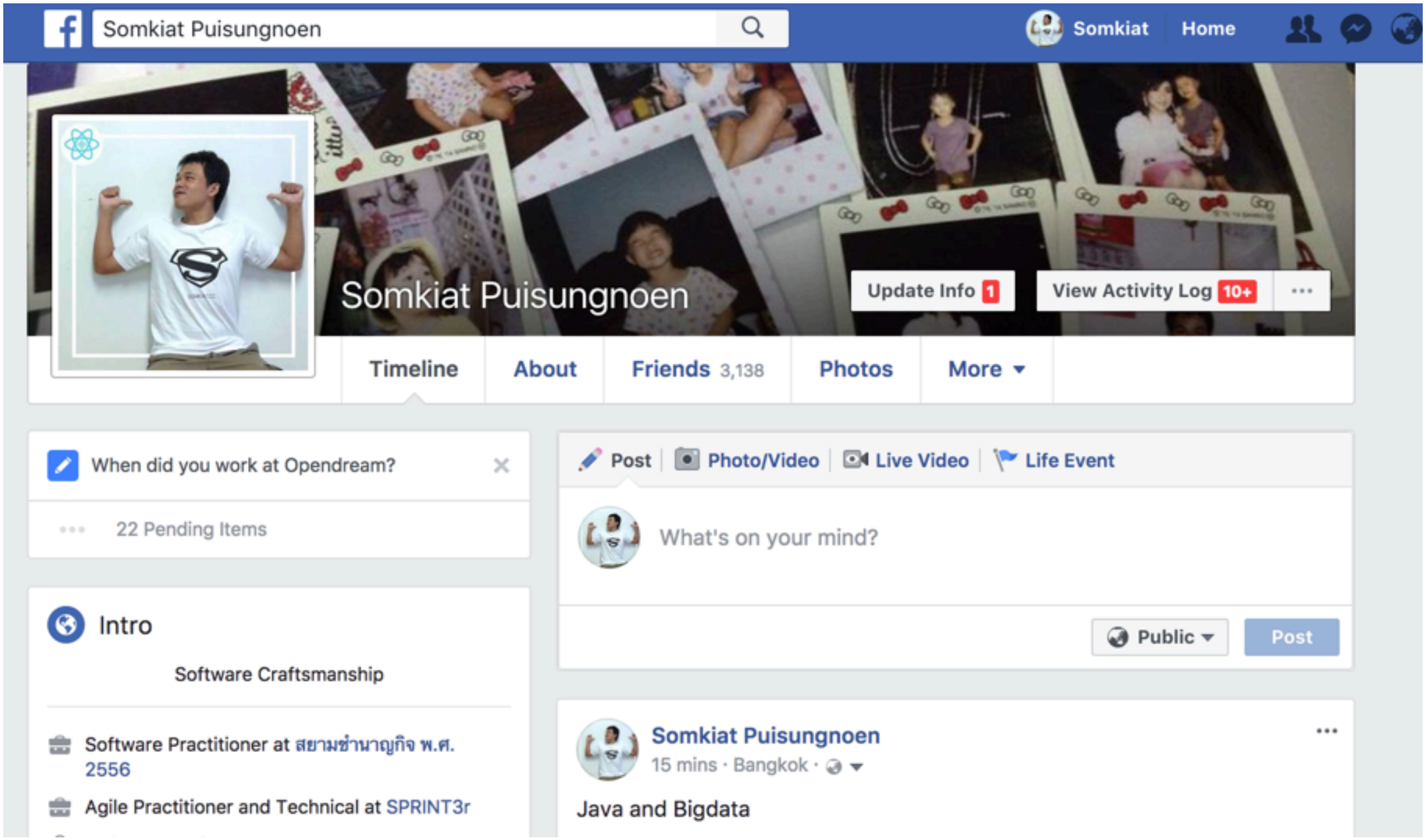
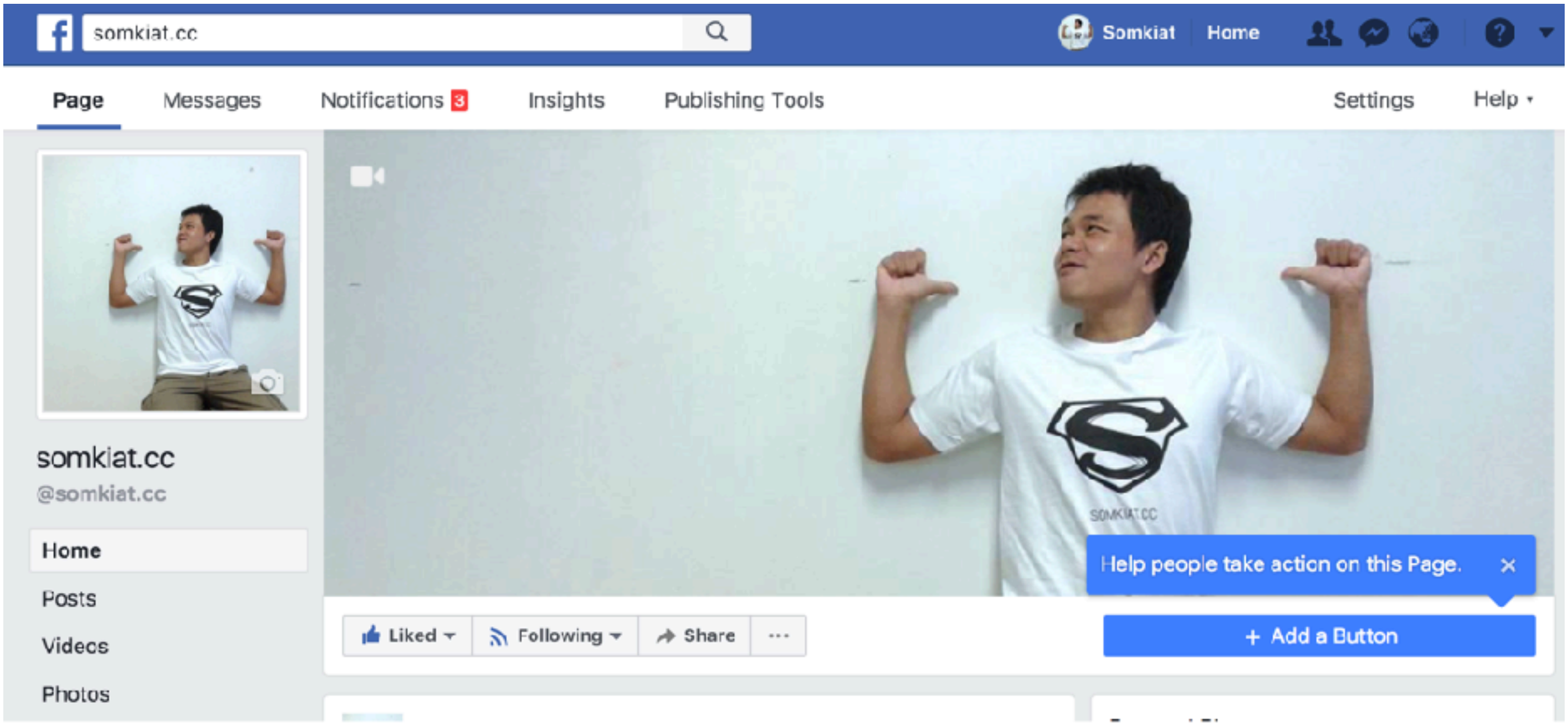
สยามชำนาญกิจ
SIAM CHAMNAN KIT

ELK Stack



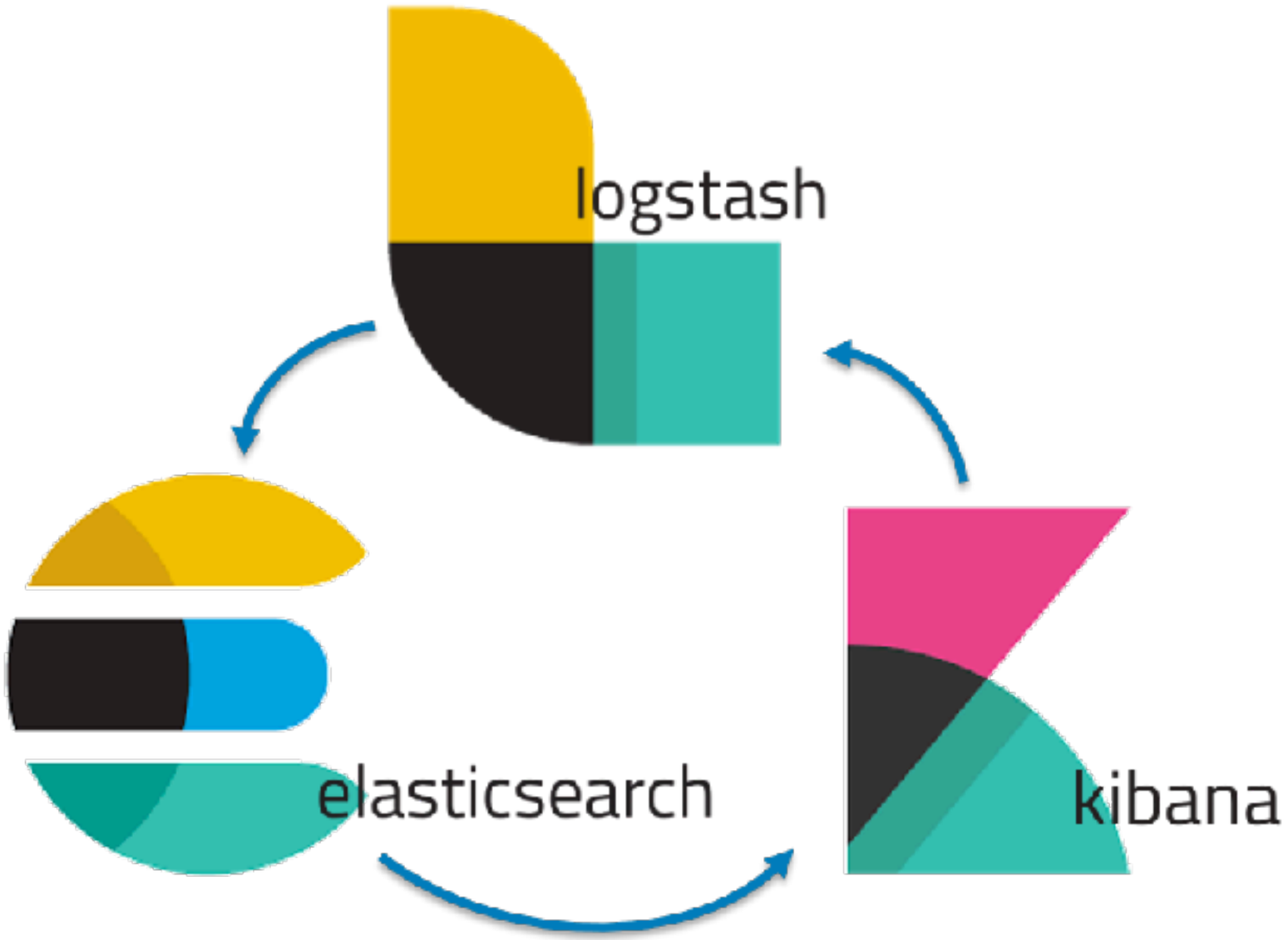


Agenda

- ELK stack
- Introduction to Elasticsearch
- CRUD (Create, Read, Update, Delete)
- Search DSL (Domain Specific Language)
- Analyzer
- Mapping
- Aggregation

Agenda

- Working with Kibana
- Useful features
  - Auto-suggestion
  - ngram algorithm
- Clustering management
- Design for scaling
- Working with Logstash



Elasticsearch ?

Elasticsearch

Search

Analytic

Real-time

Distributed

Distributed Search Engine

Open Source
Document-based
Based on Apache Lucene
JSON over HTTP

Document based

JSON (JavaScript Object Notation)
Dynamic Schema
Some relationship (nested, parent/child)

StackOverflow Question


```
{
  "items": [
    {
      "owner": {
        "reputation": 13,
        "user_id": 9796344,
        "user_type": "registered",
        "profile_image": "",
        "display_name": "Cherry",
        "link": "https://stackoverflow.com/users/9796344/cherry"
      },
      "score": 0,
      "last_activity_date": 1528986761,
      "creation_date": 1528986761,
      "post_type": "question",
      "post_id": 50859951,
      "link": "https://stackoverflow.com/q/50859951"
    }
  ],
  "has_more": false,
  "quota_max": 10000,
  "quota_remaining": 9986
}
```


https://api.stackexchange.com/docs/posts-by-ids

Ranking from DB Engine (2018)

343 systems in ranking, June 2018

Rank Jun 2018	Rank May 2018	Rank Jun 2017	DBMS	Database Model	Score Jun 2018	Score May 2018	Score Jun 2017
1.	1.	1.	Oracle	Relational DBMS	1311.25	+20.84	-40.51
2.	2.	2.	MySQL	Relational DBMS	1233.69	+10.35	-111.62
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1087.73	+1.89	-111.23
4.	4.	4.	PostgreSQL	Relational DBMS	410.67	+9.77	+42.13
5.	5.	5.	MongoDB	Document store	343.79	+1.67	+8.79
6.	6.	6.	DB2	Relational DBMS	185.64	+0.03	-1.86
7.	7.	↑ 9.	Redis	Key-value store	136.30	+0.95	+17.42
8.	↑ 9.	↑ 11.	Elasticsearch	Search engine	131.04	+0.60	+19.48
9.	↓ 8.	↓ 7.	Microsoft Access	Relational DBMS	130.99	-2.12	+4.44
10.	10.	↓ 8.	Cassandra	Wide column store	119.21	+1.38	-4.91
11.	11.	↓ 10.	SQLite	Relational DBMS	114.26	-1.19	-2.44

https://db-engines.com/en/ranking

Let’s start

Installation

Elasticsearch
Kibana

Start Elasticsearch

$./bin/elasticsearch

```
[0g8-71W] loaded module [reindex]
[0g8-71W] loaded module [repository-url]
[0g8-71W] loaded module [transport-netty4]
[0g8-71W] loaded module [tribe]
[0g8-71W] no plugins loaded
[0g8-71W] using discovery type [zen]
initialized
[0g8-71W] starting ...
[0g8-71W] publish_address {127.0.0.1:9300},
[0g8-71W] recovered [0] indices into cluster_state
ansport] [0g8-71W] publish_address {127.0.0.1:9200},
```

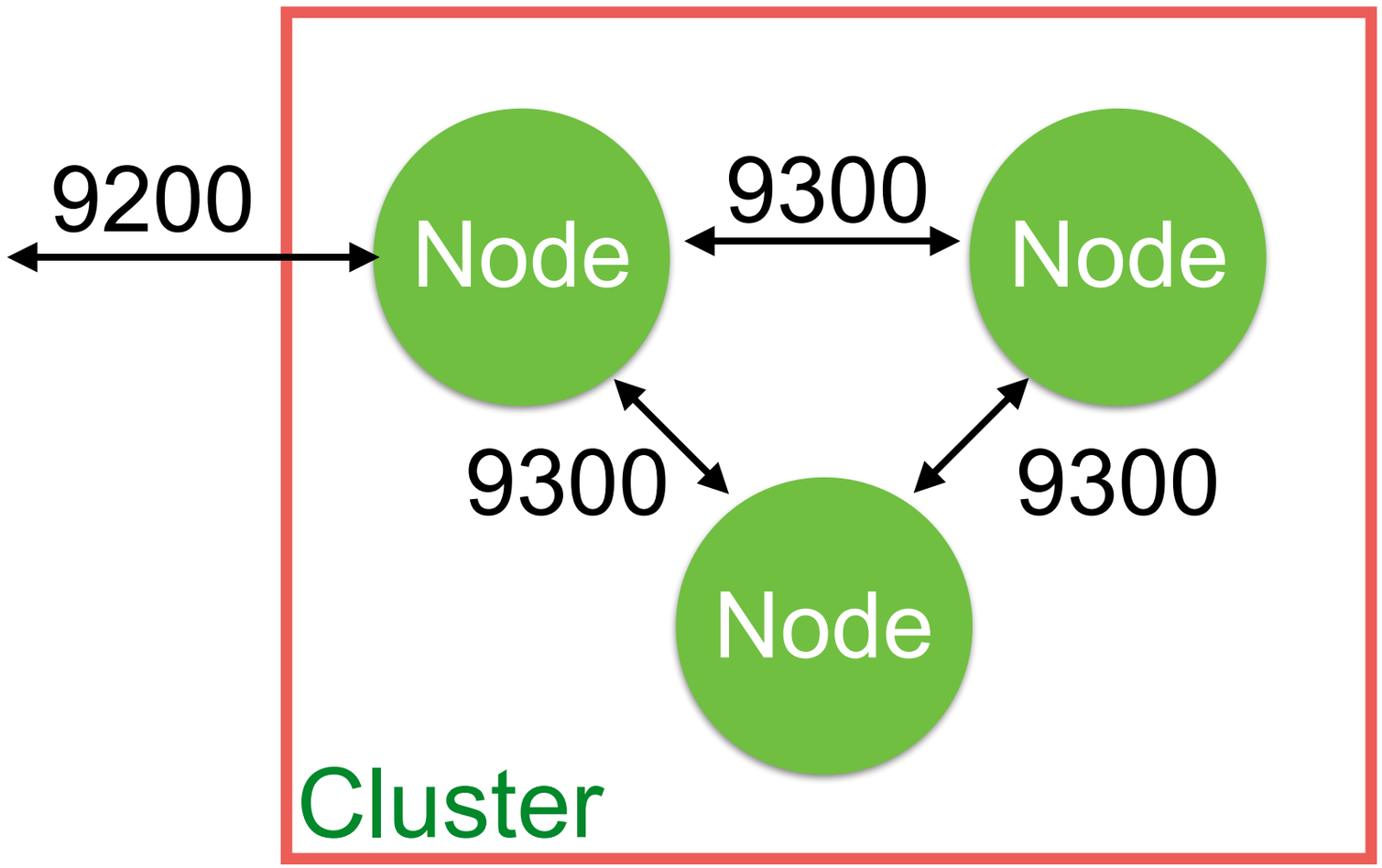
Hello Elasticsearch

http://localhost:**9200**/


```
{
    name: "0g8-71W",
    cluster_name: "elasticsearch",
    cluster_uuid: "DDqm23ZPR1q_wves145sTw",
  - version: {
        number: "6.2.4",
        build_hash: "ccec39f",
        build_date: "2018-04-12T20:37:28.497551Z",
        build_snapshot: false,
        lucene_version: "7.2.1",
        minimum_wire_compatibility_version: "5.6.0",
        minimum_index_compatibility_version: "5.0.0"
    },
    tagline: "You Know, for Search"
}
```


Ports of Elasticsearch

RESTful API with JSON Over HTTP (9200)
Java API (9300)



Health of cluster

http://localhost:9200/_cluster/health


```
{
    cluster_name: "elasticsearch",
    status: "green",
    timed_out: false,
    number_of_nodes: 1,
    number_of_data_nodes: 1,
    active_primary_shards: 0,
    active_shards: 0,
    relocating_shards: 0,
    initializing_shards: 0,
    unassigned_shards: 0,
    delayed_unassigned_shards: 0,
    number_of_pending_tasks: 0,
    number_of_in_flight_fetch: 0,
    task_max_waiting_in_queue_millis: 0,
    active_shards_percent_as_number: 100
}
```


Health of cluster

Status	Meaning
Green	All shards are allocated
Yellow	Primary shard is allocated, but replicas are not
Red	Shard not allocated in the cluster

cat APIs

http://localhost:9200/_cat

```
=^.^=
/_cat/allocation
/_cat/shards
/_cat/shards/{index}
/_cat/master
/_cat/nodes
/_cat/tasks
/_cat/indices
/_cat/indices/{index}
/_cat/segments
/_cat/segments/{index}
/_cat/count
/_cat/count/{index}
/_cat/recovery
/_cat/recovery/{index}
/_cat/health
/_cat/pending_tasks
/_cat/aliases
/_cat/aliases/{alias}
```

https://www.elastic.co/guide/en/elasticsearch/reference/current/cat.html

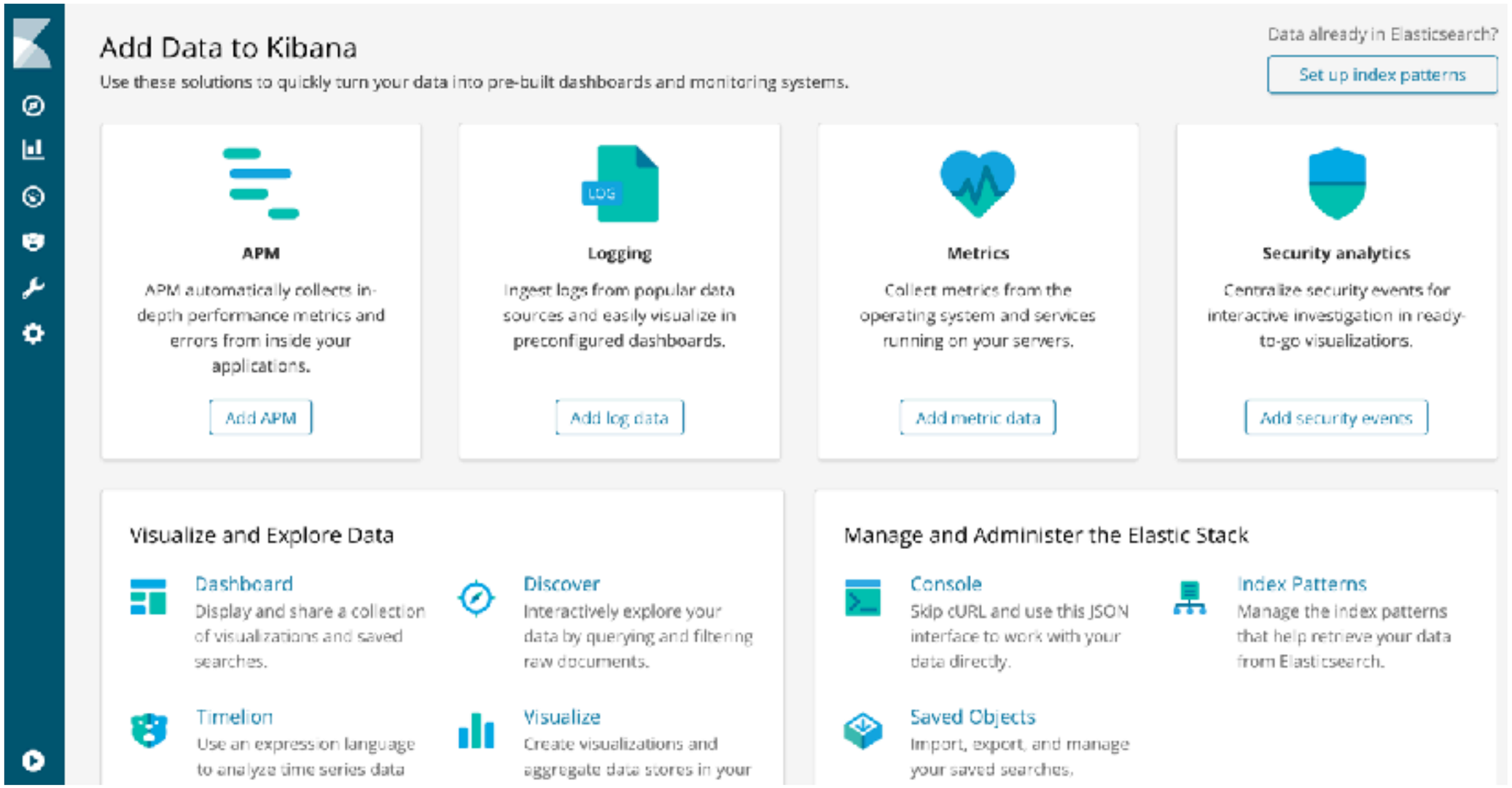
Start Kibana

$./bin/kibana

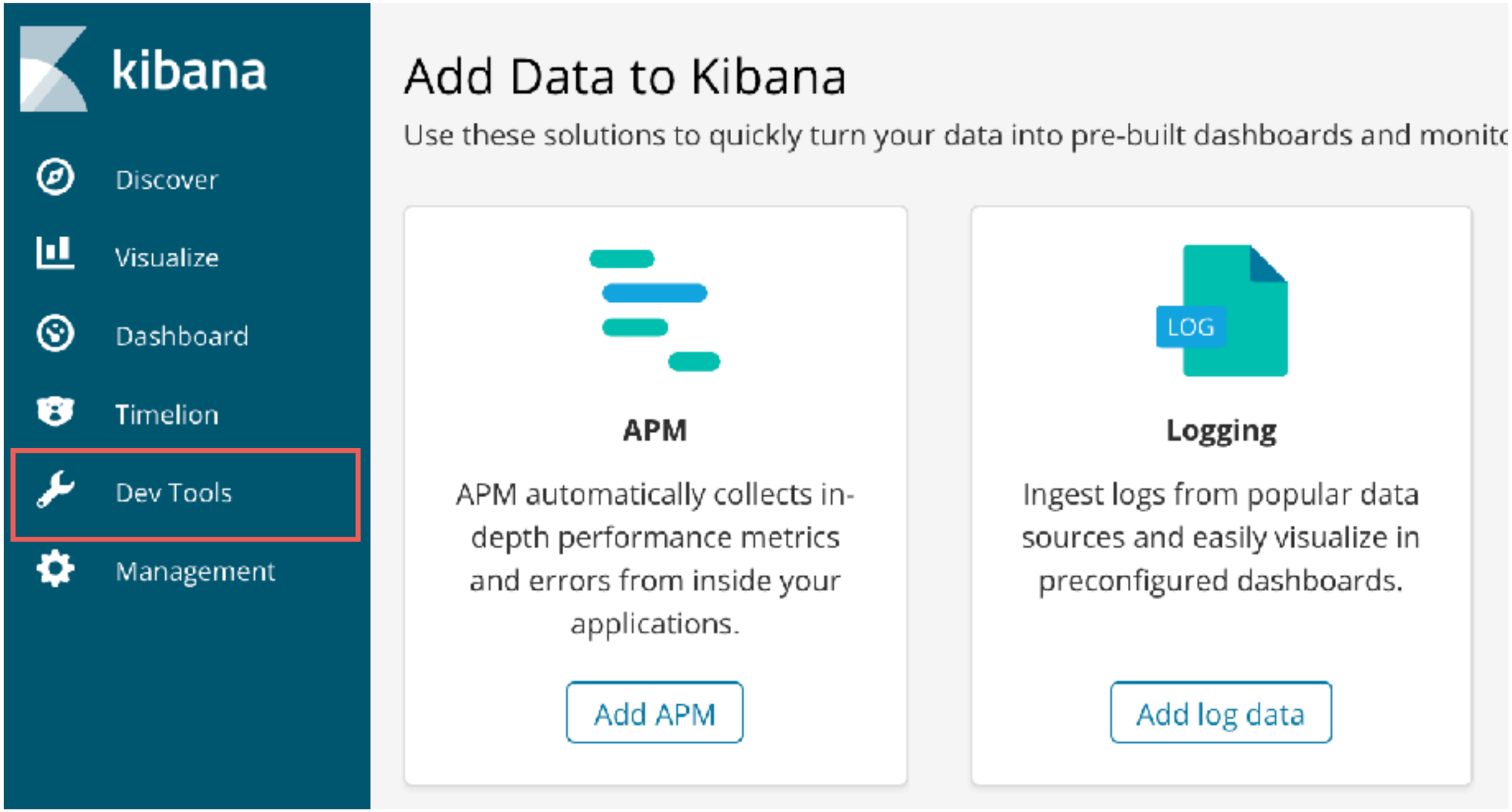
```
 [info][status][plugin:kibana@6.2.4] Status changed from uninit

 [info][status][plugin:elasticsearch@6.2.4] Status changed from
asticsearch
 [info][status][plugin:timelion@6.2.4] Status changed from unin

 [info][status][plugin:console@6.2.4] Status changed from unini

 [info][status][plugin:metrics@6.2.4] Status changed from unini

 [info][listening] Server running at http://localhost:5601
 [info][status][plugin:elasticsearch@6.2.4] Status changed from
```

Hello Kibana

http://localhost:**5601**/



Using Dev Tools



Ready to start

CRUD with Elasticsearch

Create document

Read document

Update document

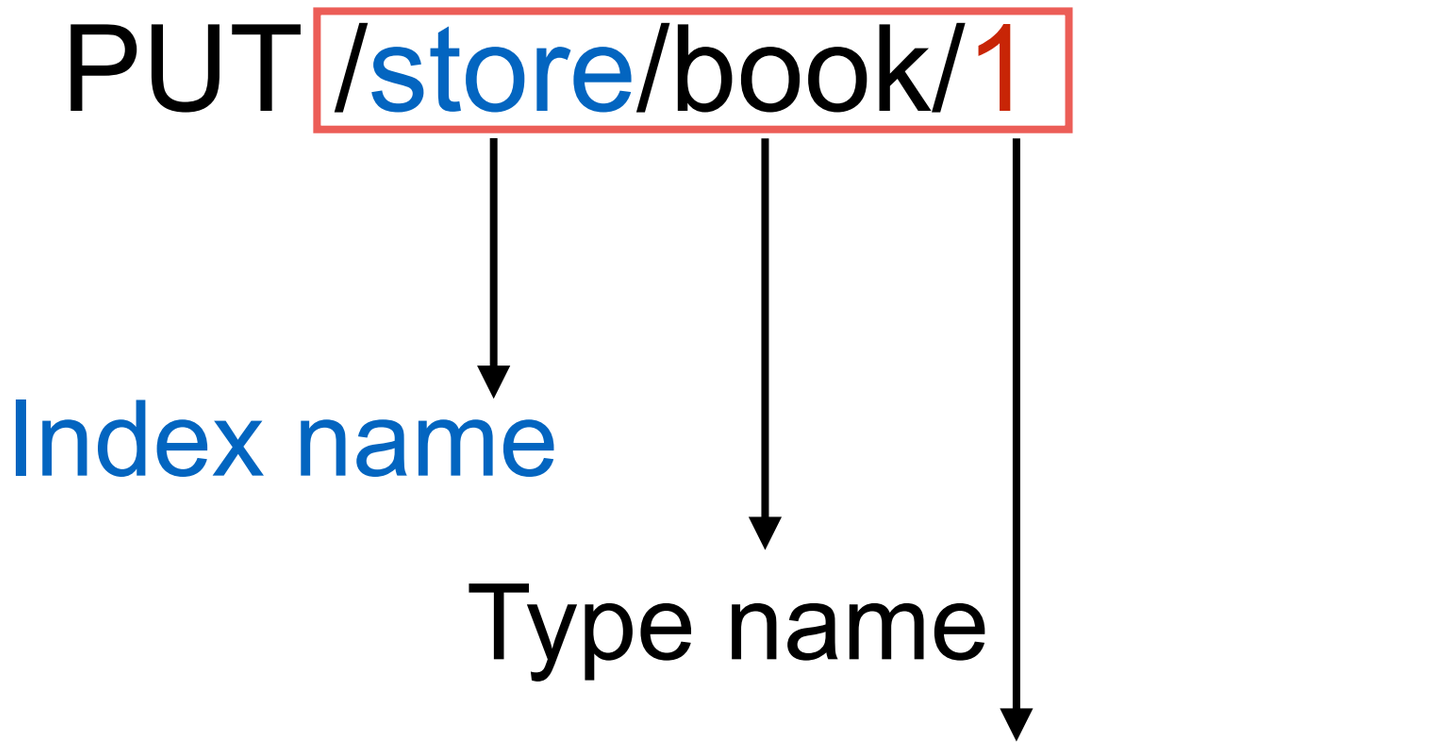
Delete document

Create a document

PUT /store/book/1

```
{
  "title": "Elasticsearch: The Definitive Guide",
  "author_name": [
    "Clinton Gormley",
    "Zachary Tong"
  ],
  "tag": [
    "search",
    "computer"
  ],
  "isbn-13": "978-1449358549",
  "isbn-10": "1449358543",
  "price": 44.3,
  "page": 724,
}
```

Create document



Compare with RDBMS

Database	Table	Row	Column
Index	Type*	Document	Field

** Only 1 type per index*

Read document

GET /store/book/1


```
{
  "_index": "store",
  "_type": "book",
  "_id": "1",
  "_version": 1,
  "found": true,
  "_source": {
    "title": "Elasticsearch: The Definitive Guide",
    "author_name": [
      "Clinton Gormley",
      "Zachary Tong"
    ],
    "tag": [
      "search",
      "computer"
    ]
  }
```


Information of document

Delete document

DELETE /store/book/1


```
{
  "_index": "store",
  "_type": "book",
  "_id": "1",
  "_version": 2,
  "result": "deleted",
  "_shards": {
    "total": 2,
    "successful": 1,
    "failed": 0
  },
  "_seq_no": 1,
  "_primary_term": 1
}
```


More features

Update by query
Delete by query
Partial update document

Workshop

02-crud/book_document.json

Bulk API

03-bulk/book_bulk.json

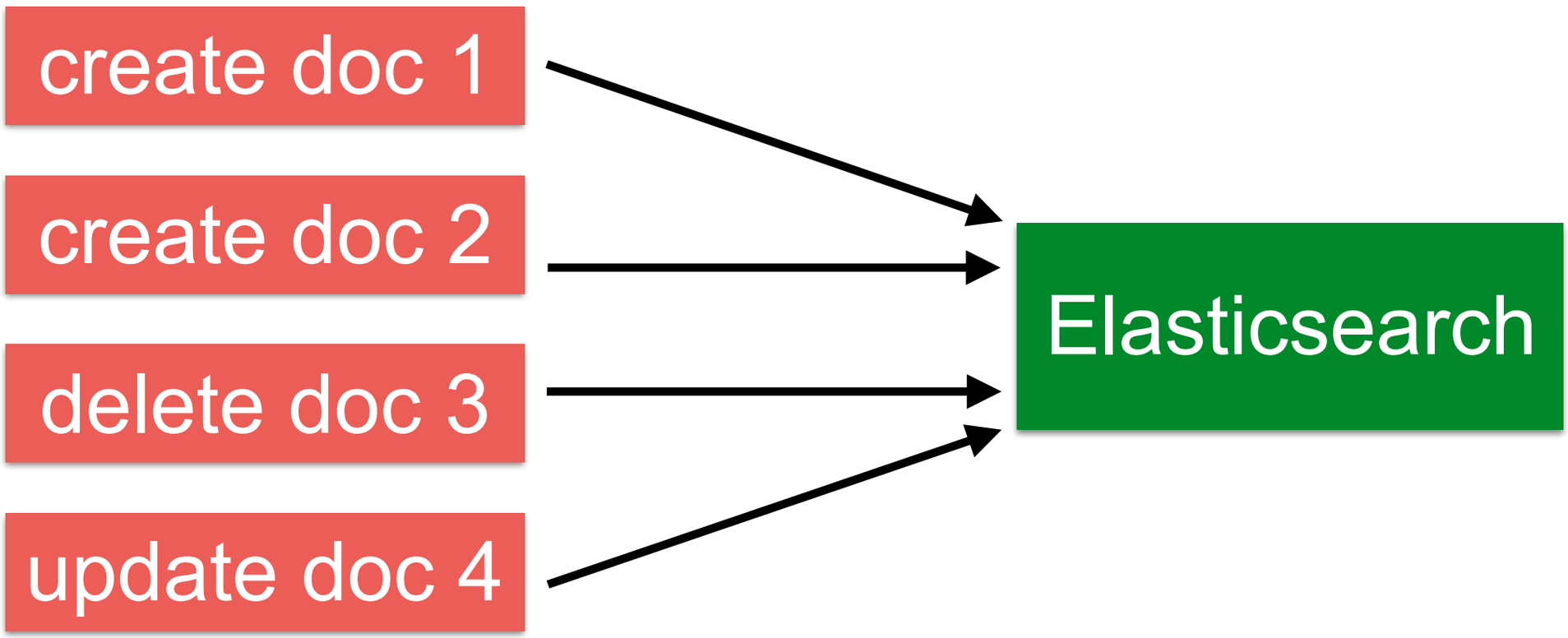
https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-bulk.html

Bulk API

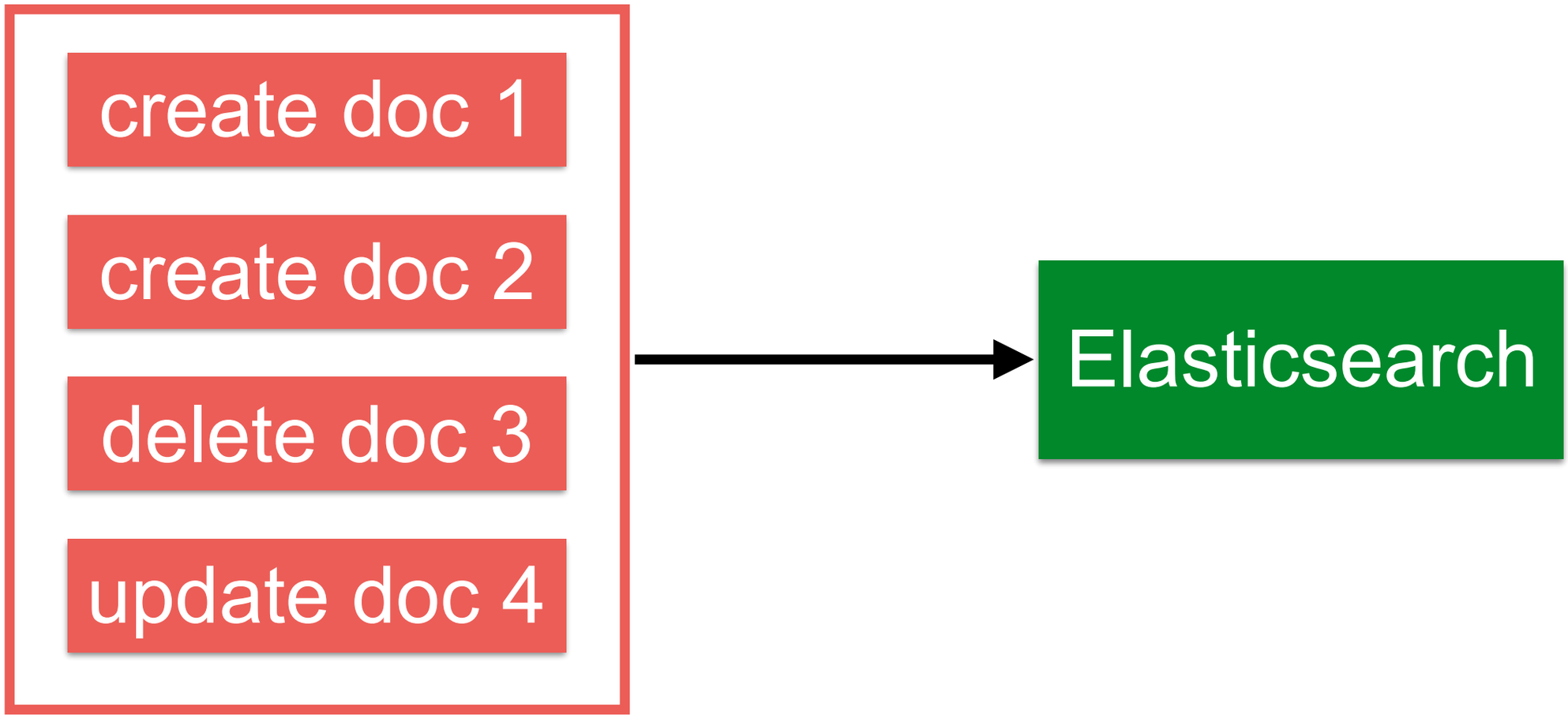
Perform many index/delete operation in single API call

Increase indexing speed

Without Bulk API

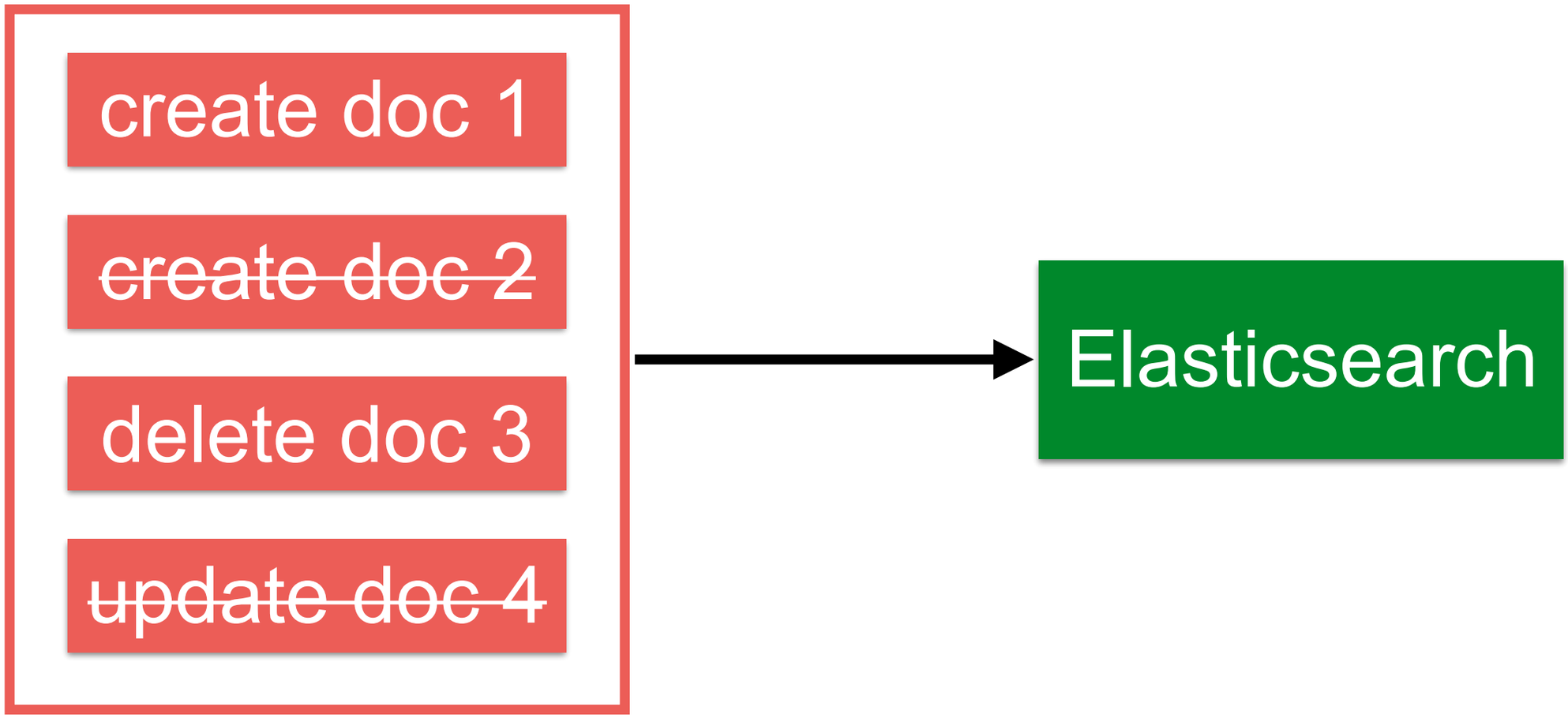


With Bulk API



Store in memory 5-15 MB

No transaction in bulk api



Create a document

POST /store/book/**_bulk**

```
{"create":{"_id":"1001"}}
{"title":"new book 1000","description":"my new book"}
```

Response from Bulk API

```
{
  "took": 89,
  "errors": false,
  "items": [
    {
      "create": {
        "_index": "store",
        "_type": "book",
        "_id": "1001",
        "_version": 1,
        "result": "created",
        "_shards": {
          "total": 2,
          "successful": 1,
          "failed": 0
        },
        "_seq_no": 0,
        "_primary_term": 1,
        "status": 201
```

Time in milliseconds

HTTP Status 201 = Created

Search API

04-search/book_search.json

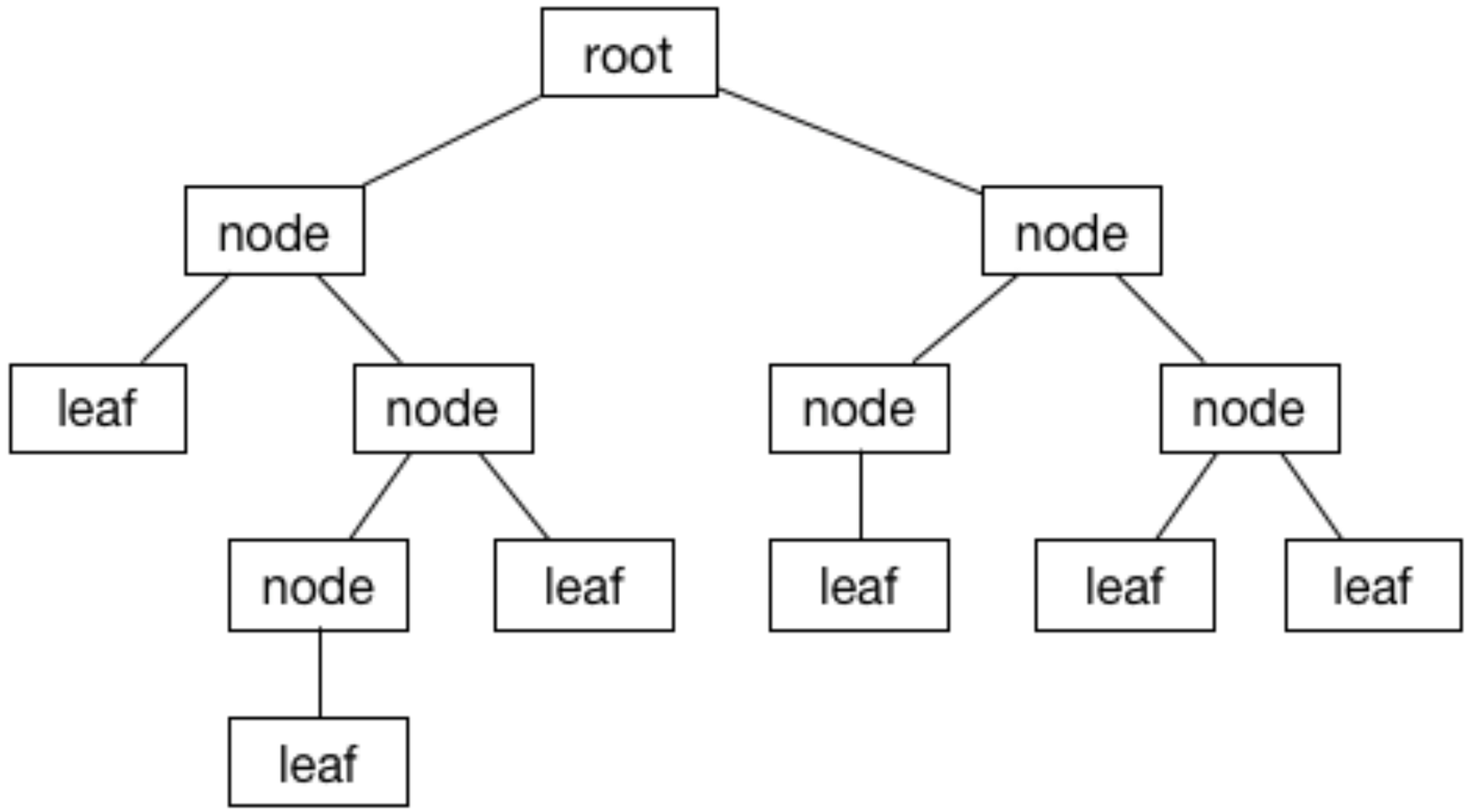
Query DSL

Domain Specific Language for query data
Flexible query language
Based on JSON format

https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl.html

Query DSL

1. Leaf query clause
2. Compound query clause



Query DSL

Query (unstructured data)
Filter (structured data)

Query DSL

Query	Filter
Relevance	Boolean, yes/no
Full text search	Exact values
Not cached	Cached
Slower	Faster

***Filter first**, then query remaining documents*

Leaf query clause

GET /store/book/_search

```
{
    "query": {
      "match_all": {}
    }
}
```

Compound query clause

GET /store/book/_search

```
{
   "query": {
      "bool": {
         "must": [{}],
         "should": [{}],
         "must_not": [{}]
      }
   }
}
```

Aggregation API

05-aggregation/book_aggregation.json

https://www.elastic.co/guide/en/elasticsearch/reference/current/search-aggregations.html

```
SELECT count(1), sum(price)
FROM some_table
GROUP BY some_column
```

Aggregation Types

Bucketing

Metric

Matrix

Pipeline

Structure

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

Count by category

GET /store/book/_search

```
{
 "aggs": {
    "all_book_title": {
      "terms": {
        "field": "category.keyword"
      }
    }
  }
}
```

Count by category

GET /store/book/_search

```
{
 "aggs": {
    "all_book_title": {
       "terms": {
          "field": "category.keyword"
       }
    }
 }
}
```

Aggregation name

Count by category

GET /store/book/_search

```
{
 "aggs": {
    "all_book_title": {
      "terms": {
        "field": "category.keyword"
      }
    }
  }
}
```

Aggregation type

Result of aggregation

Search result


```
{
  "hits": {
    "total": 5,
    "max_score": 1,
    "hits": [
      {
        "_source": {
          "title": "The Logstash Book"
        }
      },
      {
        "_source": {
          "title": "Elasticsearch Server: Second Edition"
        }
      }
    ]
  }}
```


Result of aggregation

```
"aggregations": {
  "all_book_title": {
    "doc_count_error_upper_bound": 0,
    "sum_other_doc_count": 0,
    "buckets": [
      {
        "key": "Computer & Technology",
        "doc_count": 5
      },
      {
        "key": "Online Searching",
        "doc_count": 3
      },
      {
        "key": "Java Programming",
        "doc_count": 2
      }
]
```

Aggregation result

Show only aggregation result

GET /store/book/_search

```
{
  "size": 0,
  "aggs": {
    "all_book_title": {
      "terms": {
        "field": "category.keyword"
      }
    }
  }
}
```

Set search result size = 0

Range of price

GET /store/book/_search

```
{
  "size": 0,
  "aggs": {
    "price_range": {
      "range": {
        "field": "price",
        "ranges": [
          { "from": 0,"to": 10 },
          { "from": 11,"to": 20 },
          { "from": 21,"to": 50 }
        ]
      }
    }
  }
}
```

Result of aggregation


```
"buckets": [
    {
      "key": "0.0-10.0",
      "from": 0,
      "to": 10,
      "doc_count": 1
    },
    {
      "key": "11.0-20.0",
      "from": 11,
      "to": 20,
      "doc_count": 0
    },
    {
      "key": "21.0-50.0",
      "from": 21,
      "to": 50,
      "doc_count": 3
    }
```


Range of price and ordering

GET /store/book/_search

```
{
  "size": 0,
  "aggs": {
    "price_range": {
      "range": {
        "field": "price",
        "ranges": [
          { "from": 0,"to": 10 },
          { "from": 11,"to": 20 },
          { "from": 21,"to": 50 }
        ]
      }
    }
  }
}
```

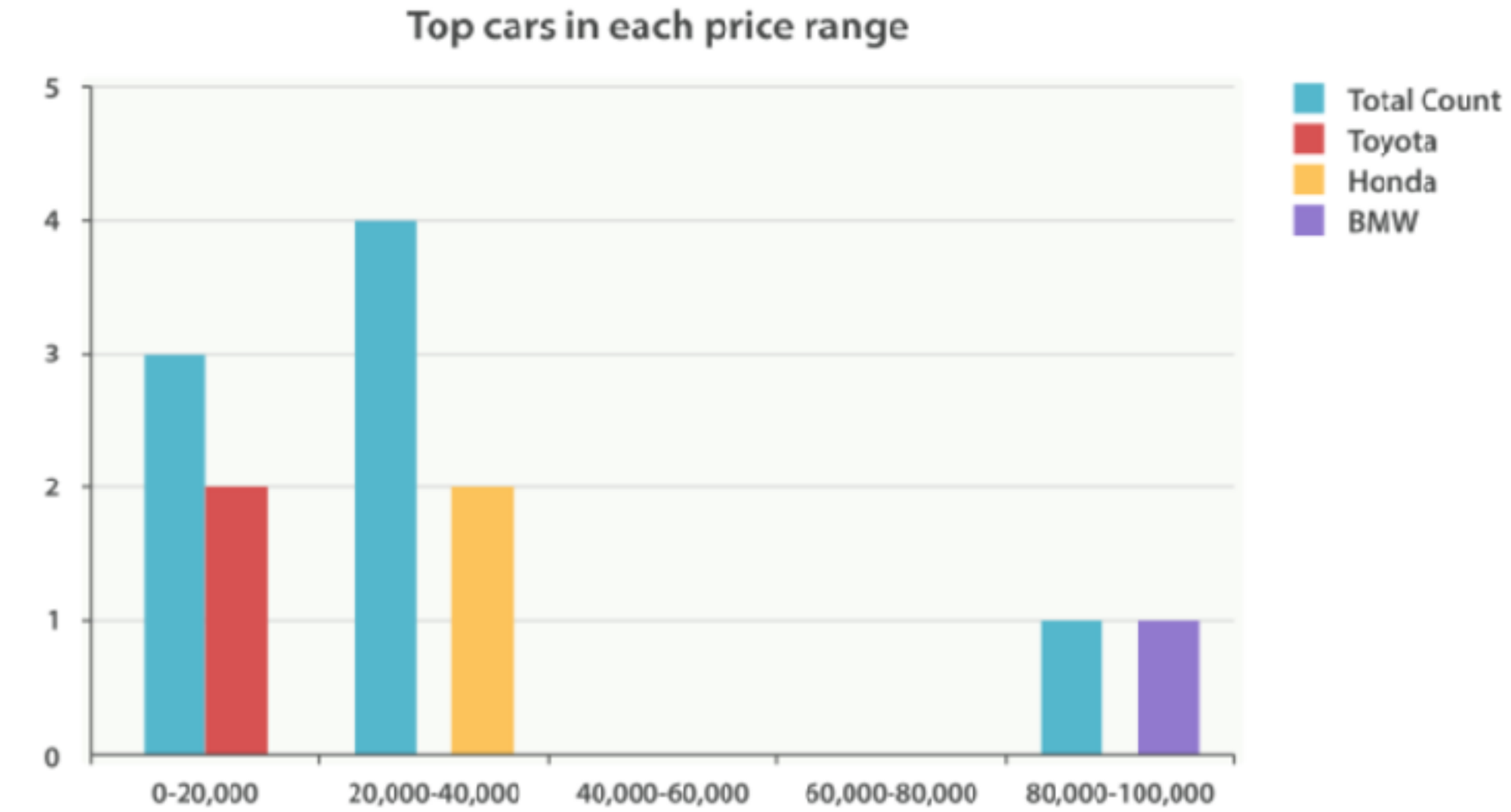
Workshop aggregation with car

05-aggregation/car.json

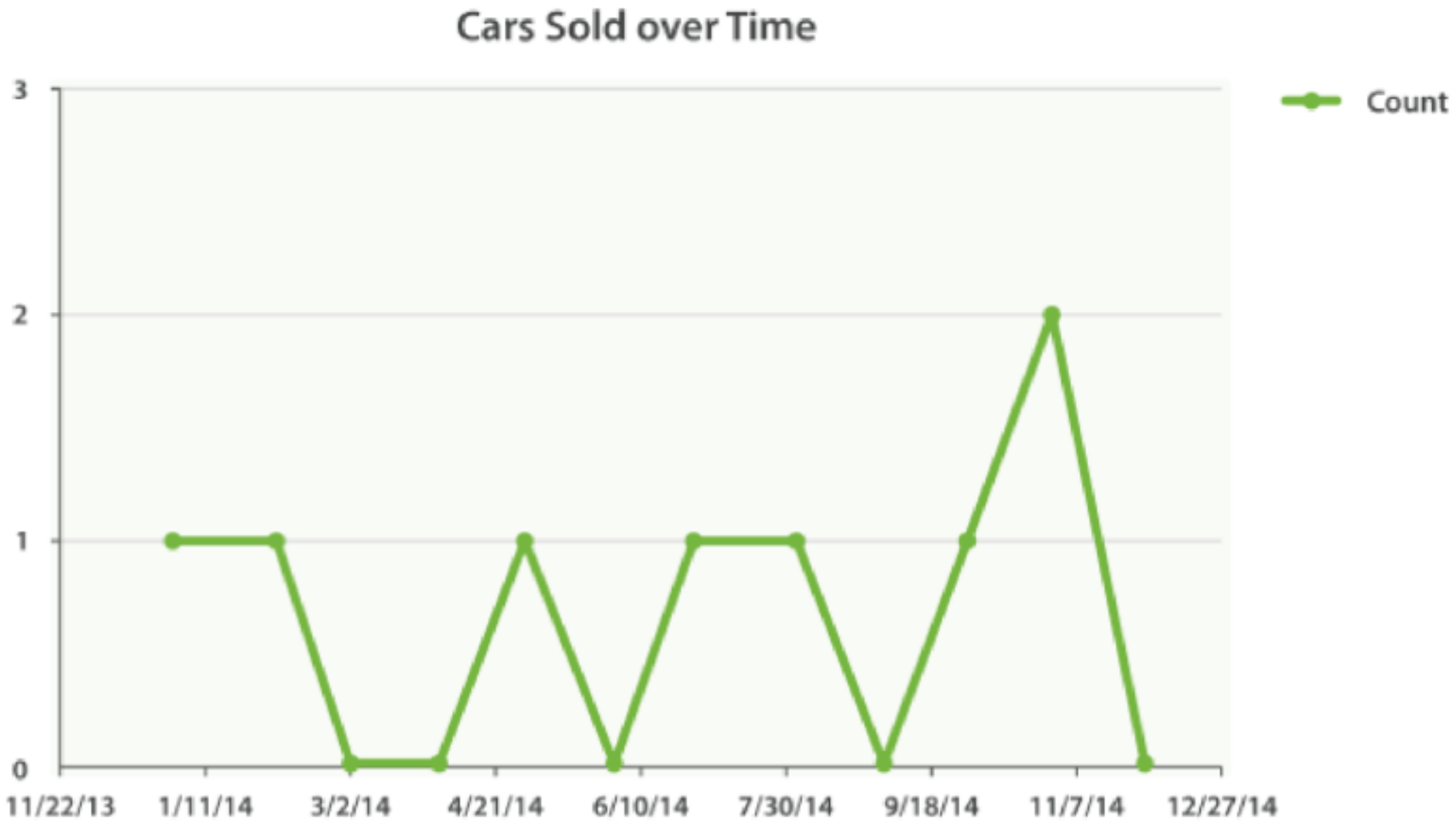
Try by yourself

Best seller by color
Statistic of best seller by color
Detail of car in each color
Min/max of price by make

Top cars in each price range ?



Cars sold over Time ?



Mapping

https://www.elastic.co/guide/en/elasticsearch/reference/current/mapping.html

Mapping type

Meta-fields
Field or properties

Meta-field

Metadata of document
_index, _type, _id, _source

Field or properties

List of fields or properties of document

Mapping/Schema of document

GET /store/_mapping/book

```
"mappings": {
      "book": {
        "properties": {
          "author_name": {
            "type": "text",
          "fields": {
            "keyword": {
              "type": "keyword",
              "ignore_above": 256
            }
          }
      }
}
```

Mapping/Schema of document

GET /store/_mapping/book


```
"mappings": {
      "book": {
          "page": {
            "type": "long"
          },
          "price": {
            "type": "float"
          },
          "published_date": {
            "type": "date"
          }
      }
  }
}
```

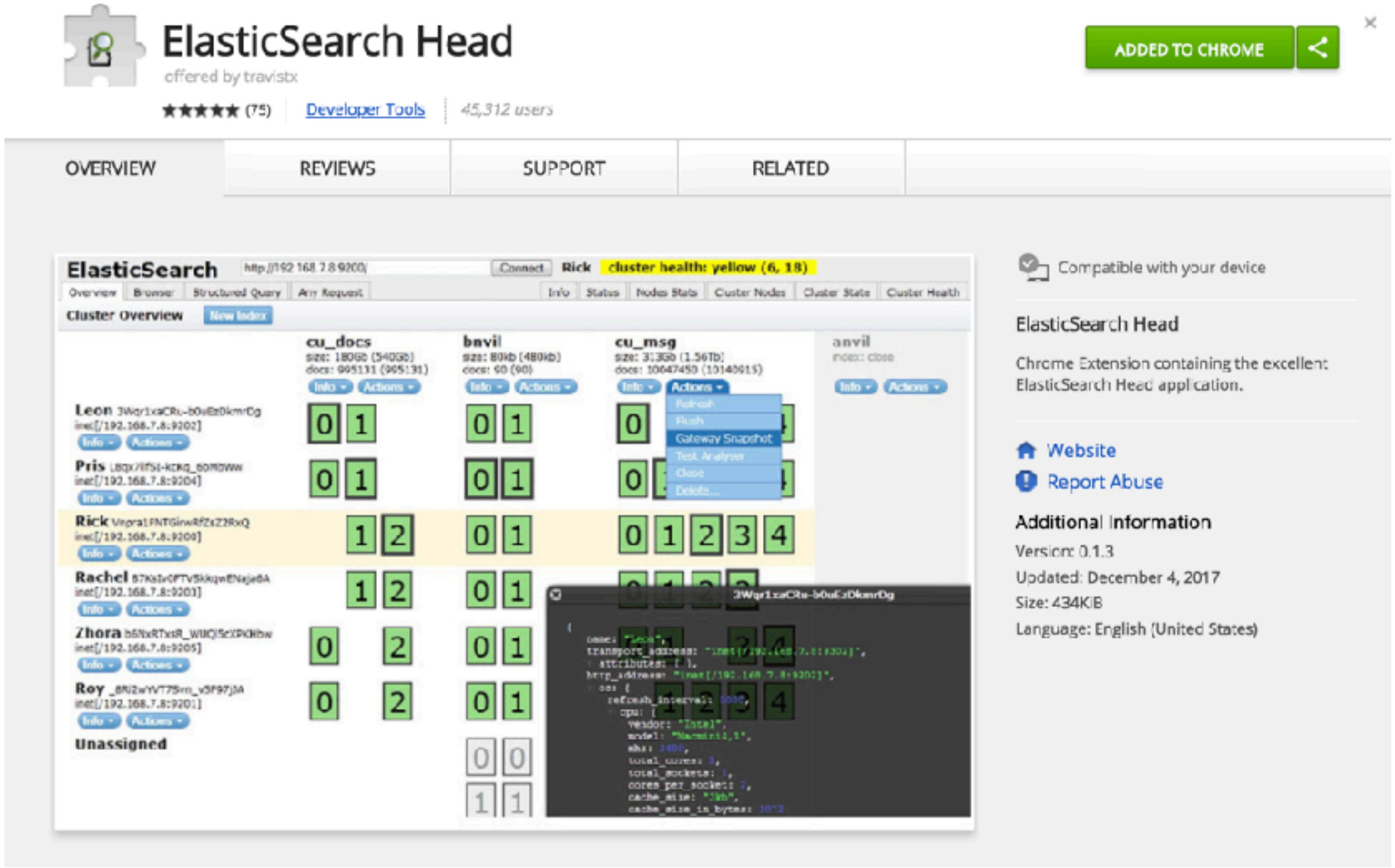

Field Datatypes

text	date
keyword	ip
long	boolean
double	completion
geo_point	geo_shape

Analyzer

More tools

Elasticsearch Head



Elasticsearch Dump

Kibana

Day 2

Geo Location

08-geo-location/sample_geo.json

Geo Location Type

Geo-point
Geo-shape

Geo-point

Must pre-define in mapping of index

```
PUT /my_map
{
  "mappings": {
    "city": {
      "properties": {
"name": {
  "type": "text"
},
"location": {
  "type": "geo_point"
}
```

Geo-point Format

Geo-point as object

Geo-point as string

Geo-point as array

Geo-point as geohash

https://www.elastic.co/guide/en/elasticsearch/reference/current/geo-point.html

Geo-point Format

Type	Format
Object	lat = lon =
String	lat, lon
Array	[lon, lat] ** GeoJSON **

https://en.wikipedia.org/wiki/GeoJSON

Geohash converter

http://geohash.co/

Geo-point query

Geo-bounding-box
Geo-distance
Geo-polygon

https://www.elastic.co/guide/en/elasticsearch/reference/current/geo-queries.html

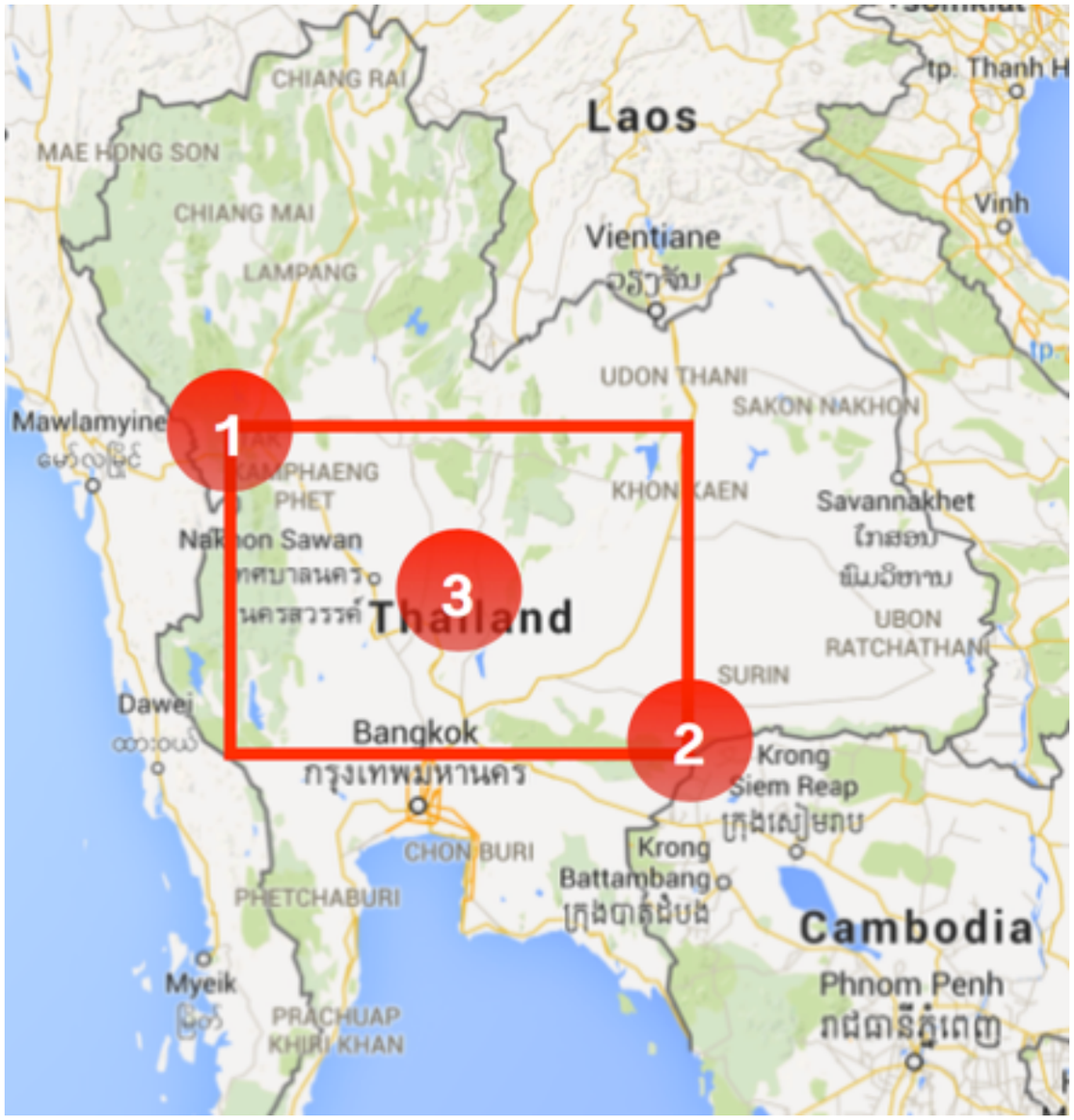
Bounding Box

https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-geo-bounding-box-query.html

Geo Distance

https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-geo-distance-query.html

Try to ordering result



Explain and Profiling your query

2 ways

Explain API

Profile API

Explain API

```
GET /my_map/_search
{
  "explain": true,
  "query": {
    "bool": {
```

https://www.elastic.co/guide/en/elasticsearch/reference/6.3/search-explain.html

Profile API

Debugging tool

Add overhead to search execution

Output is verbose and depend on internal operation

https://www.elastic.co/guide/en/elasticsearch/reference/6.3/search-profile.html

Profile API

```
GET /my_map/_search
{
  "profile": true,
  "query": {
    "bool": {
```

Working with Data

https://www.elastic.co/guide/en/kibana/current/tutorial-load-dataset.html

Working with Data

$elasticsearch-plugin install **ingest-geoip**

https://www.elastic.co/guide/en/elasticsearch/plugins/current/ingest-geoip.html

GeoIP with Elasticsearch

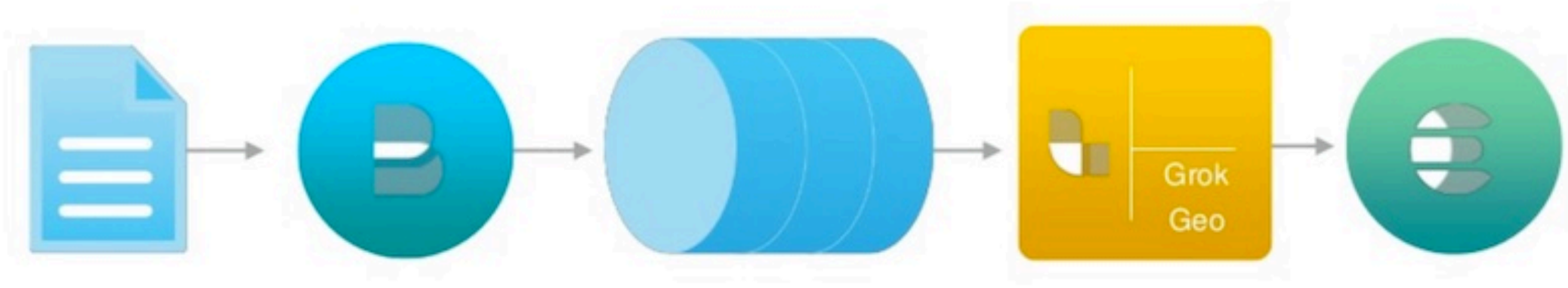
geoip/instruction.json

Sample Data

```
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T09:03:25.877Z","ip":"185.124.182.12
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T12:28:25.013Z","ip":"79.1.14.87","e
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T17:44:34.357Z","ip":"178.209.1.7","
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T13:04:18.120Z","ip":"118.140.92.127
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T11:37:40.653Z","ip":"235.154.34.221
{"index":{"_index":"logstash-2015.05.18","_type":"log"}}
{"@timestamp":"2015-05-18T08:46:07.025Z","ip":"228.216.38.41"
```

Working with Ingest

Pre-process document before actual indexing



https://www.elastic.co/guide/en/elasticsearch/reference/current/ingest.html

Install plugin

$elasticsearch-plugin install **ingest-geoip**

https://www.elastic.co/guide/en/elasticsearch/plugins/current/ingest-geoip.html

Working with Logstash

https://www.elastic.co/guide/en/logstash/current/index.html

Logstash

Input

Filter

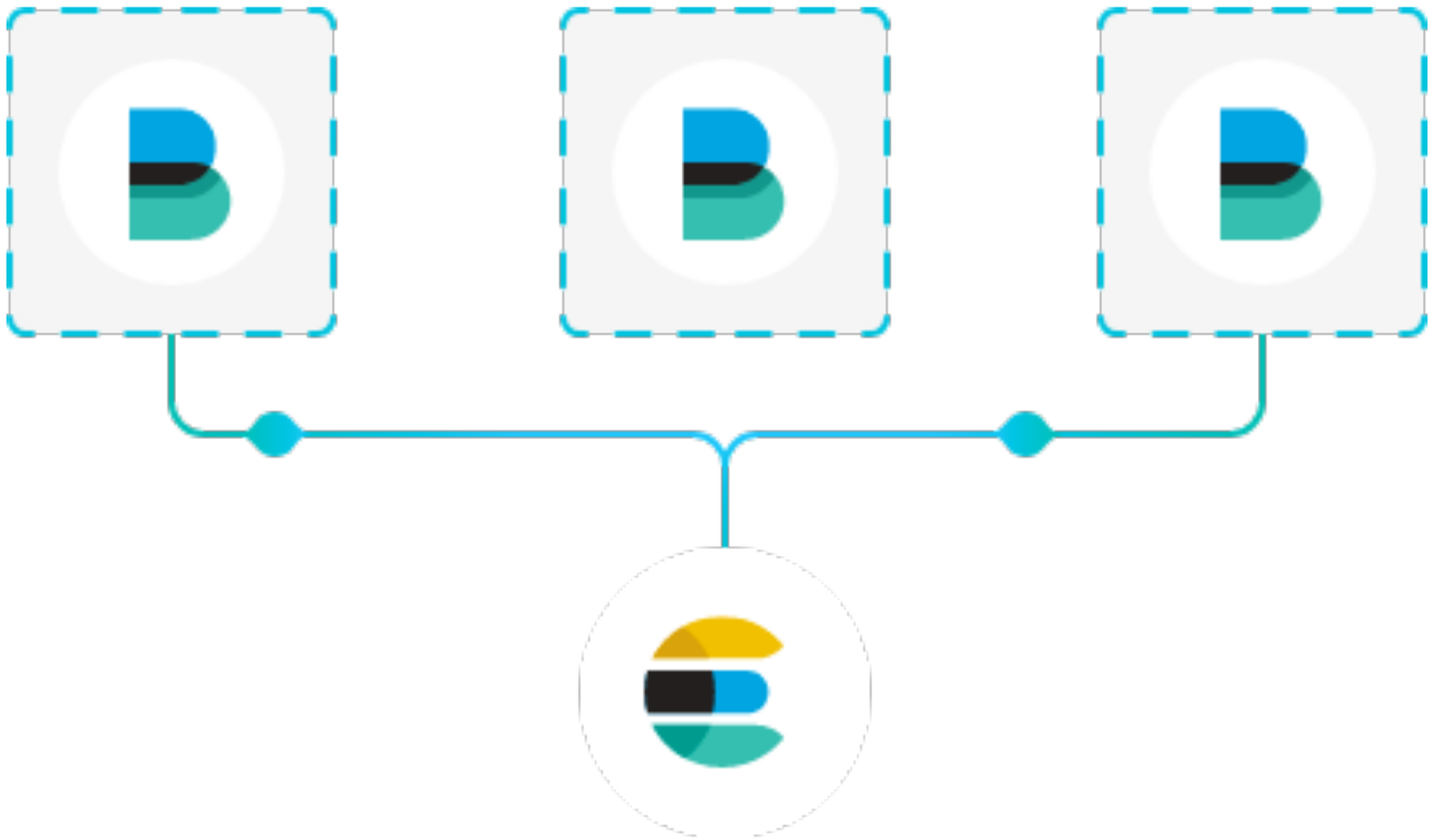
Output

Design your input first !!

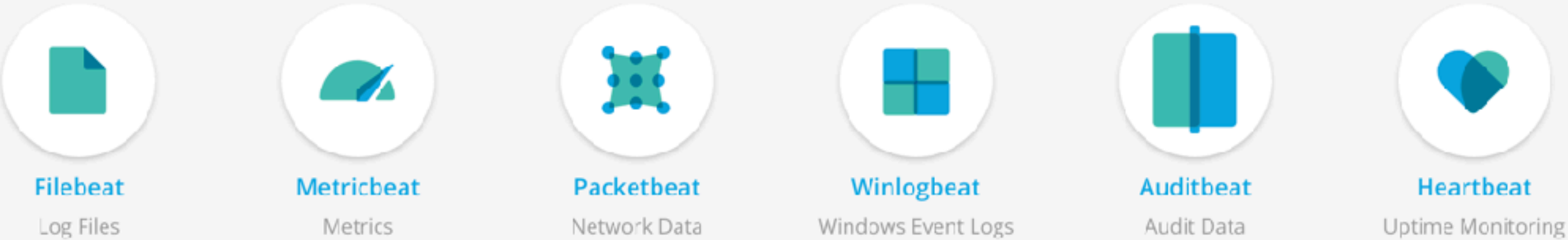
Use beats is better

https://www.elastic.co/products/beats

Beat



Beat

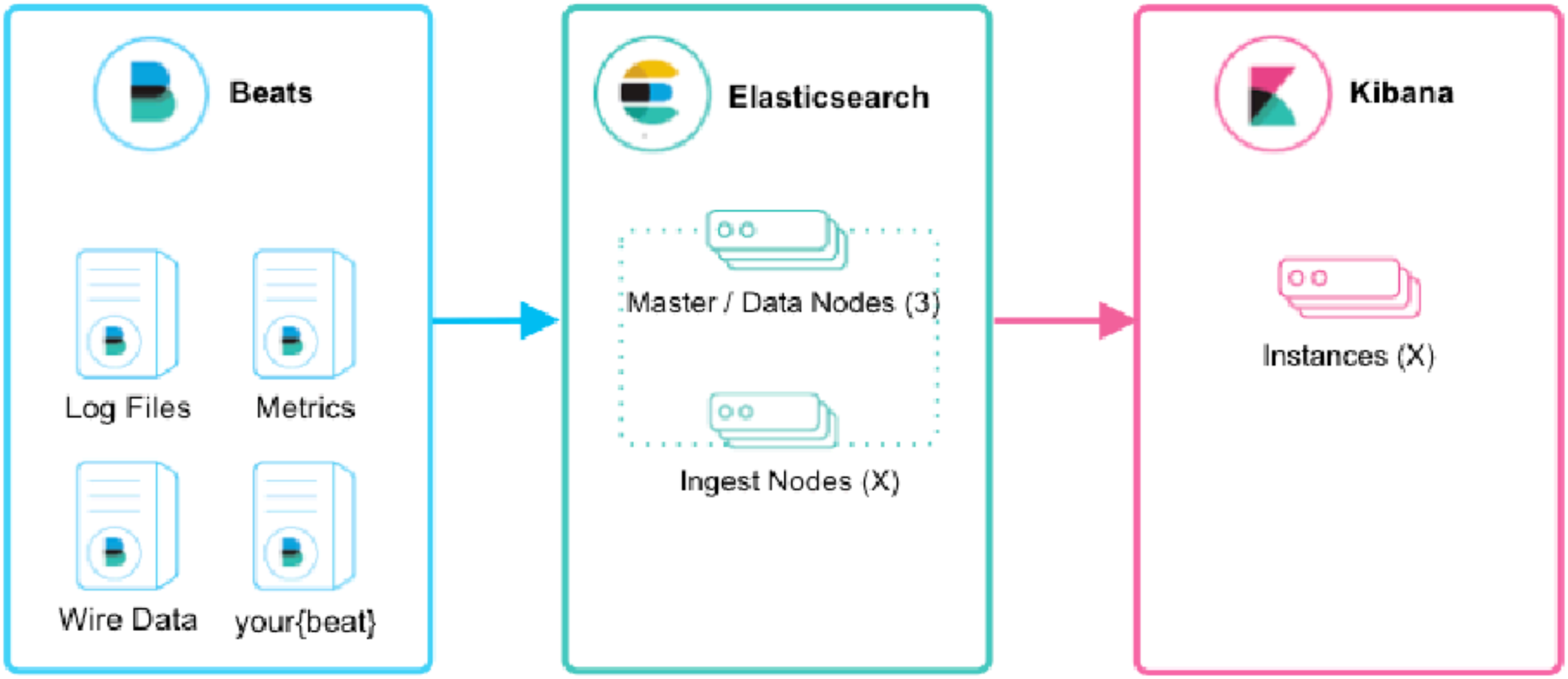


Example

$filebeat -e -c **beat.yml** -d "publish"

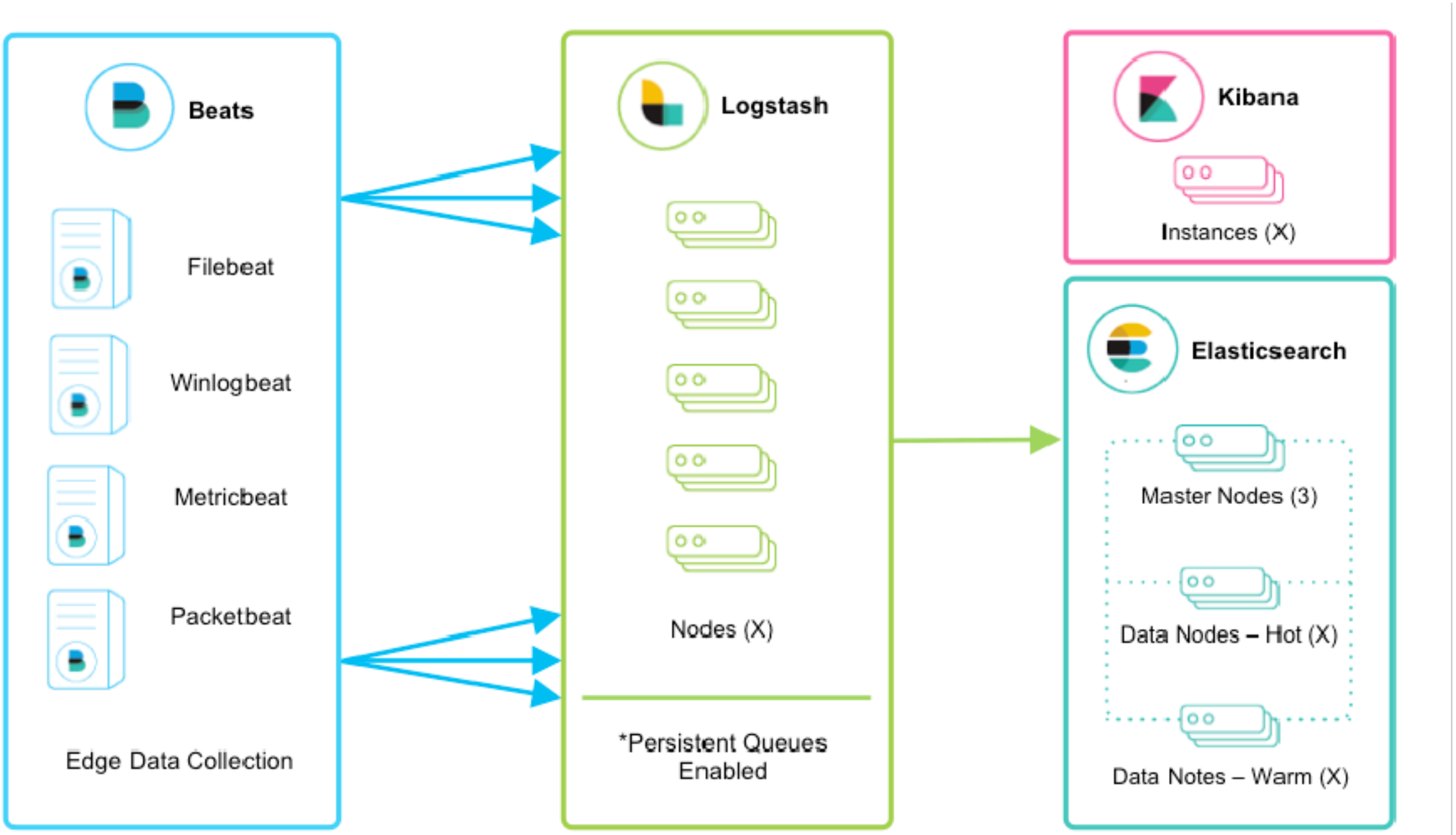
https://www.elastic.co/guide/en/beats/filebeat/current/index.html

Beat and Logstash



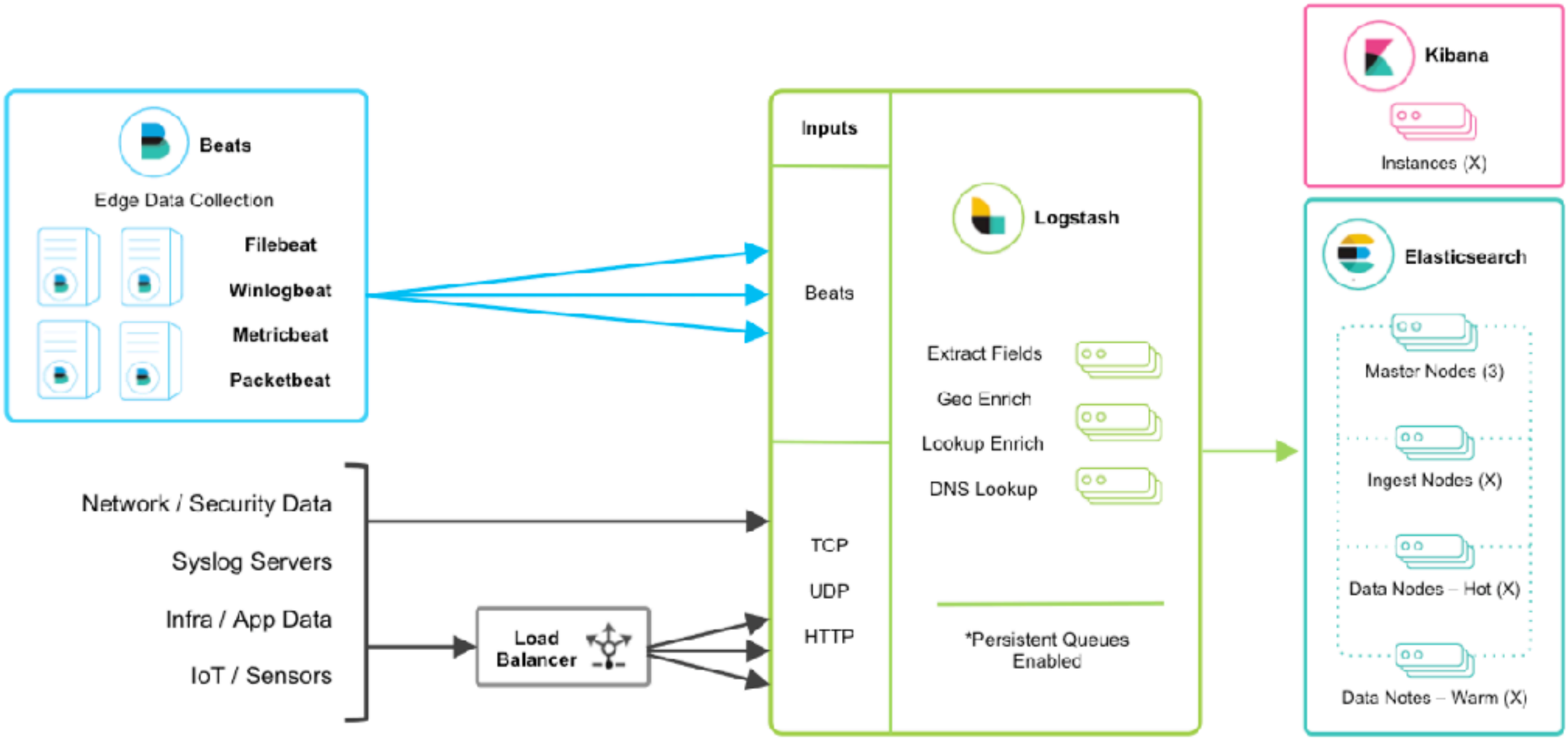
https://www.elastic.co/guide/en/logstash/current/deploying-and-scaling.html

Scaling



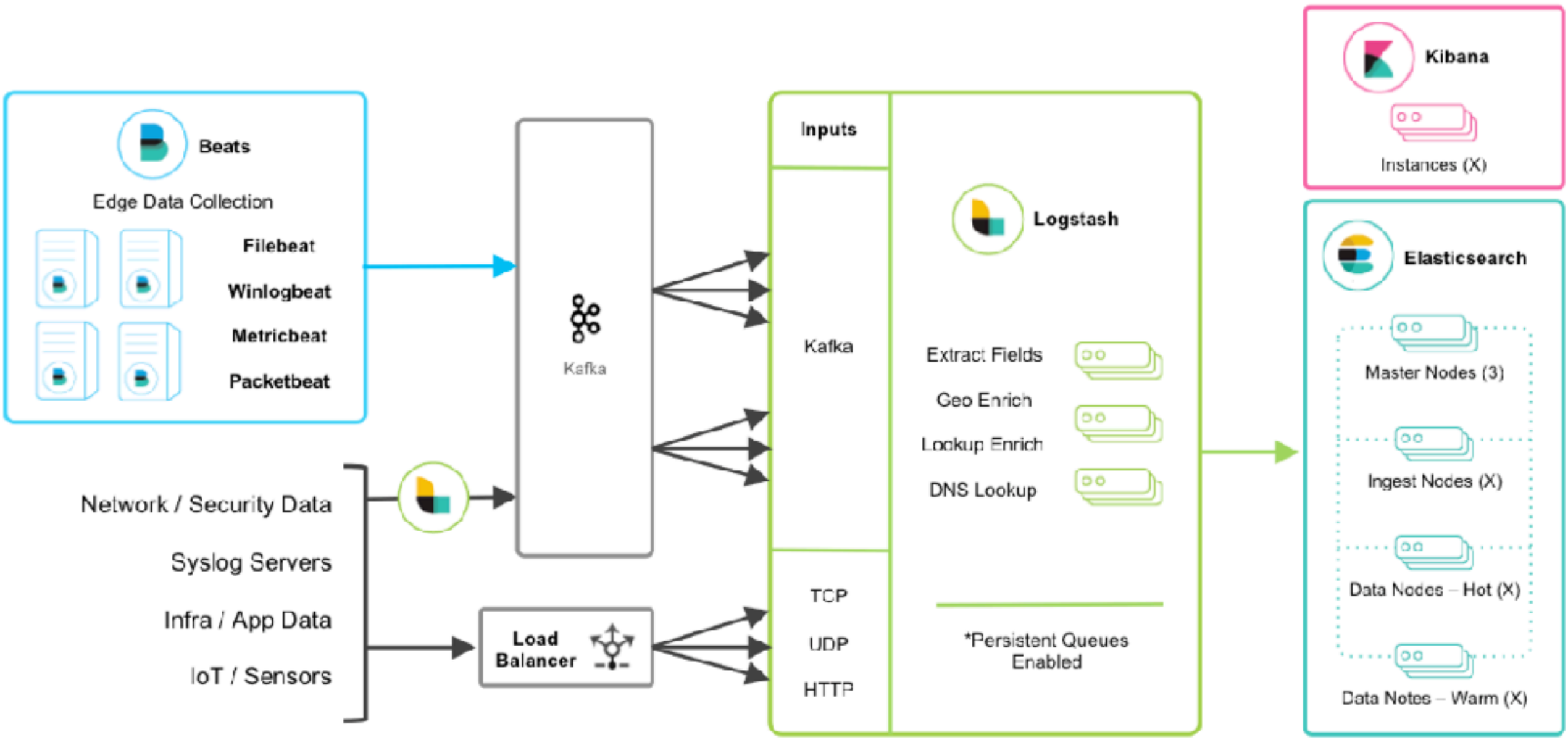
https://www.elastic.co/guide/en/logstash/current/deploying-and-scaling.html

More data sources



https://www.elastic.co/guide/en/logstash/current/deploying-and-scaling.html

Use messaging Queue



https://www.elastic.co/blog/logstash-persistent-queue

Design for Failure

Elasticsearch Nodes

Node Type	Description
Master	Control the cluster
Data	Keep/store data
HTTP/Query	Run your query
Coordinating	Smart Load Balancer

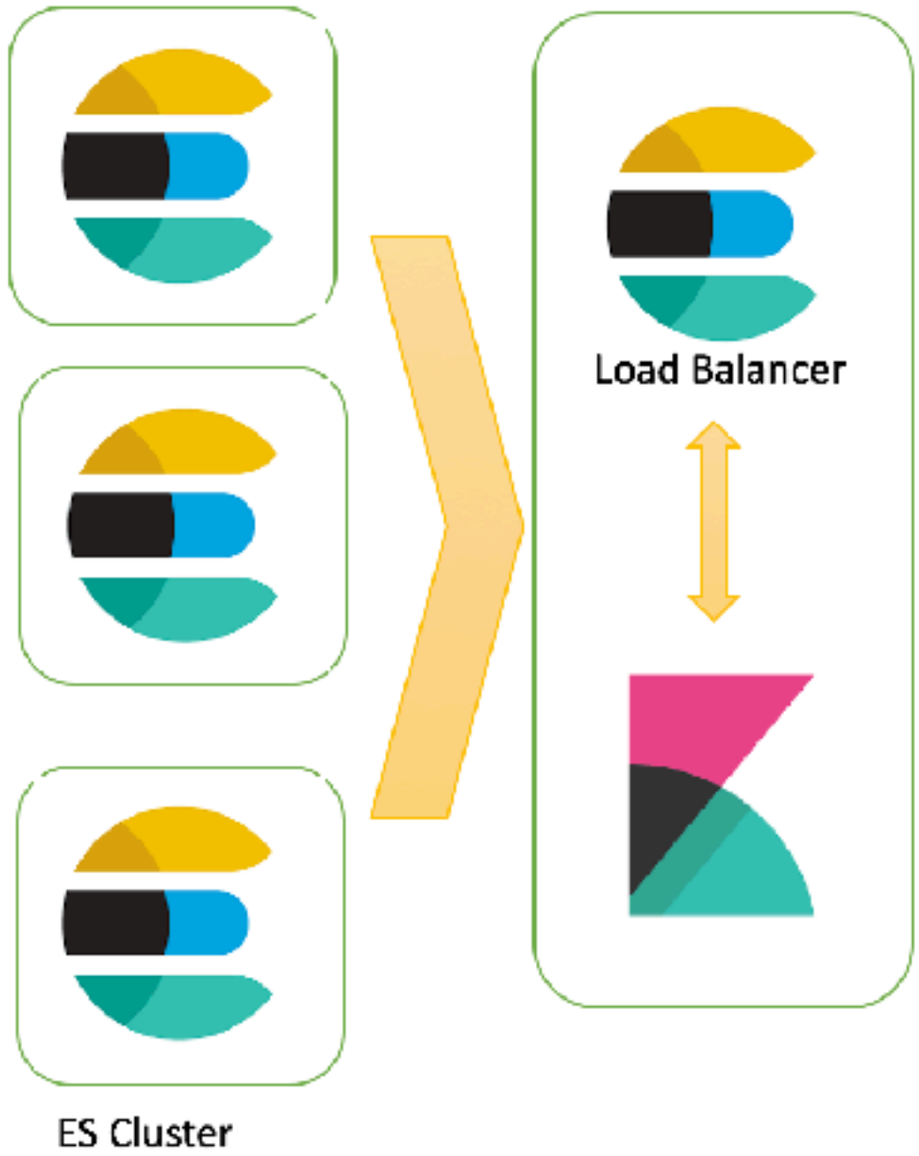
https://www.elastic.co/guide/en/elasticsearch/reference/current/modules-node.html

Elasticsearch Nodes

Not Secure | 35.240.161.188:9200/_cat/nodes?v&h=ip,name,node.role,master,heap.percent,ram.percent

```
ip          name        node.role master heap.percent ram.percent
10.148.0.2 master       m           *             17          33
10.148.0.4 query        -           -             10          63
10.148.0.5 coordinator  -           -              9          78
10.148.0.3 data         d           -             13          63
```

Elasticsearch Nodes

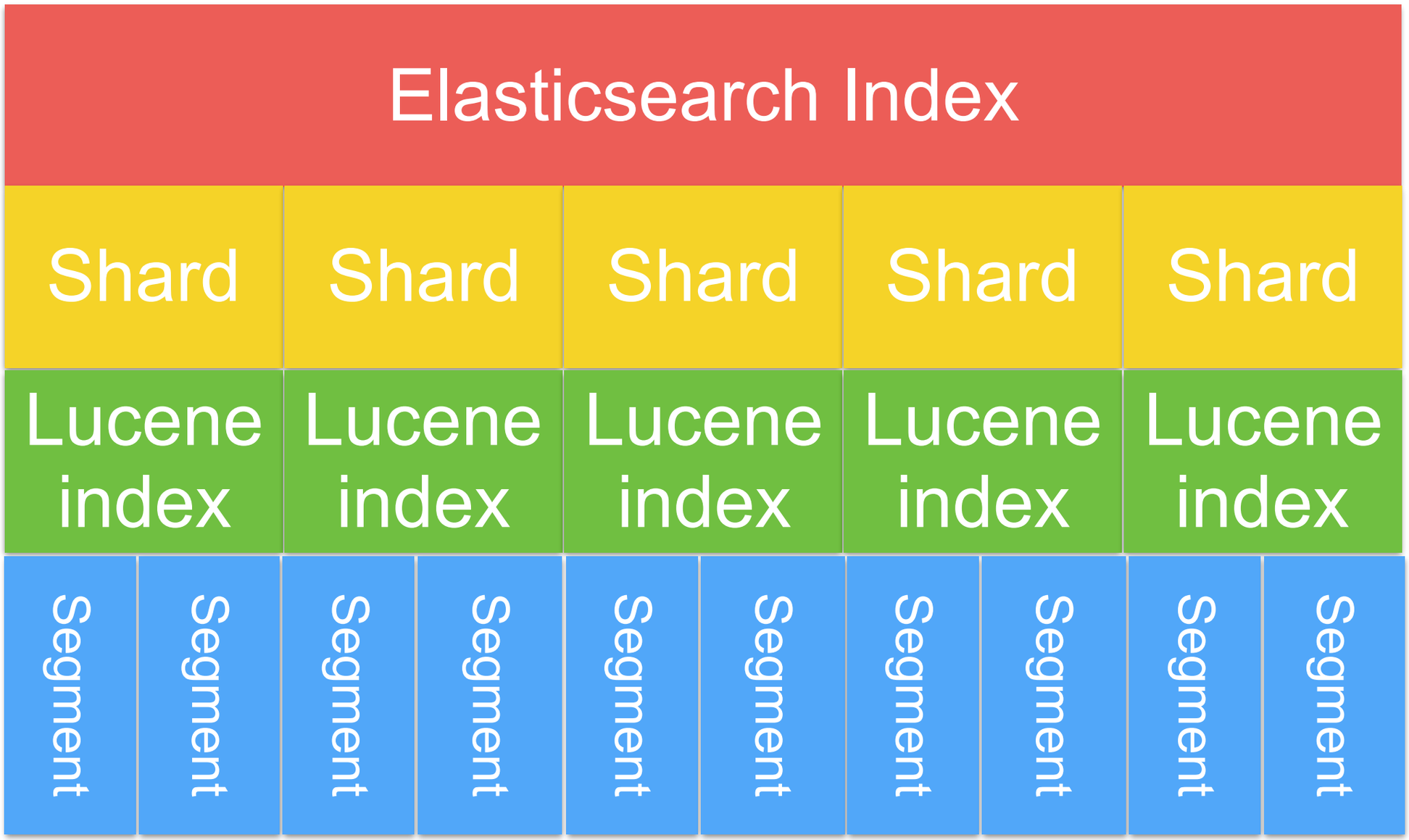


https://www.elastic.co/guide/en/kibana/current/production.html#load-balancing

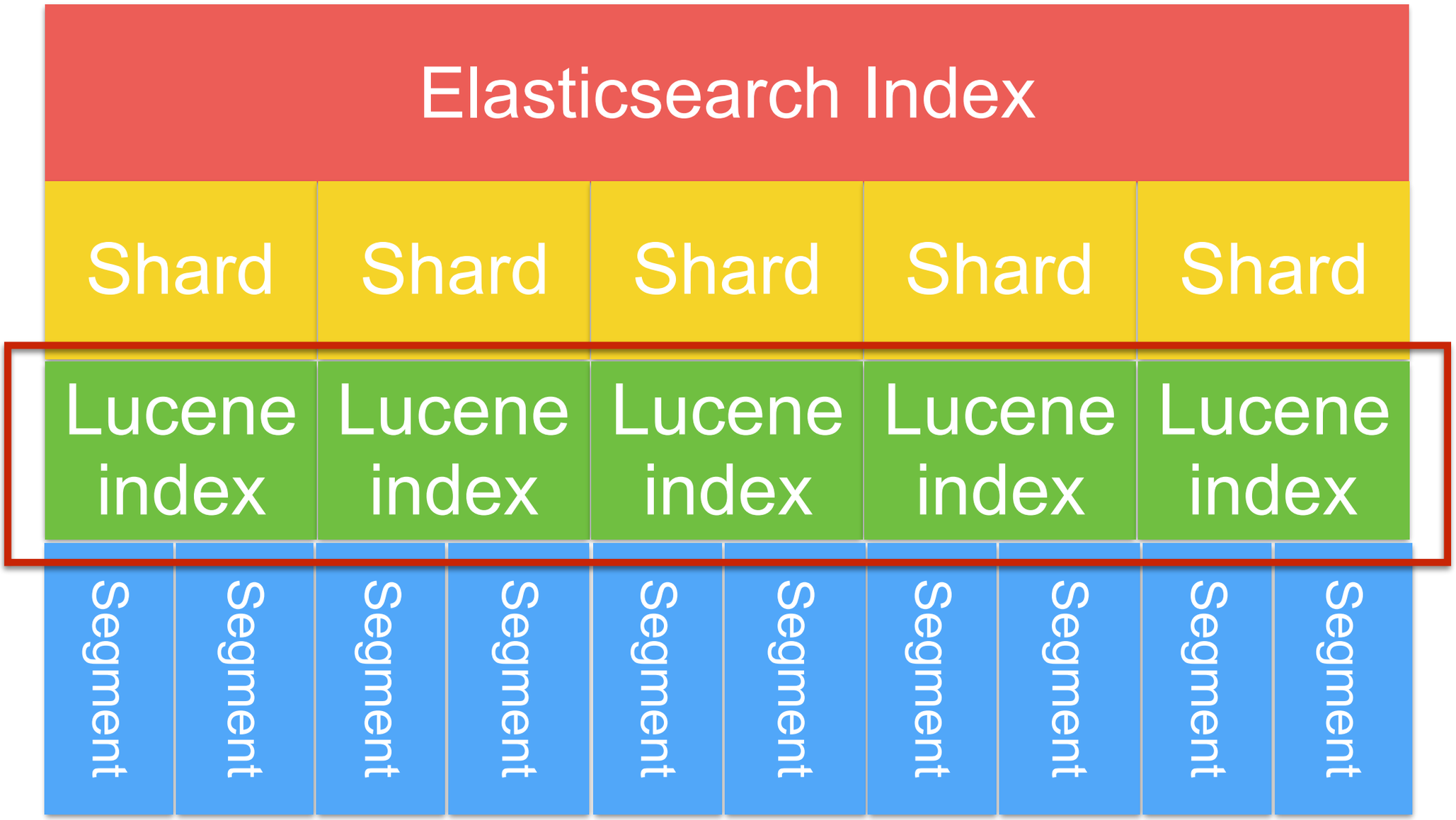
Apache Lucene

http://lucene.apache.org/

Apache Lucene

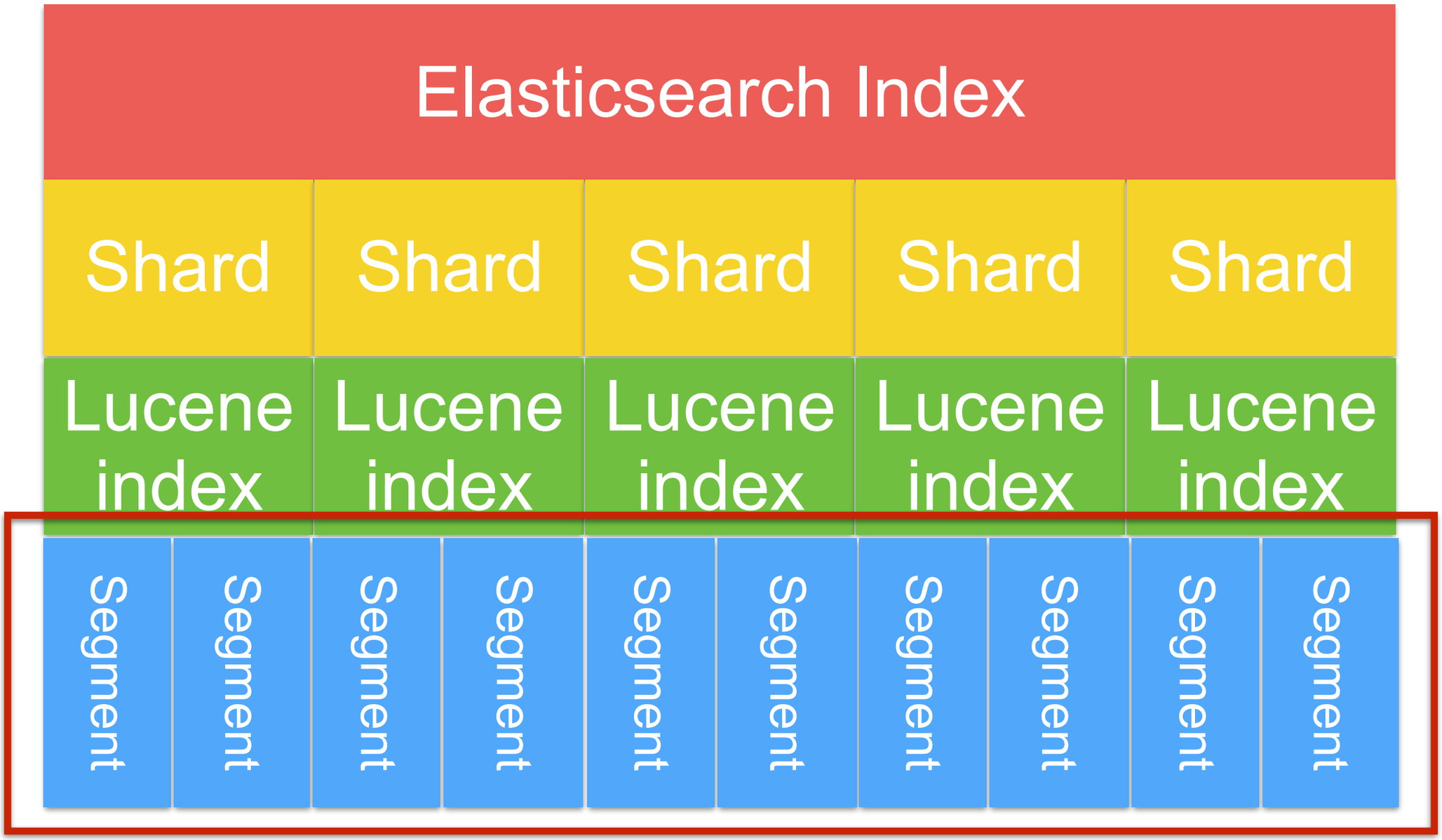


Apache Lucene



Max # of document of Lucene index = 2,147,483,519

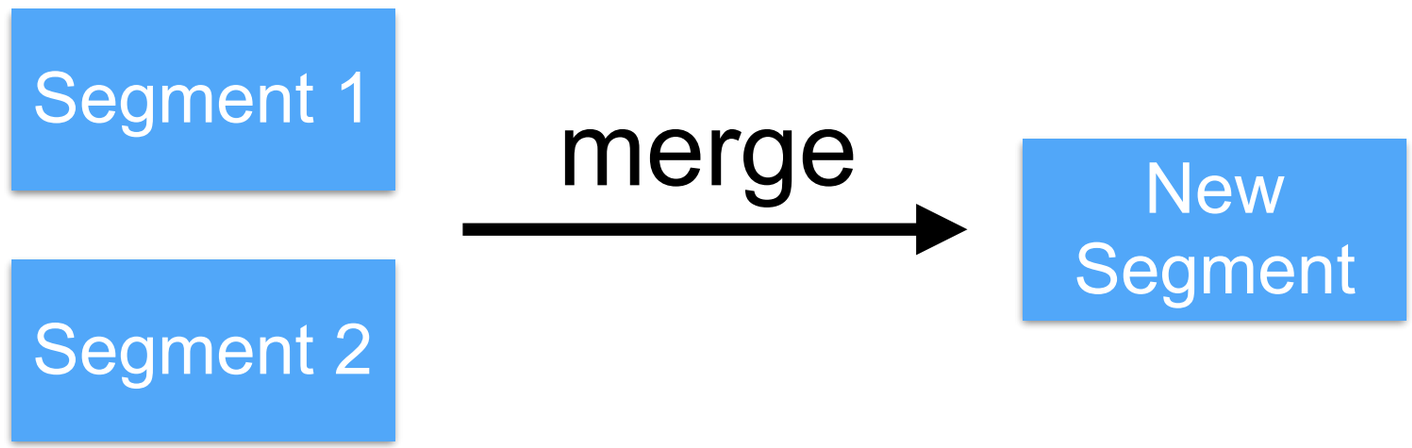
Apache Lucene



Segments are immutable

Segment

More shards, more segments
Documents are never delete !!
Lucene segment **merge** use more CPU/IO
Segments are immutable



Hardware

Hardware

CPU

Memory

Network

Storage

Memory

Enable bootstrap.memorylock
Disable all swap files
Change ES_HEAP_SIZE (default 1G)

https://www.elastic.co/guide/en/elasticsearch/reference/current/setup-configuration-memory.html

Design your index

Design your index

Sharding
Replication

Sharding

Elasticsearch divides the data in **logical** parts

# of sharding is define when index created

How many shard ?

Need to know your size of data

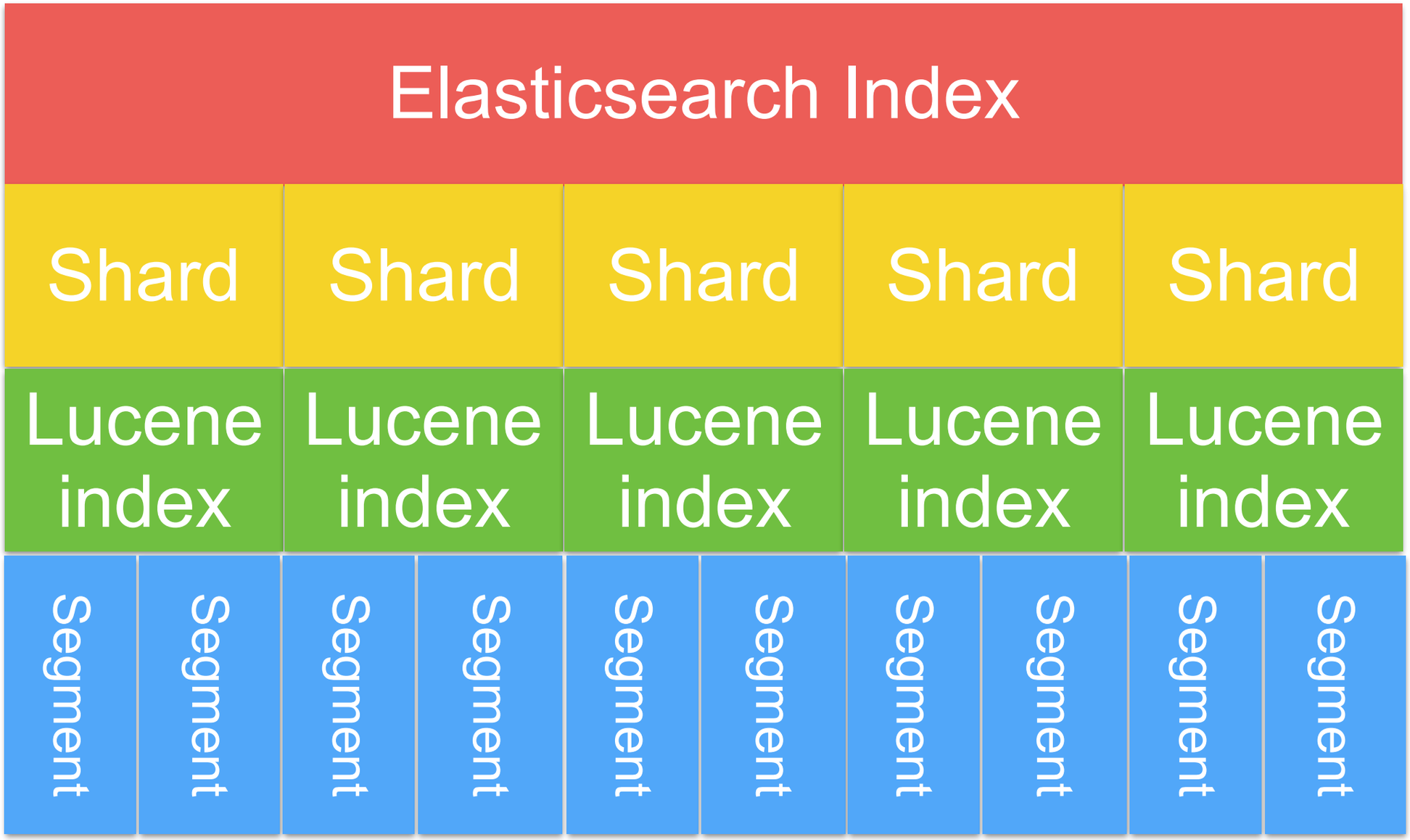
Data Size	# of shard
< 3M	1
>3M <5M	2
>5M	(# of document / 5M) + 1

Sharding

Small shards on multiple nodes make the cluster recovery faster

Small shards on a lot of nodes solve memory mgt problem when query on large data

More shard, more Segment !!



Need to config file descriptor

https://www.elastic.co/guide/en/elasticsearch/reference/current/system-config.html

Don’t create more shard than you need !!

Replication

Prevent data loss

Default = 1

nodes = [(primary + # replication) /2] + 1

Problems with scaling

CPU consumption
Load average
Request rate
Search latency

Slow log

```
PUT /myindex/_settings
{
    "index.search.slowlog.threshold.query.warn: 1s",
    "index.search.slowlog.threshold.query.info: 500ms",
    "index.search.slowlog.threshold.query.debug: 1500ms",
    "index.search.slowlog.threshold.query.trace: 300ms",
    "index.search.slowlog.threshold.fetch.warn: 500ms",
    "index.search.slowlog.threshold.fetch.info: 400ms",
    "index.search.slowlog.threshold.fetch.debug: 300ms",
    "index.search.slowlog.threshold.fetch.trace: 200ms"
}
```

If can’t optimize then add more resources or rewrite

https://www.elastic.co/guide/en/elasticsearch/reference/current/index-modules-slowlog.html

Indexing Data

Indexing data

Must be define data schema for your need
Default mapping == more cost (Memory/Disk)
Default for data is “text” + “keyword”
Understand analyzer and tokenizer
Use auto generated IDs if possible

Indexing data

Prefer bulk indexing

Change refresh interval

Time based index for log data

https://www.elastic.co/guide/en/elasticsearch/reference/current/indices-update-settings.html

For Large data

Increase refresh interval
Decrease replica number

```
PUT /logstash-2015.05.20/_settings
{
    "index" : {
        "refresh_interval" : "-1",
        "number_of_replicas" : 0
    }
}
```

https://www.elastic.co/guide/en/elasticsearch/reference/current/indices-update-settings.html

Query Data

Query data

Use filters as much as possible

Use scan and scroll for dumping large data

Node query cache

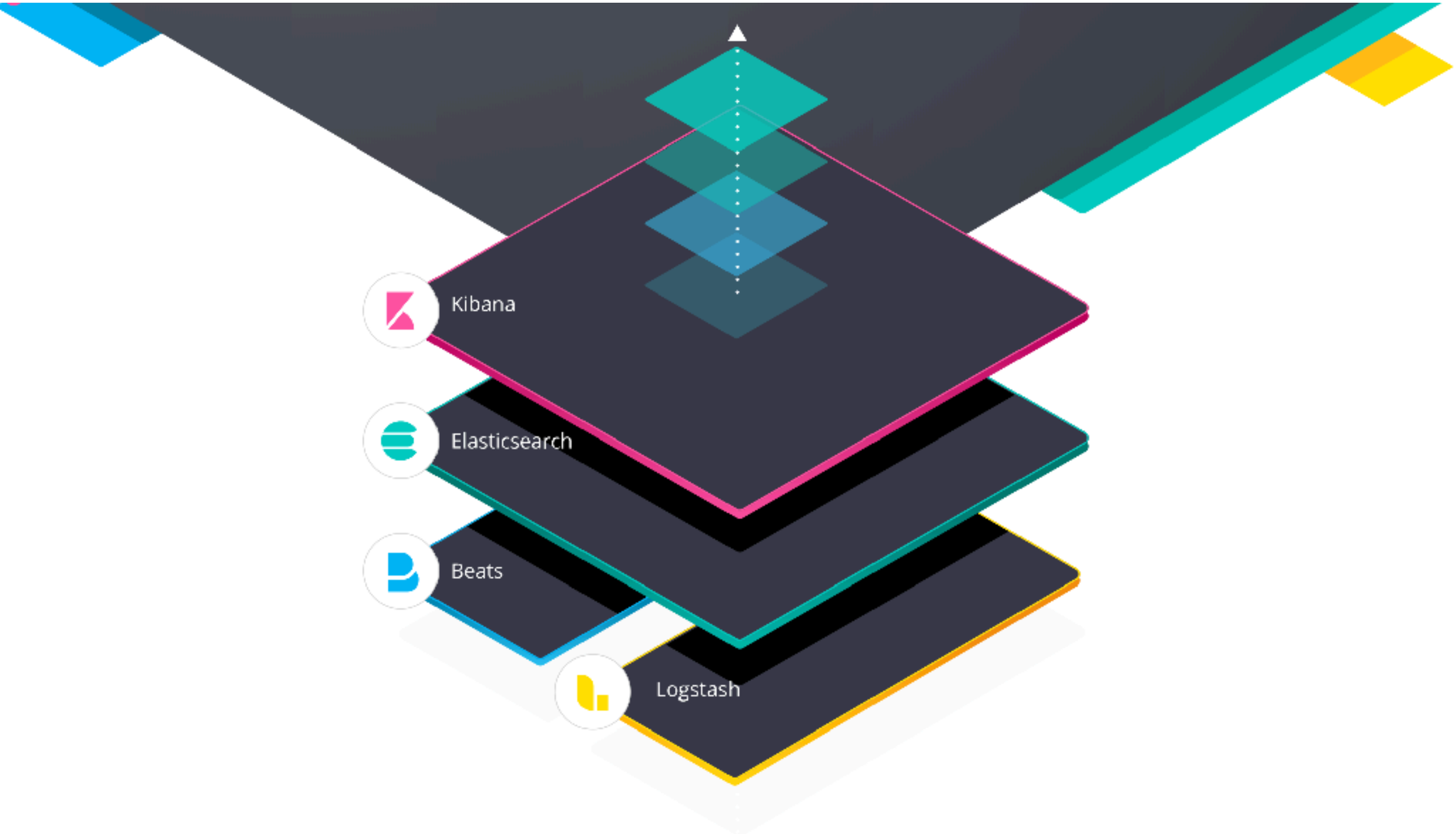
Shard query cache

Retrieve only necessary fields

https://www.elastic.co/guide/en/elasticsearch/reference/current/query-cache.html

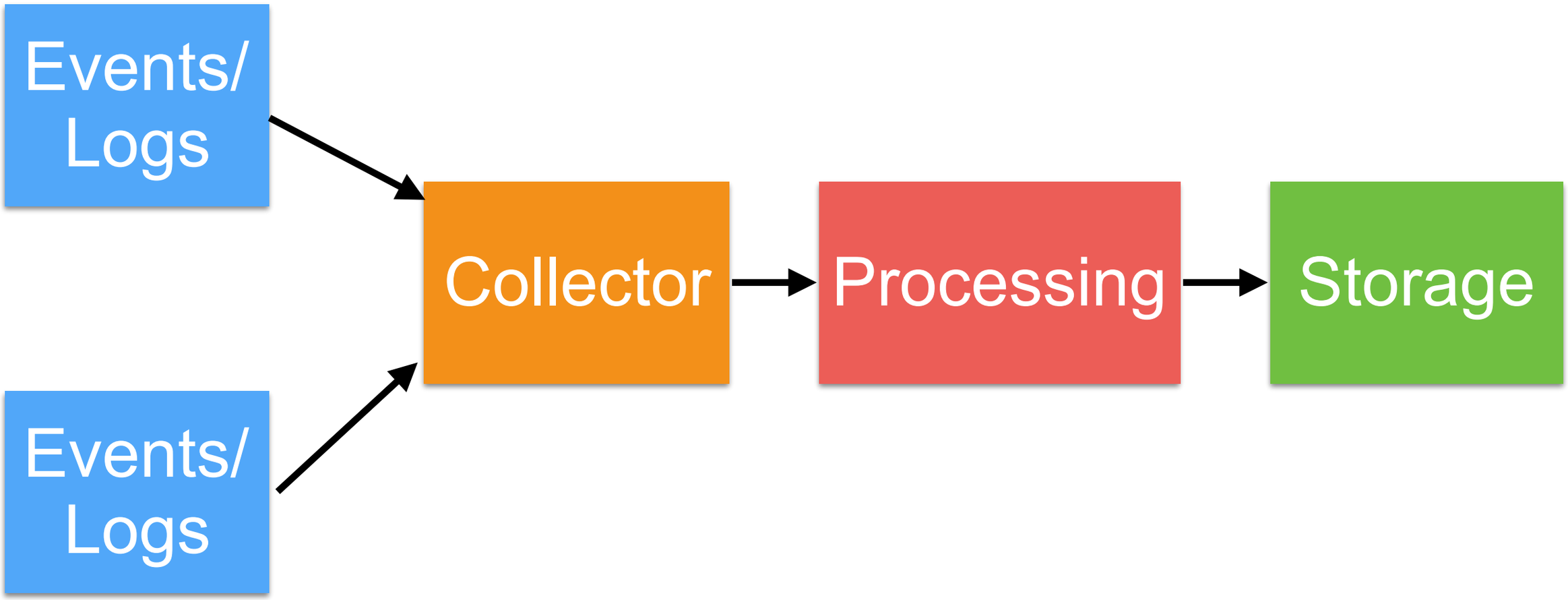
Use cases

ELK stack



https://www.elastic.co/elk-stack

Event or Logging from Servers



Event or Logging from Servers

Data Collection	Data Aggregation & Processing	Indexing & storage	Analysis & visualization

Event or Logging from Servers

RabbitMQ

NGINX

Data Collection	Buffering	Data Aggregation & Processing	Indexing & storage	Analysis & visualization