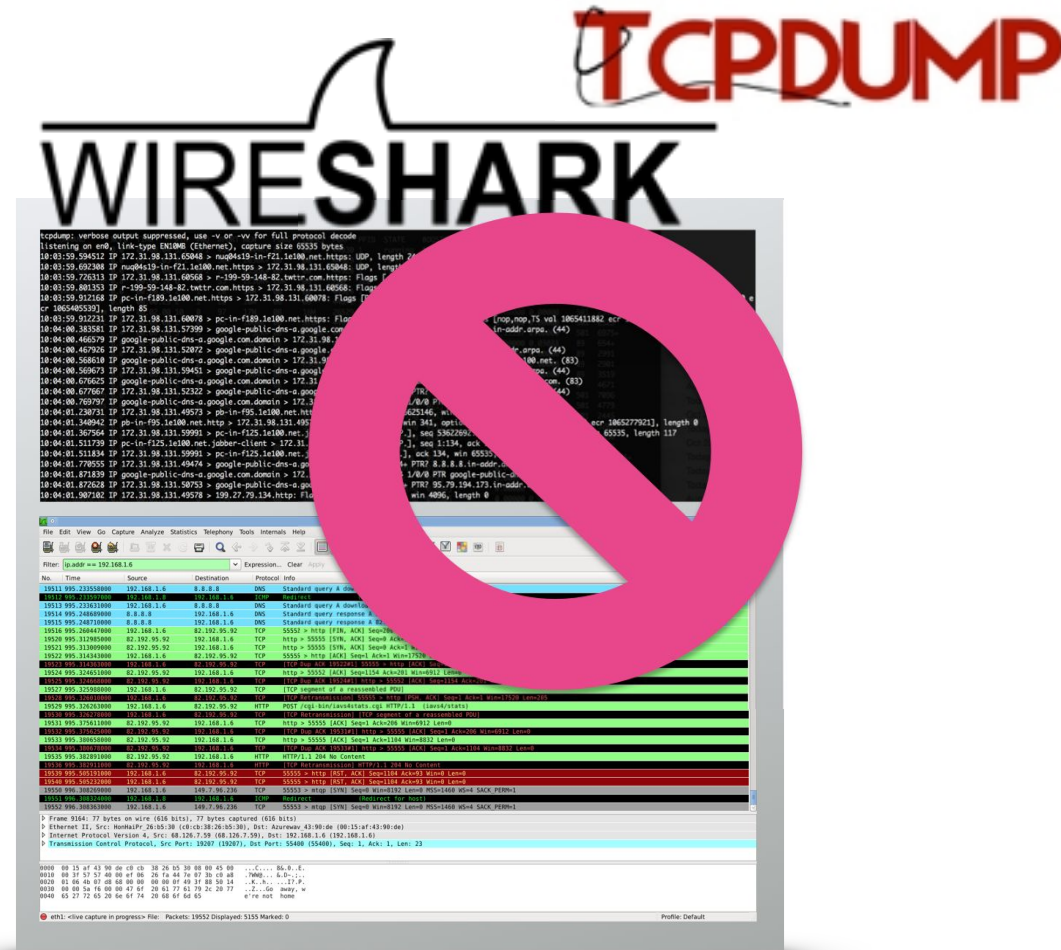
# Packetbeat: Real-time application monitoring

**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
- ICMP
- AMQP

Let's go realtime!



# winlogbeat!

Forwards Windows Event logs to Elasticsearch

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](#)

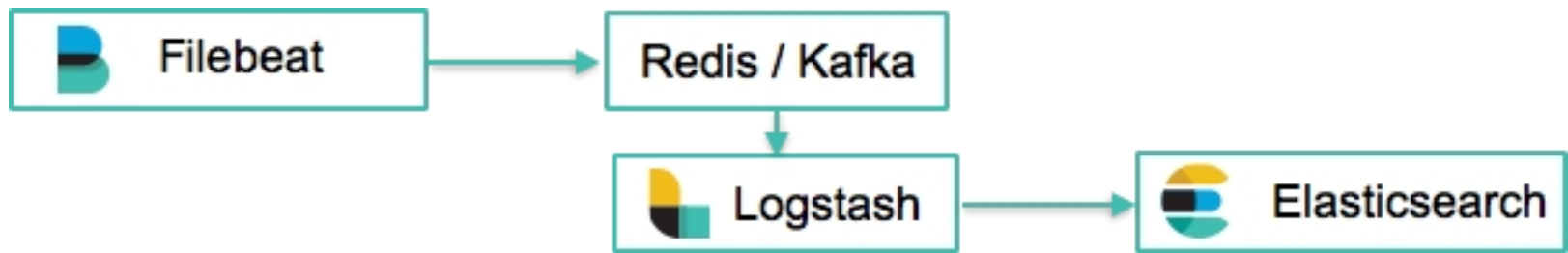
Logged: 1/13/2015 9:26:35 AM  
Task Category: None  
Keywords: Classic  
Computer: vagrant-2012-r2

# Filebeat

A more lightweight log shipper

- **Generic filtering**

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



# 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 ...

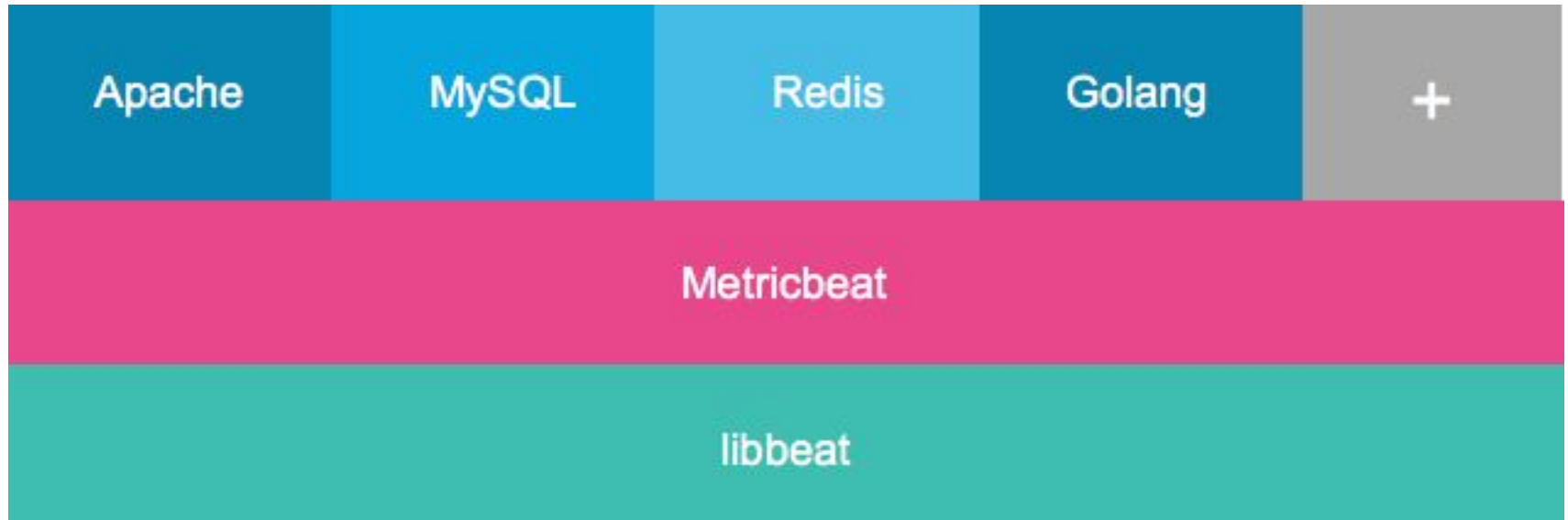
```
Processes: 367 total, 3 running, 6 stuck, 358 sleeping, 1833 threads
Load Avg: 2.79, 2.69, 2.66 CPU usage: 35.77% user, 8.23% sys, 55.90% idle
SharedLibs: 161M resident, 22M data, 18M linkedit. MemRegions: 132196 total, 5764M resident, 90M private
PhysMem: 13G used (3762M wired), 2581M unused.
VM: 2084G vsize, 527M framework vsize, 56231732(0) swappins, 59291827(0) swapouts.
Networks: packets: 66492613/48G in, 57364574/31G out. Disks: 6744547/369G read, 34720568/883G written.
```

| PID    | COMMAND      | %CPU | TIME     | #TH   | #WQ | #PORTS | MEM   | PURG  | OPRS  | PGPR  | 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 | 02:05.82 | 74    | 0   | 186    | 309M  | 00    | 69M   | 29185 | 98883 | sleeping | *0[2]     |
| 15112  | topbeat      | 18.7 | 02:56:57 | 12    | 0   | 61     | 9000K | 00    | 1108K | 15112 | 15111 | sleeping | *0[1]     |
| 325    | tfTerm       | 12.8 | 02:18:11 | 12    | 4   | 366    | 90M   | 4096K | 40M   | 325   | 1     | sleeping | *0[19746] |
| 65237  | top          | 7.4  | 00:04.34 | 1/1   | 0   | 24     | 7012K | 00    | 00    | 65237 | 18734 | running  | *0[1]     |
| 0      | kernel_task  | 4.6  | 23:39:40 | 220/4 | 0   | 2      | 1719M | 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     | 8248K | 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- | dbusd        | 0.4  | 01:45.43 | 1     | 0   | 7      | 4168K | 00    | 140K  | 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     | fsevents     | 0.3  | 30:30.91 | 13    | 0   | 306    | 4796K | 00    | 4104K | 46    | 1     | sleeping | *0[1]     |
| 23795- | dbusd        | 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  | 20:00:22 | 48    | 1   | 1998   | 943M  | 736K  | 1249M | 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    | symptomad    | 0.1  | 16:32.42 | 4     | 2   | 182    | 2184K | 00    | 2112K | 212   | 1     | sleeping | *0[66876] |
| 23793- | dbusd        | 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    | 98M   | 00    | 00    | 316   | 316   | sleeping | *0[2]     |
| 16578  | Google Chrom | 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 Chrom | 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    | 20M   | 54832 | 1     | sleeping | *0[1856]  |
| 62861  | Google Chrom | 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] |

# That's More!

## Metricbeat: Connecting Numb3rs

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



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

# The “Elastic Stack”

User  
Interface



Store, Index,  
& Analyze



Ingest



Extensions

# What's Kibana?



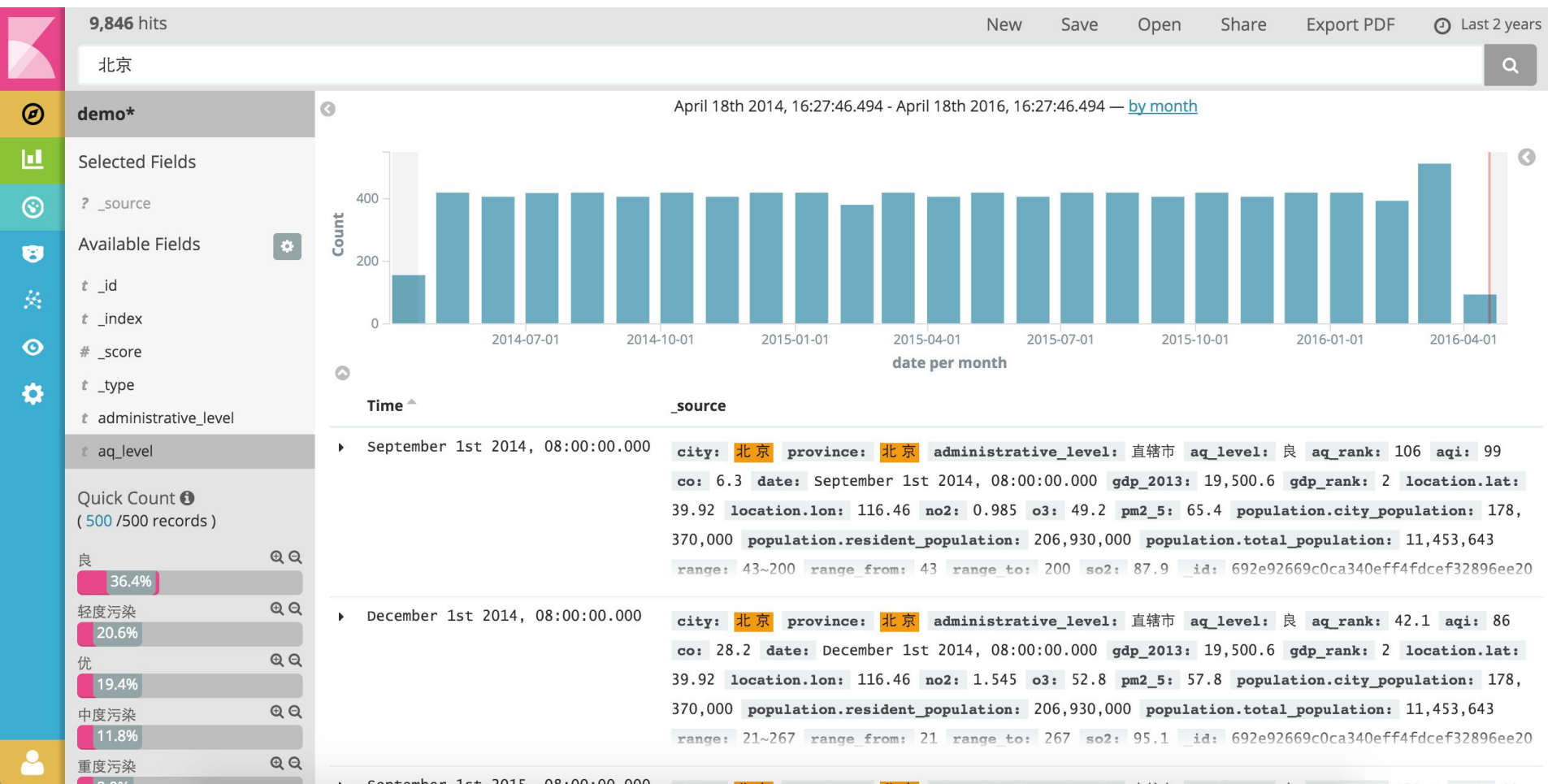
**Kibana** is an open source analytics and **visualization platform** designed to work with Elasticsearch.



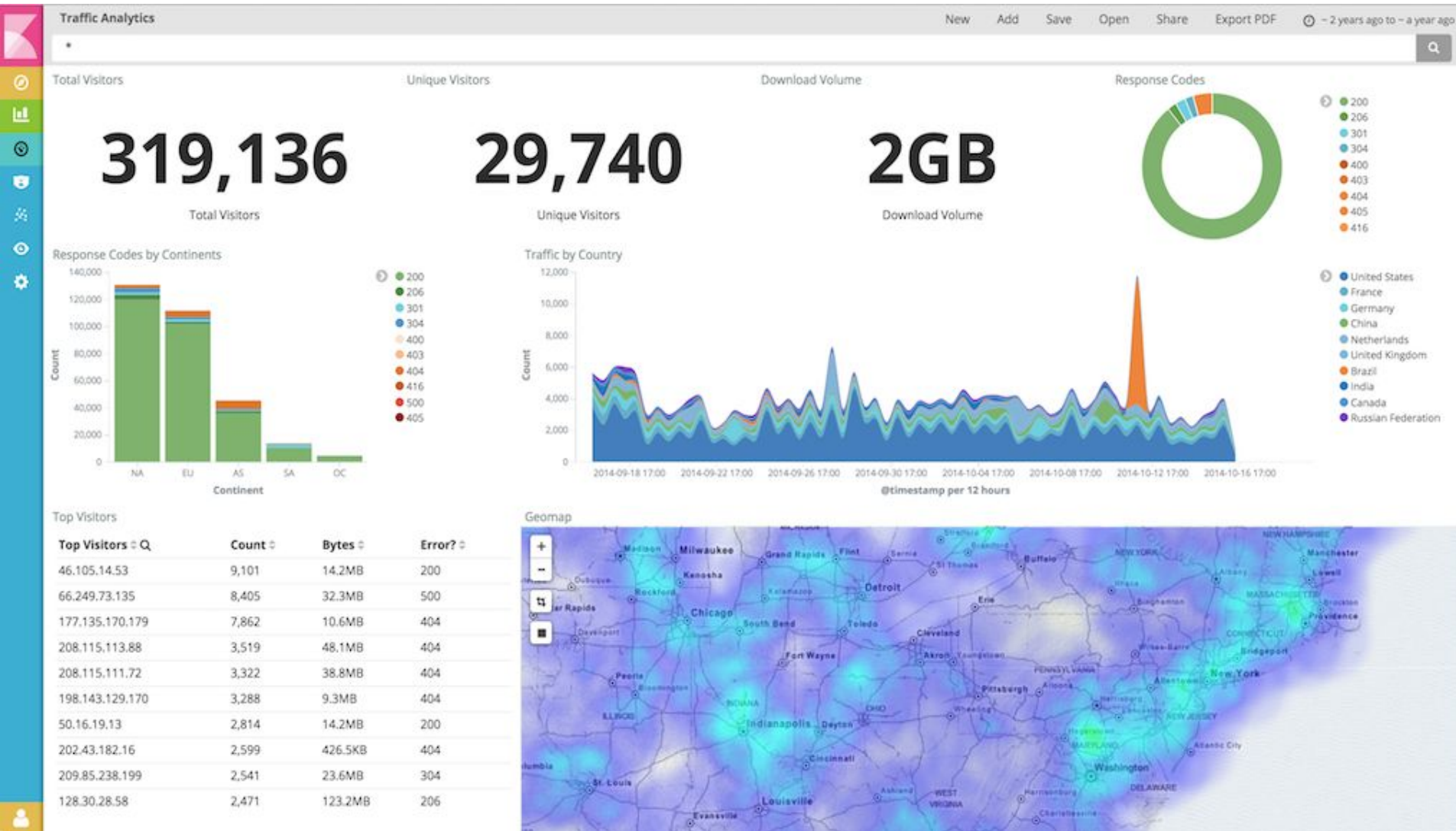
<http://github.com/elastic/kibana>

<https://github.com/elastic/generator-kibana-plugin>

# Search & Exploration



# Visualization & Dashboard



# The “Elastic Stack”

User  
Interface



kibana

Store, Index,  
& Analyze



elasticsearch

Ingest



logstash



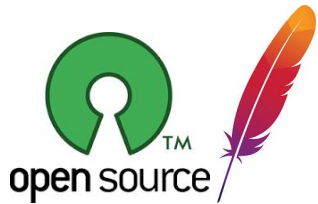
beats



Extensions



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

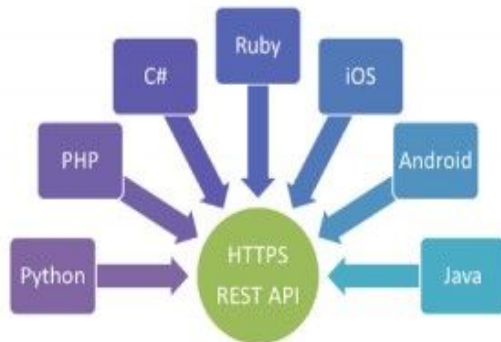


Thomson Reuters: “107 clusters ~1747 nodes” @Elastic{ON}16

<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>



慢着，Elasticsearch 不是搜索引擎么？

You know for search, and analytics !

**v0.09.0: ~~Facets~~**

**v1.0.0: Aggregation**

**v2.0.0: Pipeline Aggregation**

# Aggregation

- Analytics

柱状图、分布、统计、地理...

任何数据

能被查询到的数据就能被分析

接近实时

按需实时计算，~1s 刷新闻间隔

可嵌套组合

不像facets只有一级

# Aggregation

## Buckets:

Terms  
Histogram  
Geohash grids  
...



## Metrics:

min-avg-max  
Stats  
Cardinality  
...



```
SELECT COUNT( * ) , AVG( score) <--- Metrics
FROM `table`
GROUP BY province,city <--- Buckets
```

Aggregation == 3万英尺高空俯视 == Patterns



Find some beauty (insights)!

以 PM 2.5 数据分析为例

Or like this!

腾讯图片  
PP.OO.COM

```
{ "city": "北京", "date": "2016-02-08",  
  "aq_level": "严重污染", "aq_rank": 68,  
    "aqi": 391, "co": 115.5,  
  "no2": 1.888, "o3": 62.2, "pm2_5": 415.7,  
    "range": "74~500", "so2": 523.5,  
  "location": { "lat": 39.92, "lon": 116.46 } }
```

## Structure of an Aggregation

```
"aggregations" : {  
  "<aggregation_name>" : {  
    "<aggregation_type>" : {  
      <aggregation_body>  
    }  
    [, "meta" : { [ <meta_data_body> ] } ]?  
    [, "aggregations" : { [ <sub_aggregation> ]+ } ]?  
  }  
  [, "<aggregation_name_2>" : { ... } ]*  
}
```

## 空气质量统计（北京全年）

POST demo/\_search?siz

```
{
  "query": {...}
  "aggs": {
    "aq_stats": {
      "terms": {
        "field":
        "size": 1
      }
    }
  }
}
```

```
{
  "aggregations": {
    "aq_stats": {
      "buckets": [
        {
          "key": "良",
          "doc_count": 118
        },
        {
          "key": "优",
          "doc_count": 102
        },
        {
          "key": "轻度污染",
          "doc_count": 64
        },
        {
          "key": "中度污染",
          "doc_count": 40
        },
        {
          "key": "重度污染",
          "doc_count": 21
        },
        {
          "key": "严重污染",
          "doc_count": 18
        }
      ]
    }
  }
}
```

## 平均空气质量统计（按城市）（1）

```
{
  "aggs": {
    "city_stats": {
      "terms": {
        "field": "city",
        "size": 10
      }
    },
    "avg_pm25": {
      "avg": {
        "field": "pm25"
      }
    }
  }
}
```

```
{
  "aggregations": {
    "city_stats": {
      "buckets": [
        {
          "key": "河源",
          "doc_count": 568,
          "avg_pm25": {
            "value": 38.366549295774654
          }
        },
        {
          "key": "佛山",
          "doc_count": 562,
          "avg_pm25": {
            "value": 42.66227758007118
          }
        },
        {
          "key": "汕头",
          "doc_count": 561,
          "avg_pm25": {
            "value": 37.265953654188955
          }
        },
        {
          "key": "泰安",
          "doc_count": 561,
          "avg_pm25": {
            "value": 76.28074866310159
          }
        }
      ]
    }
  }
}
```

## 30天空气质量趋势分析 (Pip

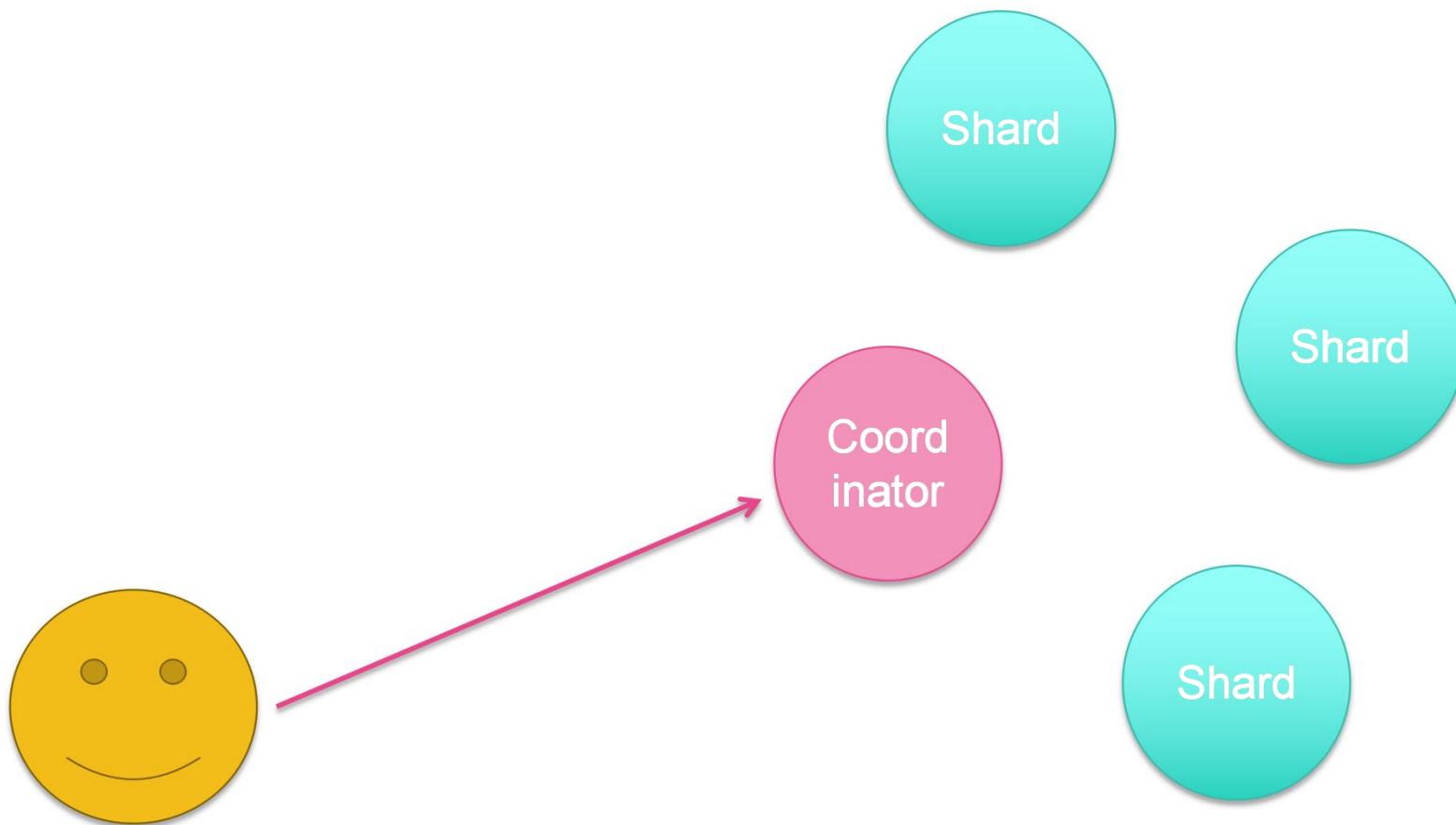
```
{ "aggs": {
  "qa_date_histo": {
    "date_histogram": {
      "field": "date",
      "interval": "30d"
    },
    "aggs": {
      "the_avg": {
        "avg": {
          "field": "aqi"
        }
      },
      "the_movavg": {
        "moving_avg": {
          "window": 30,
          "units": "days"
        }
      }
    }
  }
}
```

```
{
  "key_as_string": "2016-02-06T00:00:00.000Z",
  "key": 1454716800000,
  "doc_count": 189,
  "the_avg": {
    "value": 50.41798941798942
  },
  "the_movavg": {
    "value": 67.51189039993638
  }
},
{
  "key_as_string": "2016-02-07T00:00:00.000Z",
  "key": 1454803200000,
  "doc_count": 189,
  "the_avg": {
    "value": 65.5026455026455
  },
  "the_movavg": {
    "value": 66.97335424473356
  }
},
{
  "key_as_string": "2016-02-08T00:00:00.000Z",
  "key": 1454889600000,
  "doc_count": 189,
  "the_avg": {
    "value": 135.29470899470903
  },
  "the_movavg": {
    "value": 66.63231368035966
  }
}
```

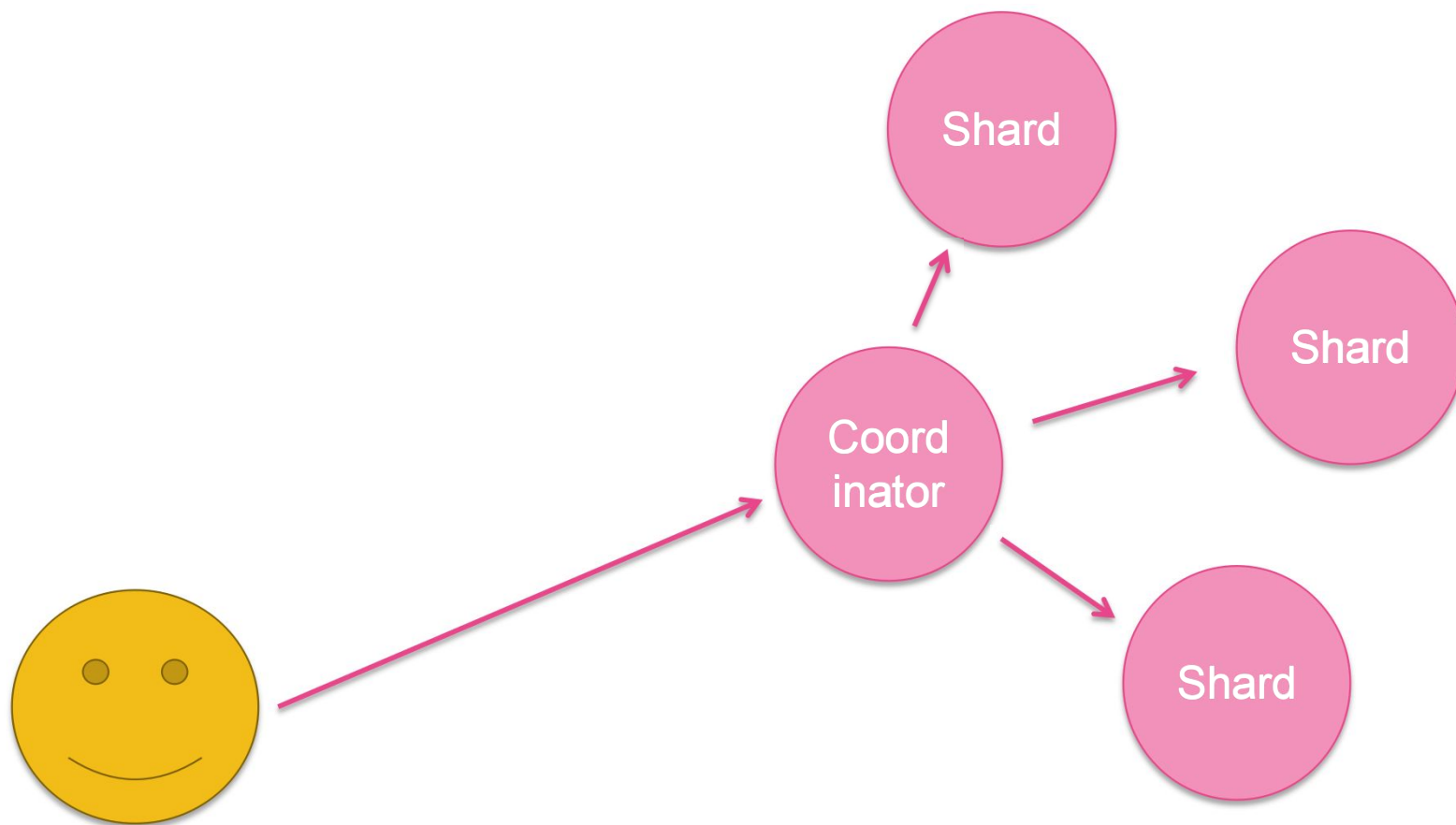
# Aggregation 工作原理

- Lucene Collector
- Optimized data structure
  - Compressed columnar datastore (previous FieldData, now DocValues)
  - Strings converted to enums (per segment)
- Single pass on your data, alone with the query
  - No matter how complex of your aggregation

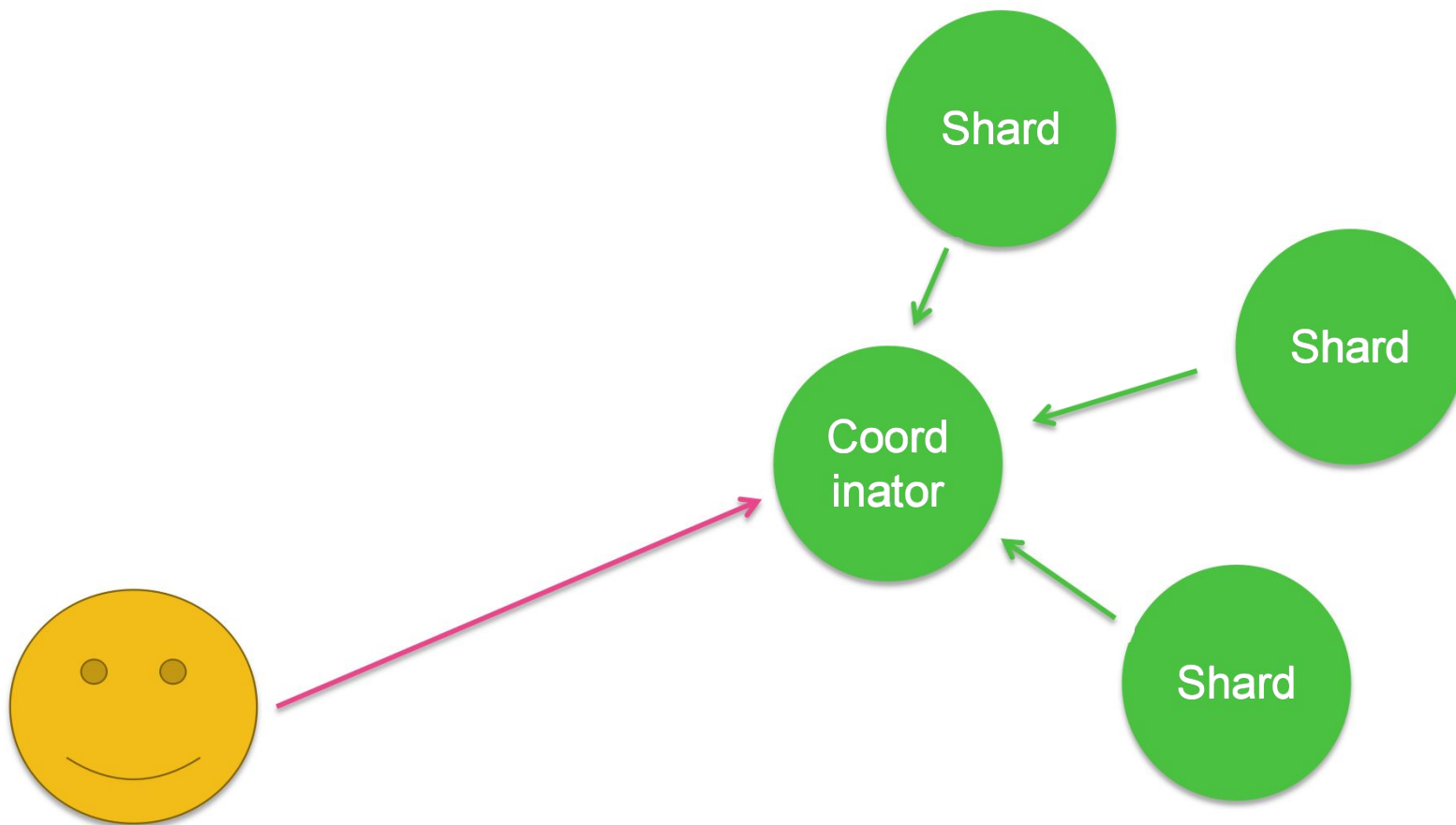
# Aggregation 工作原理



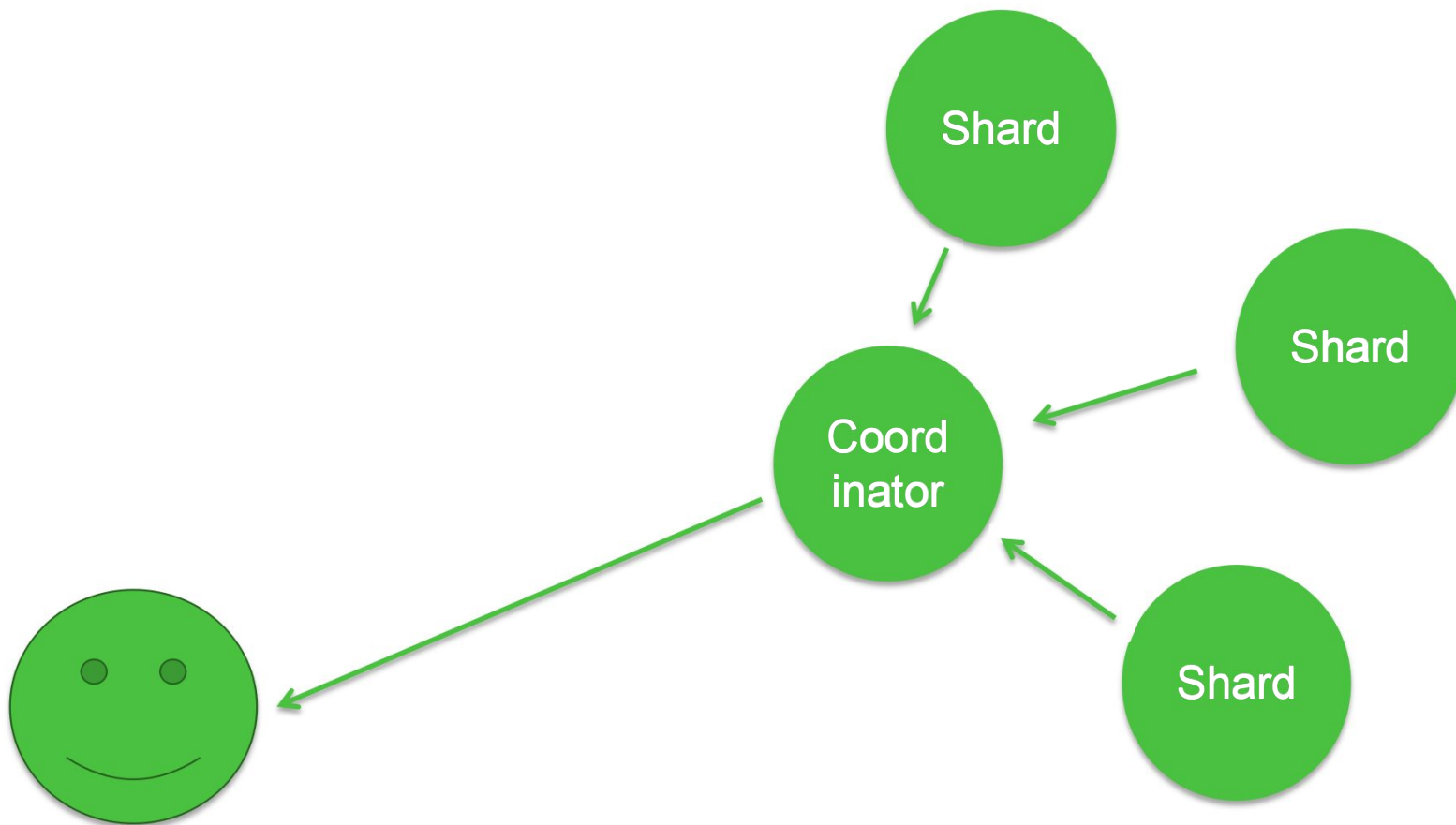
# Aggregation 工作原理



# Aggregation 工作原理



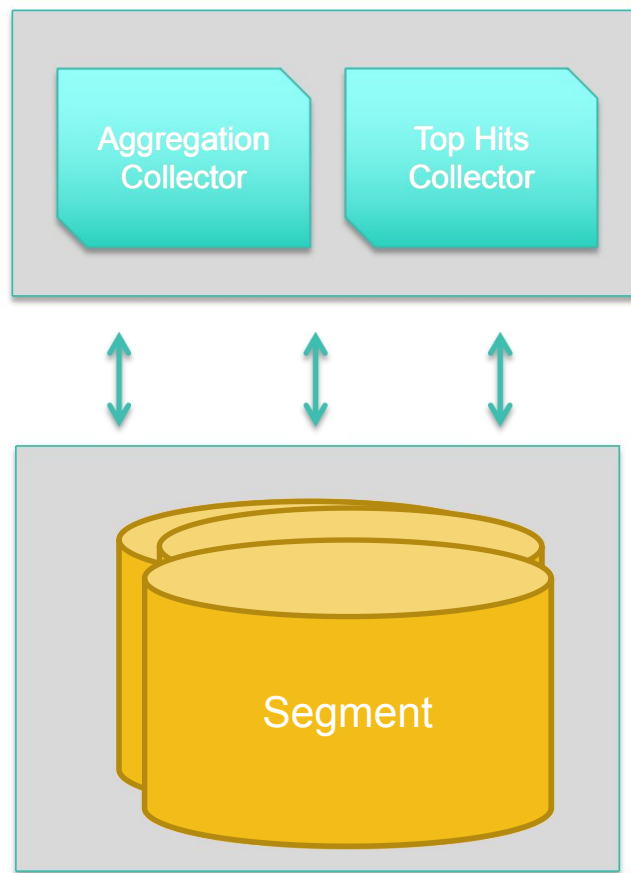
# Aggregation 工作原理



# Aggregation 工作原理

Aggregation:  
DocValues

| DocID | Field   |
|-------|---------|
| 1     | 北京      |
| 2     | 上海      |
| 3     | Beijing |
| 4     | 上海      |
|       |         |



Lucene Index/ An ES Shard

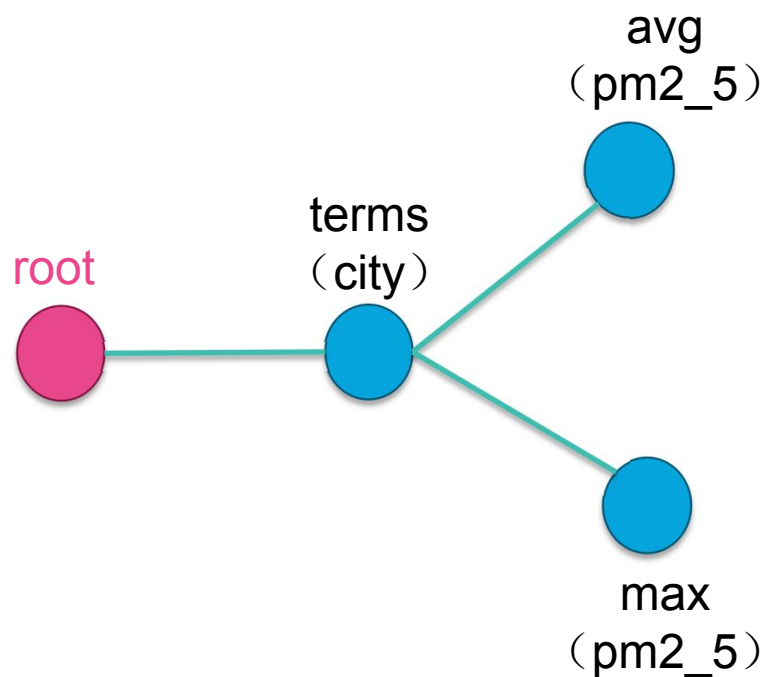
Search:  
Invert Index

| Term    | DocID         |
|---------|---------------|
| 北京      | 1, 5, 7       |
| 上海      | 2, 4, 6       |
| 广州      | , 11, 19, 23  |
| Beijing | 3, 12, 13, 15 |
|         |               |

# Aggregation 工作原理

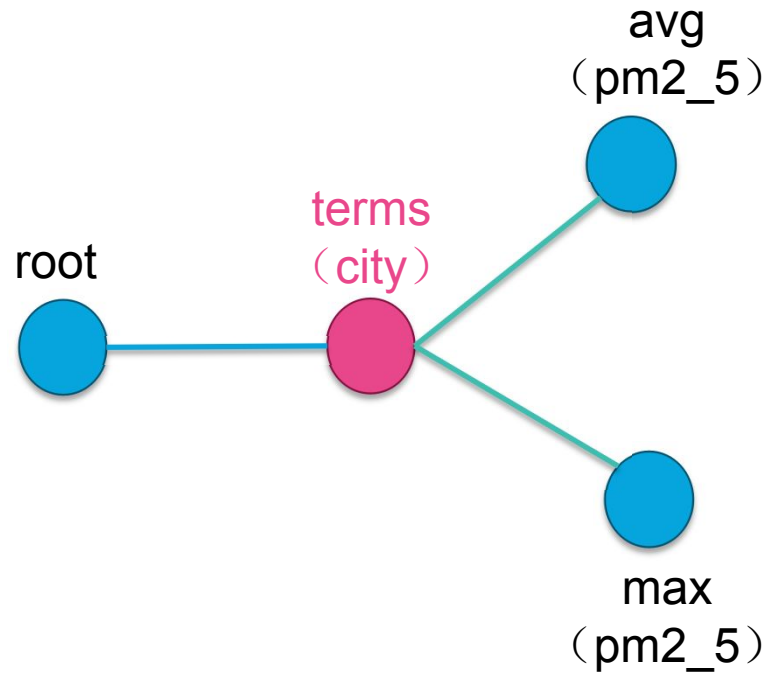
POST demo/\_search?size=0

```
{
  "aggs": {
    "city_stats": {
      "terms": {
        "field": "city",
        "size": 10
      }
    },
    "aggs": {
      "avg_pm25": {
        "avg": {
          "field": "pm2_5"
        }
      },
      "max_pm25": {
        "max": {
          "field": "pm2_5"
        }
      }
    }
  }
}
```



# Aggregation 工作原理

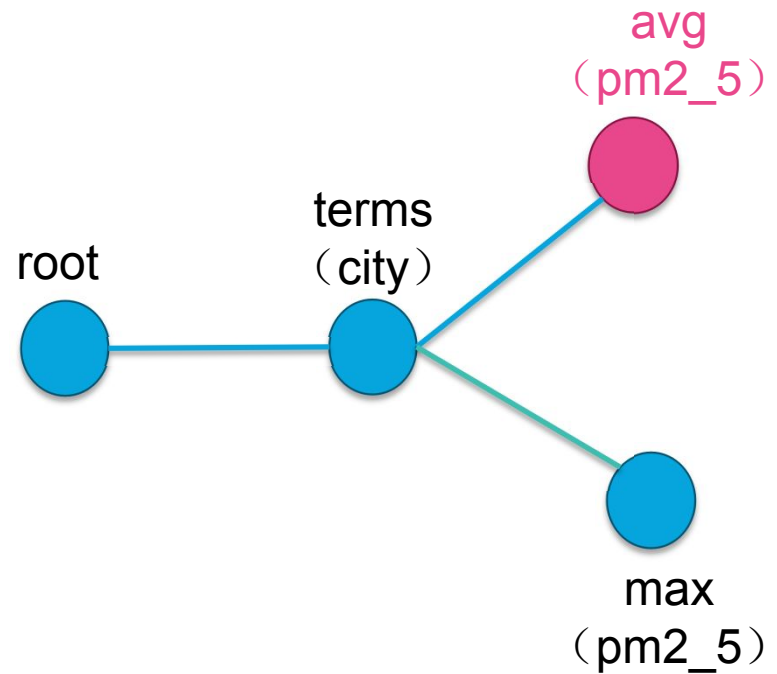
```
POST demo/_search?size=0
{
  "aggs": {
    "city_stats": {
      "terms": {
        "field": "city",
        "size": 10
      }
    },
    "aggs": {
      "avg_pm25": {
        "avg": {
          "field": "pm2_5"
        }
      },
      "max_pm25": {
        "max": {
          "field": "pm2_5"
        }
      }
    }
  }
}
```



# Aggregation 工作原理

POST demo/\_search?size=0

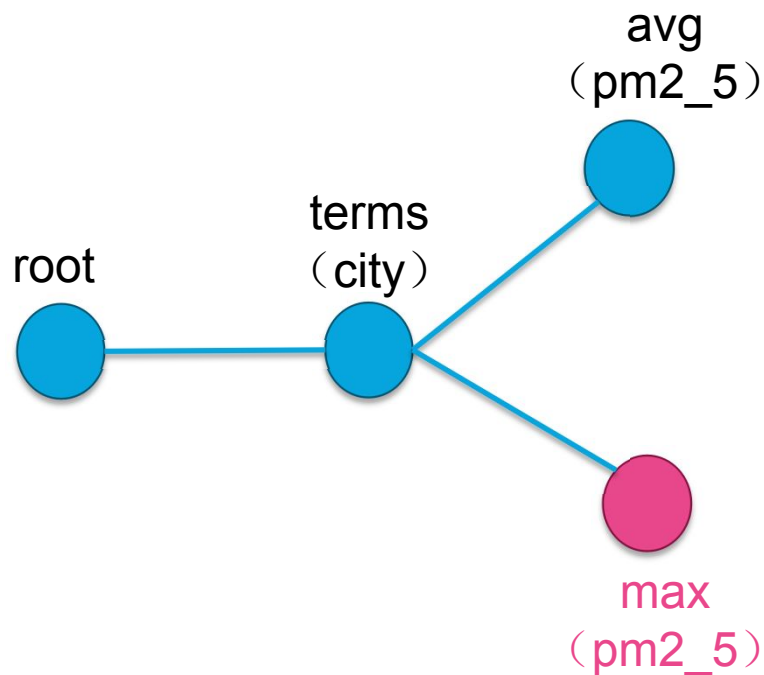
```
{
  "aggs": {
    "city_stats": {
      "terms": {
        "field": "city",
        "size": 10
      },
      "aggs": {
        "avg_pm25": {
          "avg": {
            "field": "pm2_5"
          }
        },
        "max_pm25": {
          "max": {
            "field": "pm2_5"
          }
        }
      }
    }
  }
}
```



# Aggregation 工作原理

POST demo/\_search?size=0

```
{
  "aggs": {
    "city_stats": {
      "terms": {
        "field": "city",
        "size": 10
      }
    },
    "aggs": {
      "avg_pm25": {
        "avg": {
          "field": "pm2_5"
        }
      },
      "max_pm25": {
        "max": {
          "field": "pm2_5"
        }
      }
    }
  }
}
```

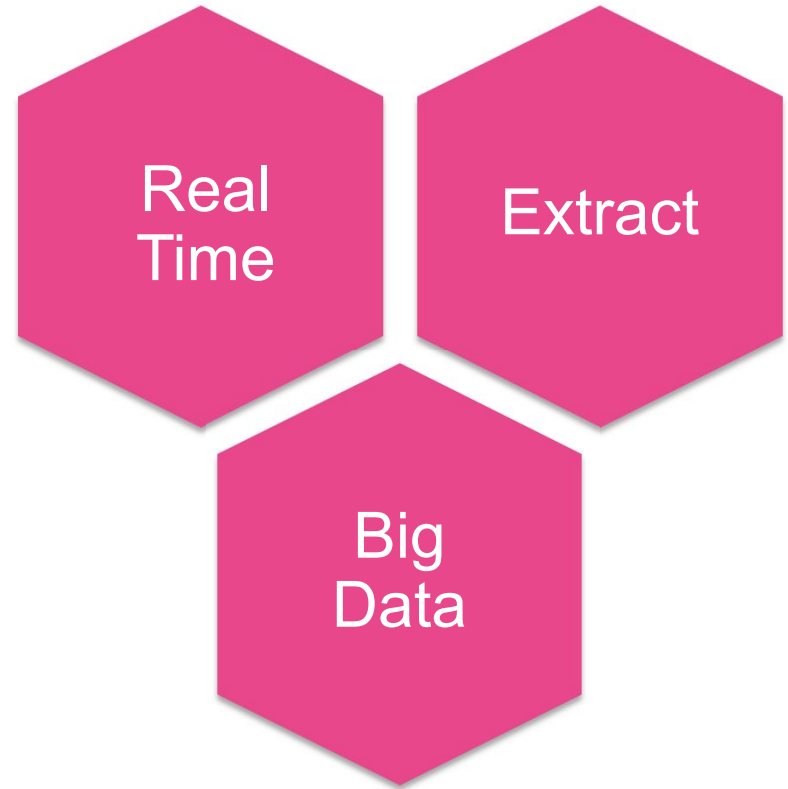


# What's more?

- 近似算法 (Approximate algorithms)

- 唯一值: Cardinality
  - Hyperloglog++
- 百分位: Percentile
  - TDigest

Fixed memory!



- 控制效率和内存占用

- Terms
  - Breath\_first collect mode
- Sampler
  - Max docs per shard

- 更多有趣的Aggregation!

- Significant Terms Aggregation
  - The uncommonly common
- Geohash grid

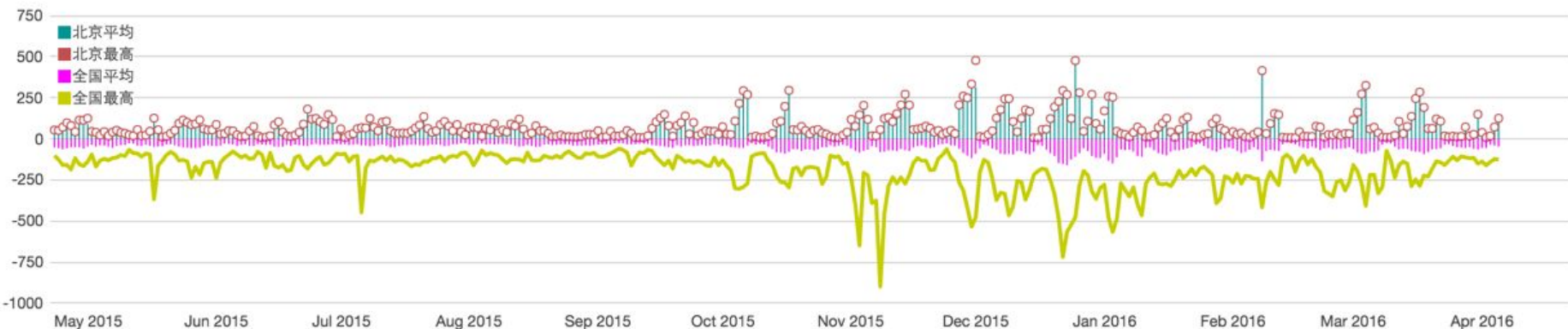


说好的图形界面呢?

# DEMO

## Kibana、Timelion、Graph

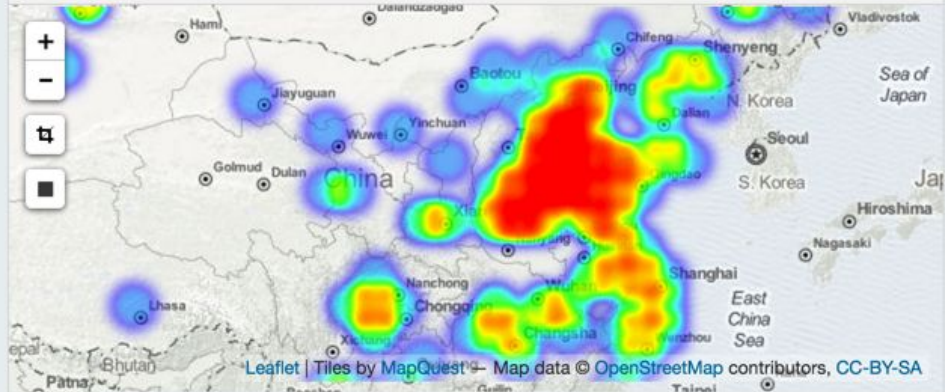
北京污染情况

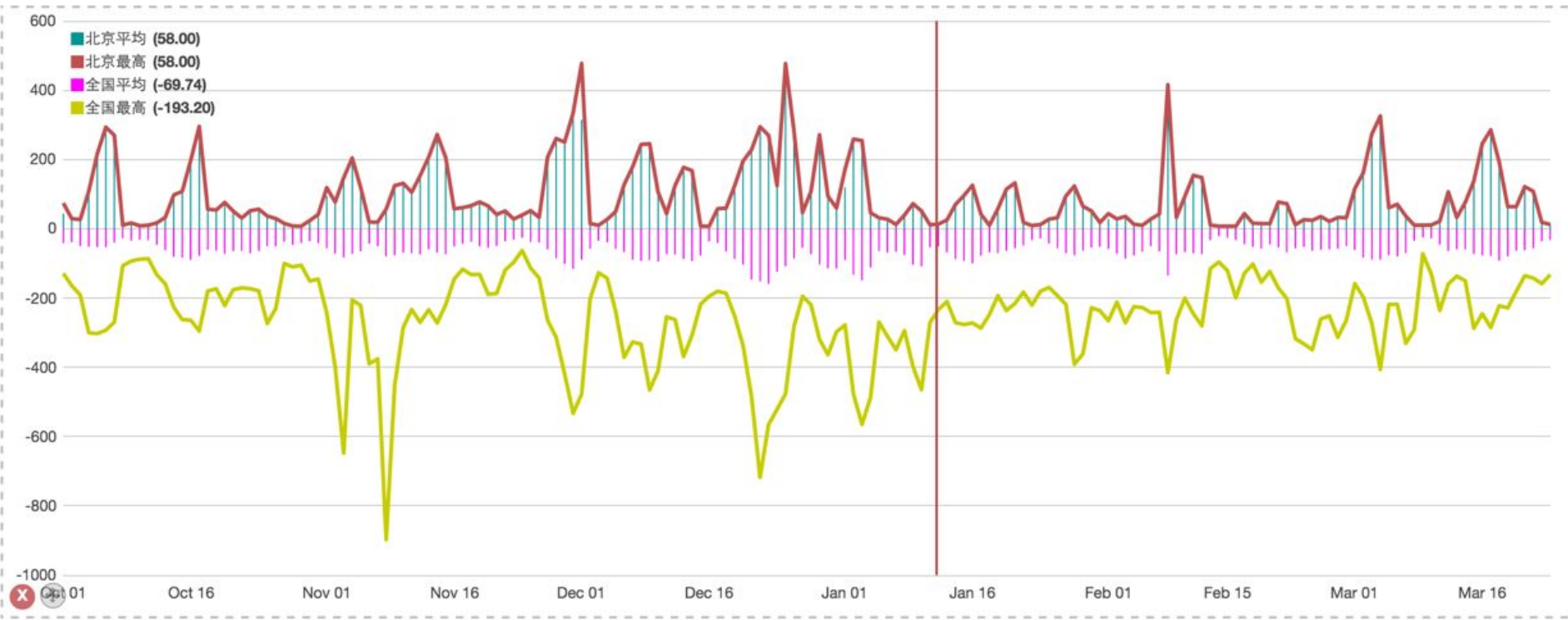


污染累计排行云图

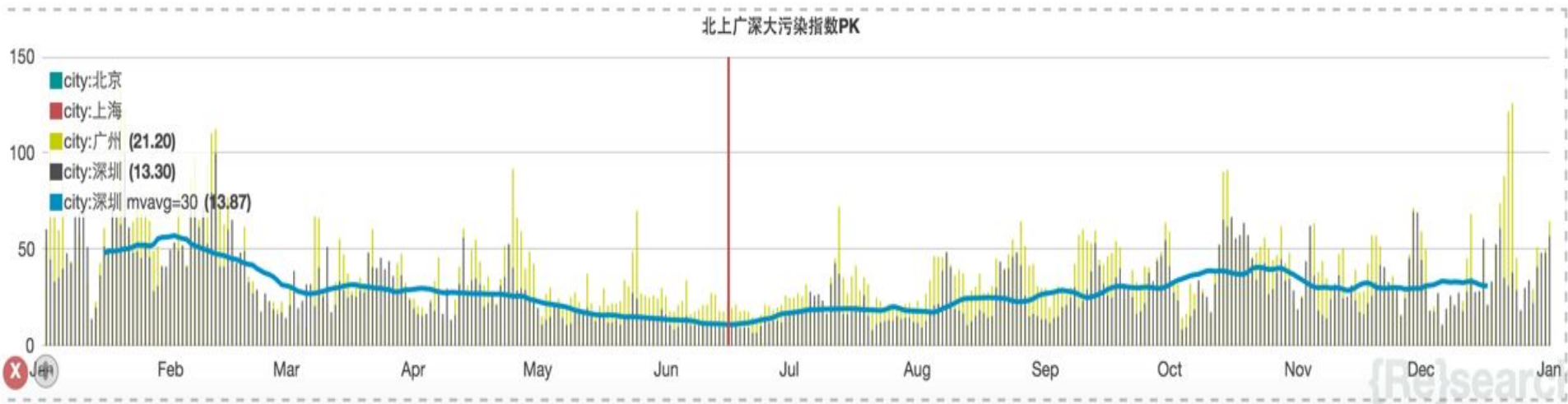
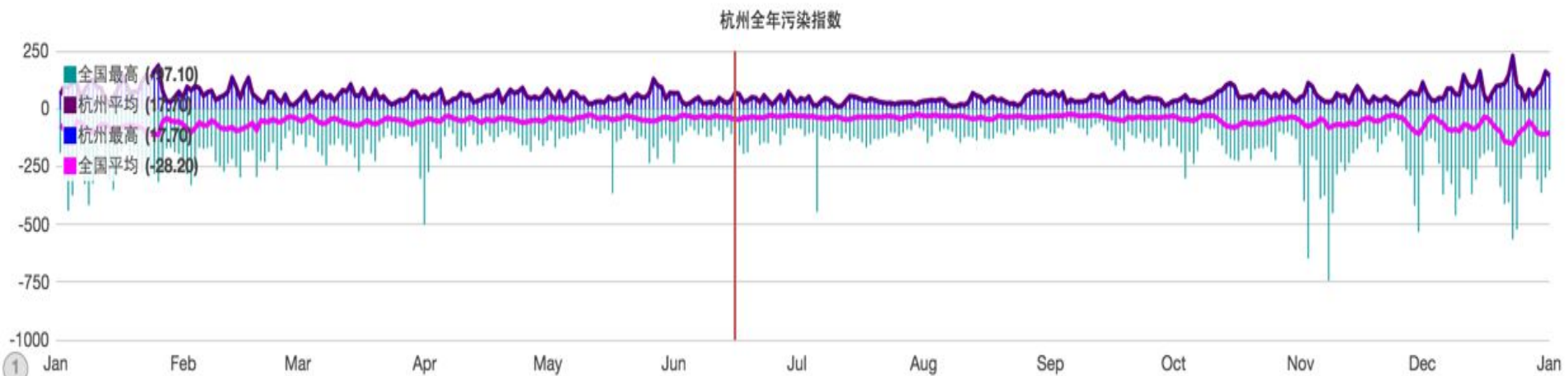


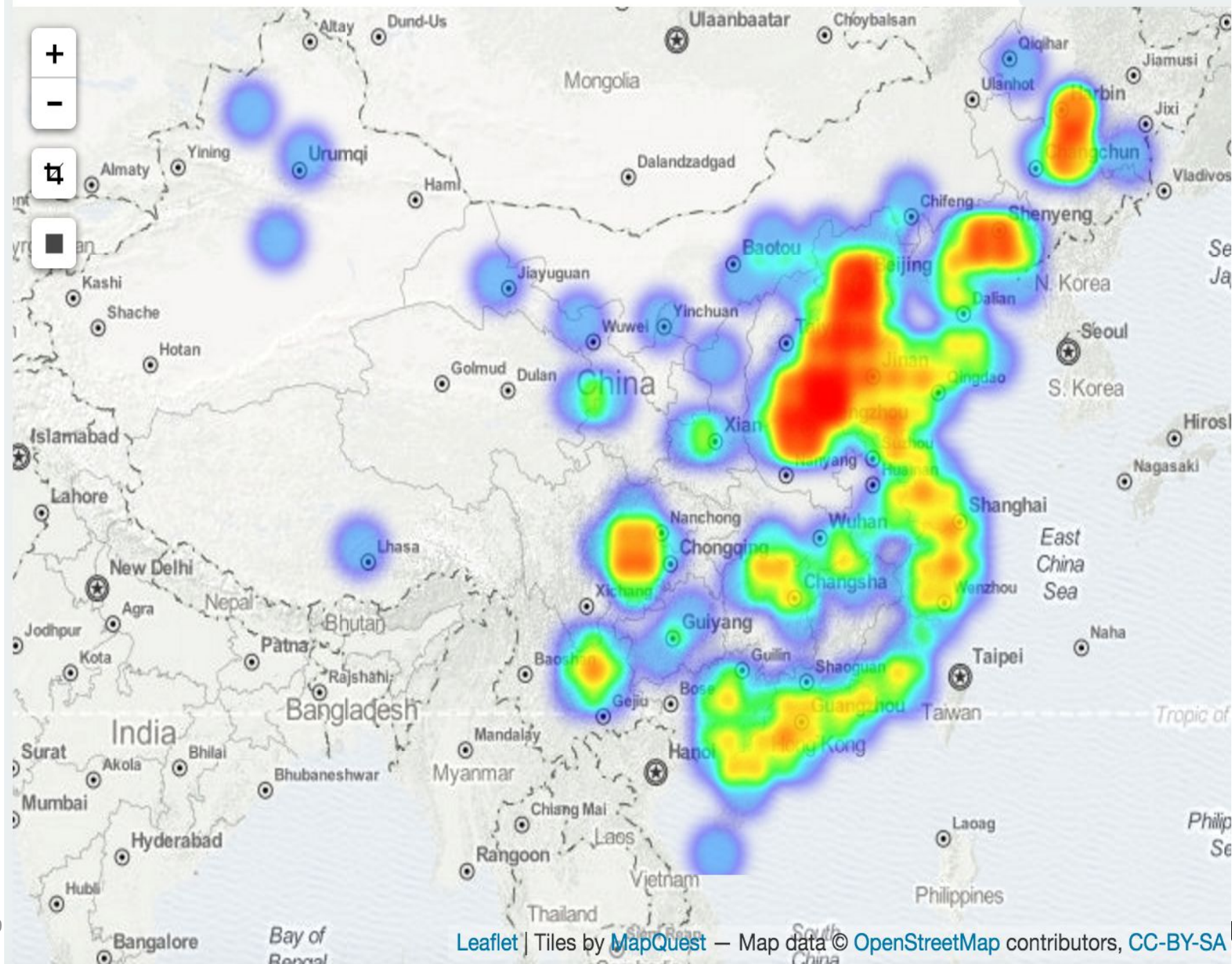
平均PM2.5热点图

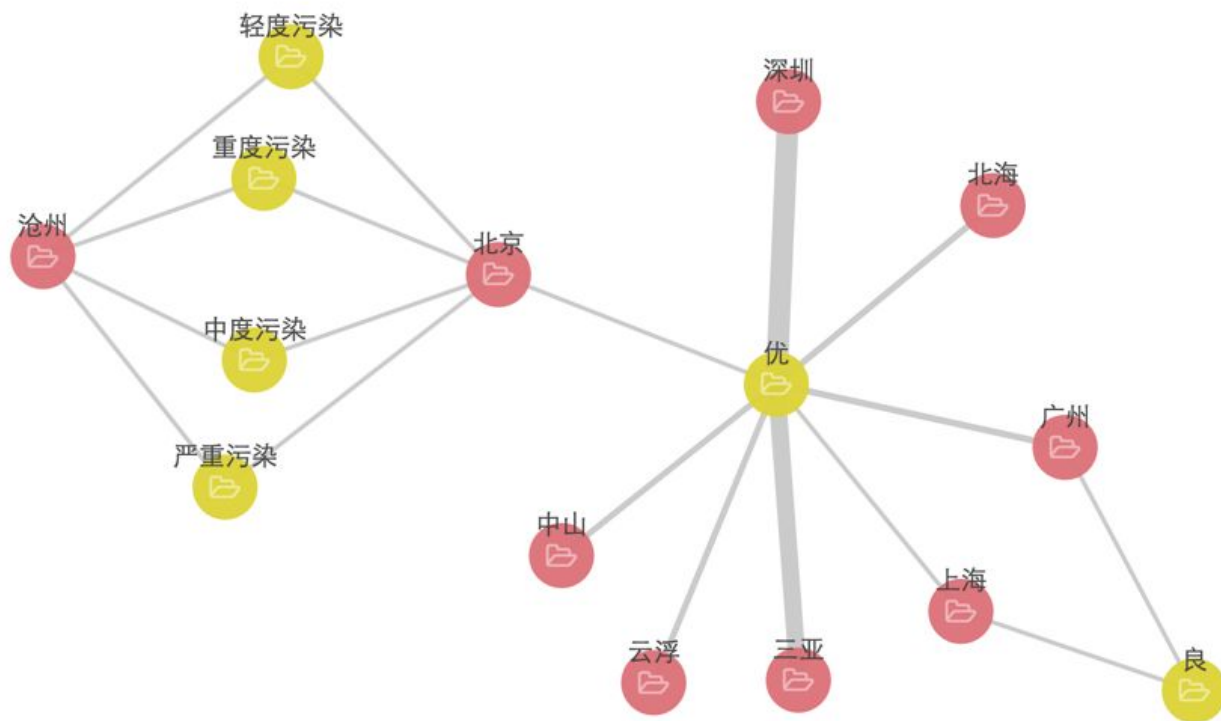




# DEMO







Selections

all

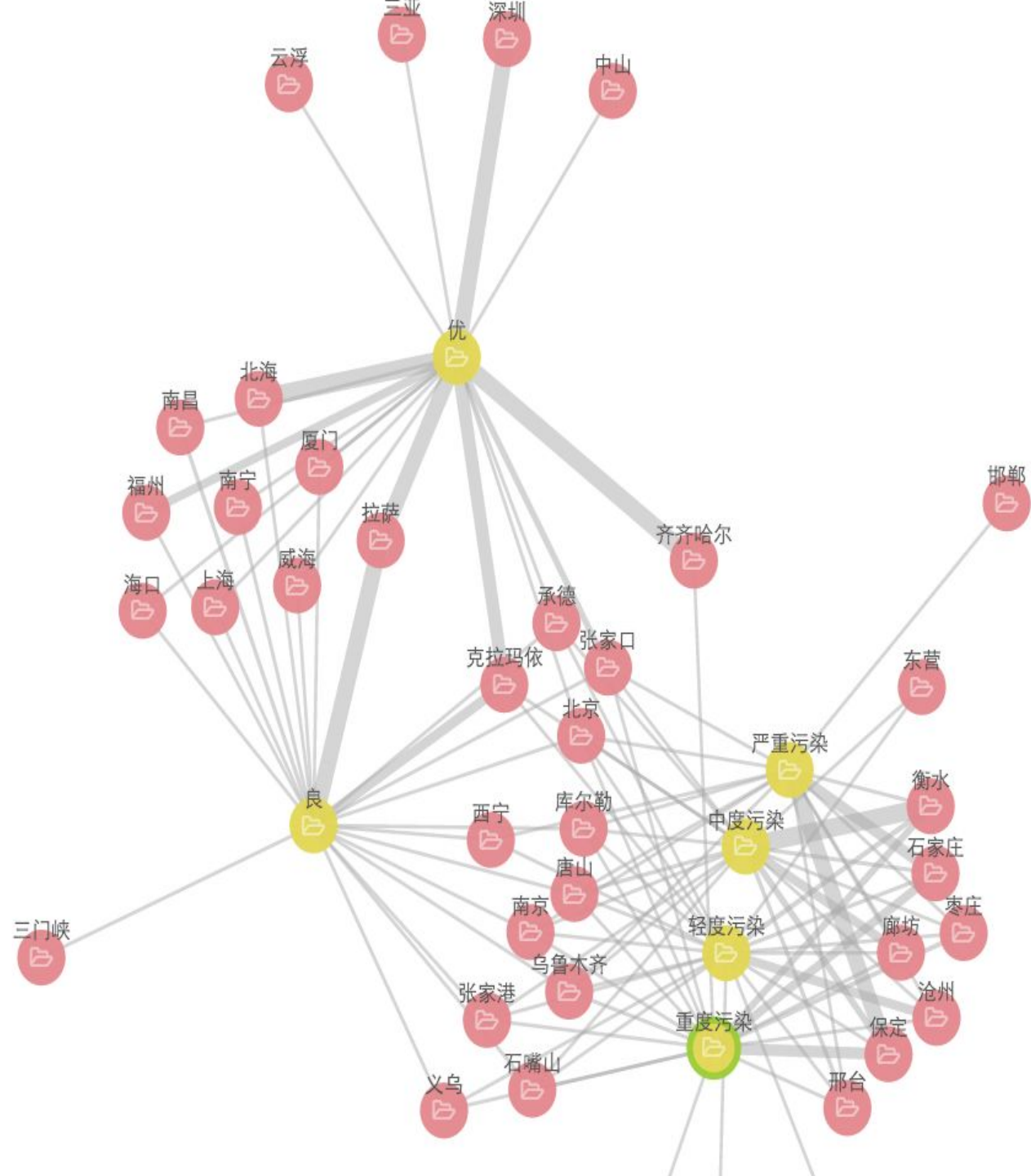
none

invert

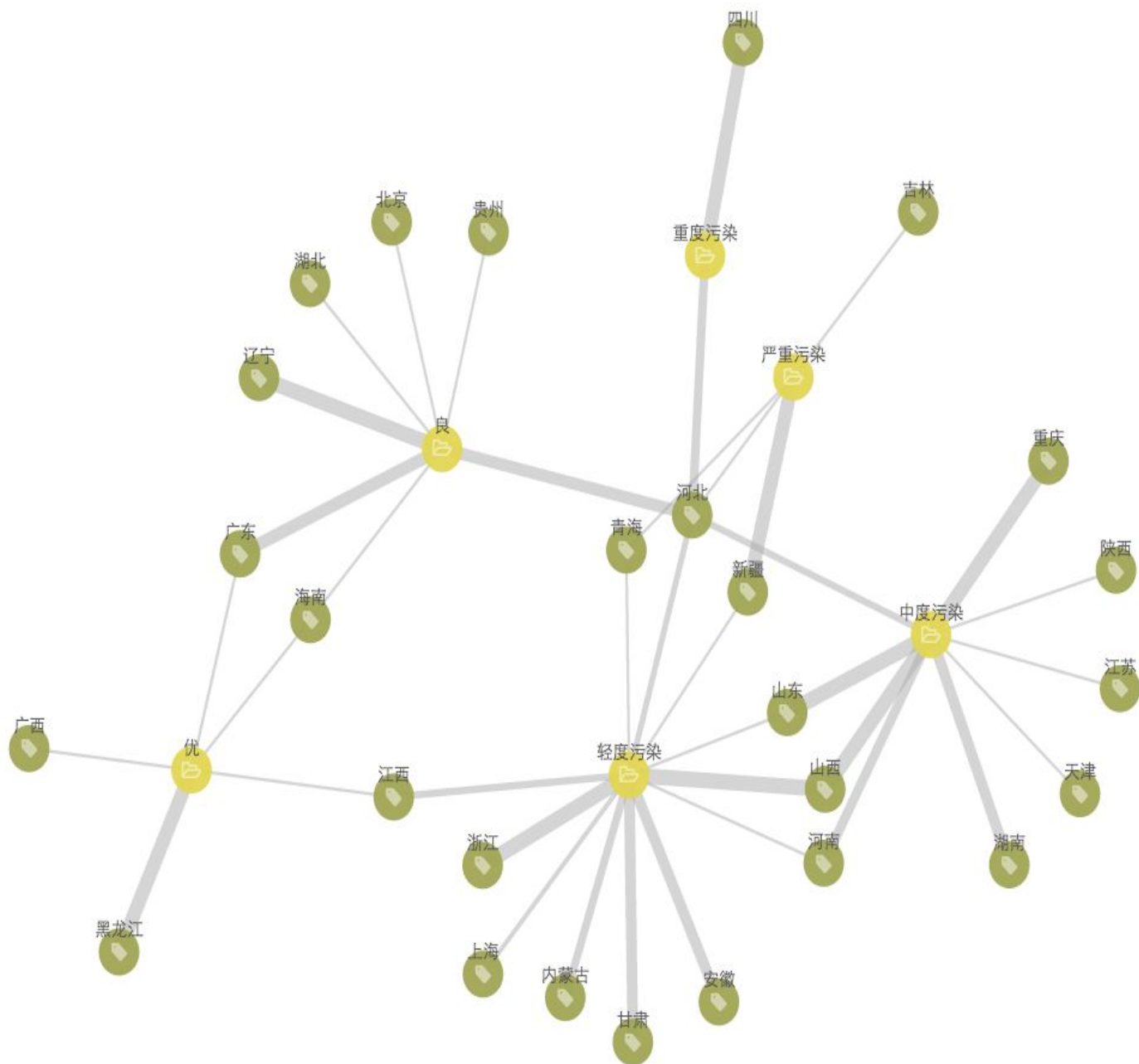
linked

No selections. Click on vertices to add

Demo



# Dem



# The “Elastic Stack”

User  
Interface



Store, Index,  
& Analyze



Ingest



Extensions

# Community

- 源码 & Issue: <http://github.com/elastic/>
- 英文社区: <http://discuss.elastic.co>
- 中文社区: <http://elasticsearch.cn>
- 官方 QQ 群: 190605846
- 下载: <https://www.elastic.co/downloads>
- 博客: <https://www.elastic.co/blog>
- 线下活动: <http://elasticsearch.meetup.com/>
- IRC: #elasticsearch, #logstash, #kibana, #beats
- 官方 Twitter [@elastic](https://twitter.com/elastic)





**More questions?**

欢迎来 Elastic 展台找我！

# Thanks!

# 附录

# ES 基本操作

## 索引（插入数据）

```
POST demo/pm2_5/1
```

```
{  
  "city": "北京", "date": "201  
  "aq_level": "严重污染", "aq_  
  "no2": 1.888, "o3": 62.2, "p  
  "so2": 523.5, "location": {  
}
```

```
{  
  "_index": "demo",  
  "_type": "pm2_5",  
  "_id": "1",  
  "_version": 1,  
  "_shards": {  
    "total": 2,  
    "successful": 1,  
    "failed": 0  
  },  
  "created": true  
}
```

# 基本操作

## 获取数据

```
GET demo/pm2_5/1
```

```
{
  "_index": "demo",
  "_type": "pm2_5",
  "_id": "1",
  "_version": 1,
  "found": true,
  "_source": {
    "aq_level": "严重污染", "aq_rank": 68,
    "aqi": 391, "city": "北京", "co": 115.5,
    "date": "2016-02-08", "province": "北京",
    "range": "74~500", "so2": 523.5
  }
}
```

# 基本操作

## 删除数据

```
DELETE demo/pm2_5/1
```

# 搜索

通过GET参数进行搜索

```
GET demo/pm2_5/_search?q=北京
```

# 搜索

查询北京的 PM2.5 数据

```
POST demo/pm2_5/_search
{
  "query": {
    "match": {
      "city": "北京"
    }
  }
}
```

# 搜索

查询北京的 PM2.5 数据

```
POST demo/pm2_5/_search
{"query": {"bool": {
  "must": [
    {"term": {"city": {"value":
"北京"}}},
    {"term": {"date": {"value":
"2016-02-08"}}}]}}
}
```

# 分析

## Aggregation

```
POST twitter/tweet/_search
```

```
{
  "aggs" : {
    "uers_stats" : {
      "terms" : { "field" : "user" }
    }
  }
}
```

```
{
  "took": 2,
  "timed_out": false,
  "_shards": {},
  "hits": {},
  "aggregations": {
    "uers_stats": {
      "doc_count_error_upper_bound": 0,
      "sum_other_doc_count": 0,
      "buckets": [
        {
          "key": "medcl",
          "doc_count": 1
        }
      ]
    }
  }
}
```