

ELK Stack





Somkiat
Home

Somkiat Puisungnoen

Update Info 1

View Activity Log 10+

...

Timeline

About

Friends 3,138

Photos

More ▾

When did you work at Opendream?

×

...

22 Pending Items

Intro

Software Craftsmanship

Software Practitioner at สยามชำนาญกิจ พ.ศ. 2556

Agile Practitioner and Technical at SPRINT3r

Post

Photo/Video

Live Video

Life Event

What's on your mind?

Public ▾

Post

Somkiat Puisungnoen

15 mins · Bangkok · 🌐 ▾

Java and Bigdata

...



Facebook interface for the page **somkiat.cc**. The top navigation bar includes the Facebook logo, the page name, a search bar, and icons for Home, Messages, Notifications (3), Insights, Publishing Tools, Settings, and Help.

The main content area features a large video player showing a man in a white Superman t-shirt with "SOMKIAT.CC" printed on it, posing against a white wall. A blue call-to-action button is overlaid on the video: "Help people take action on this Page." with a close icon (X). Below the video, there are buttons for "Liked", "Following", "Share", and a menu icon (three dots). A blue button labeled "+ Add a Button" is also visible.

The left sidebar contains the page name **somkiat.cc**, the handle **@somkiat.cc**, and a menu with options: Home, Posts, Videos, and Photos.



Agenda

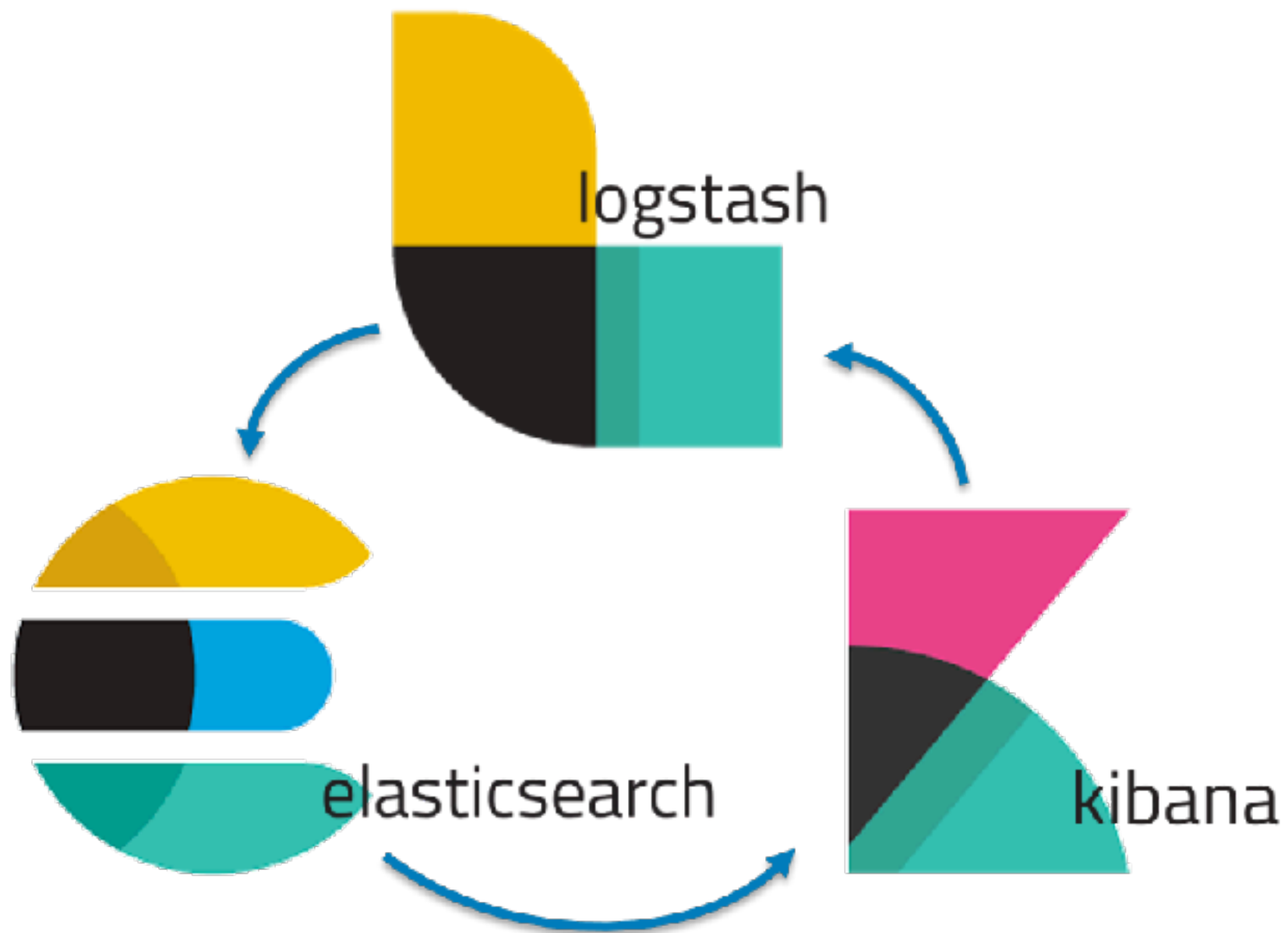
- ELK stack
- Introduction to Elasticsearch
- CRUD (Create, Read, Update, Delete)
- Search DSL (Domain Specific Language)
- Analyzer
- Mapping
- Aggregation



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	May 2018	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
[0g8-71W] publish_address {127.0.0.1:9200},
```



Hello Elasticsearch

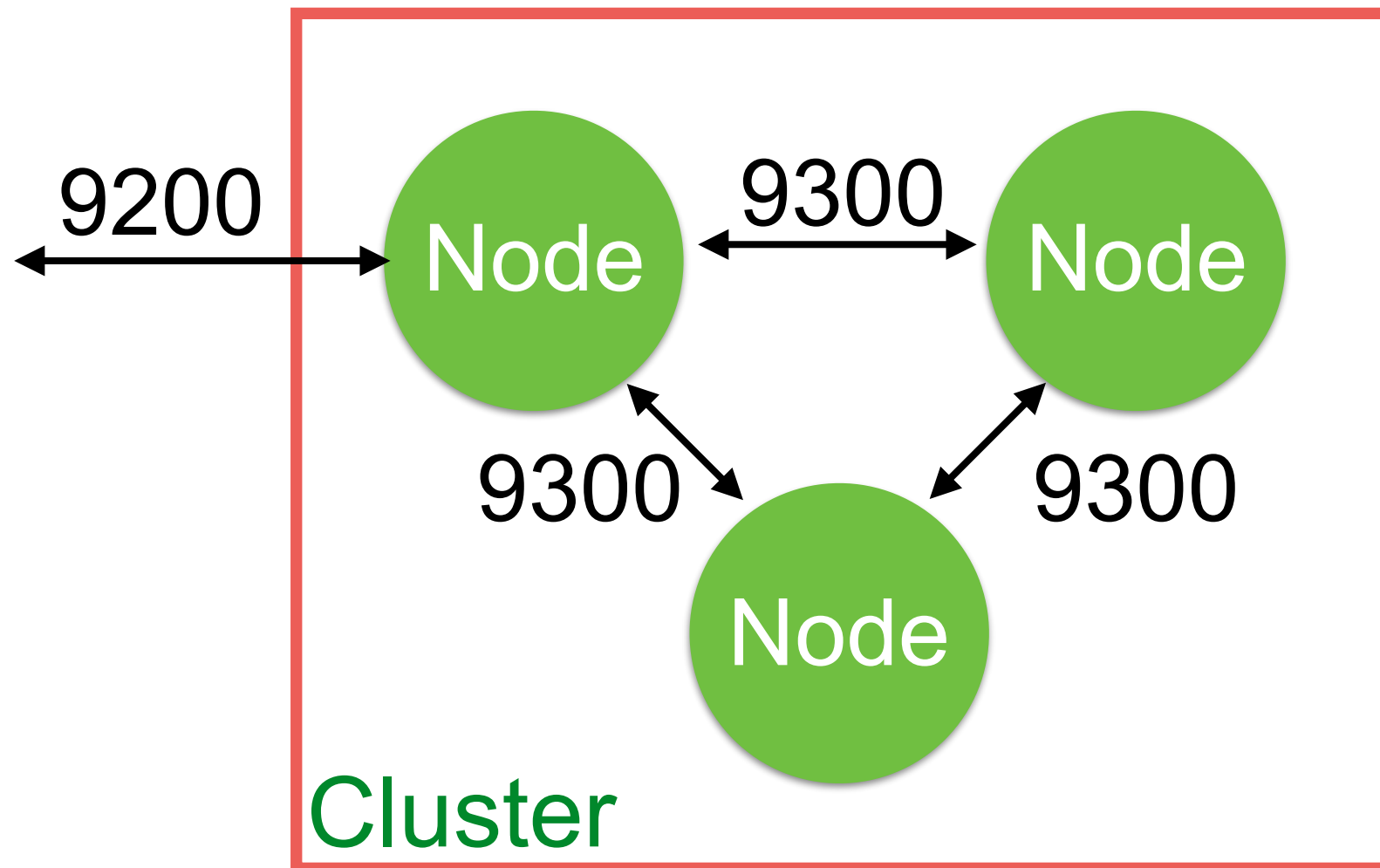
<http://localhost:9200/>

```
{
  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
lasticsearch
[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/>

Add Data to Kibana
Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM
APM automatically collects in-depth performance metrics and errors from inside your applications.
[Add APM](#)

Logging
Ingest logs from popular data sources and easily visualize in preconfigured dashboards.
[Add log data](#)

Metrics
Collect metrics from the operating system and services running on your servers.
[Add metric data](#)

Security analytics
Centralize security events for interactive investigation in ready-to-go visualizations.
[Add security events](#)

Data already in Elasticsearch?
[Set up index patterns](#)

Visualize and Explore Data

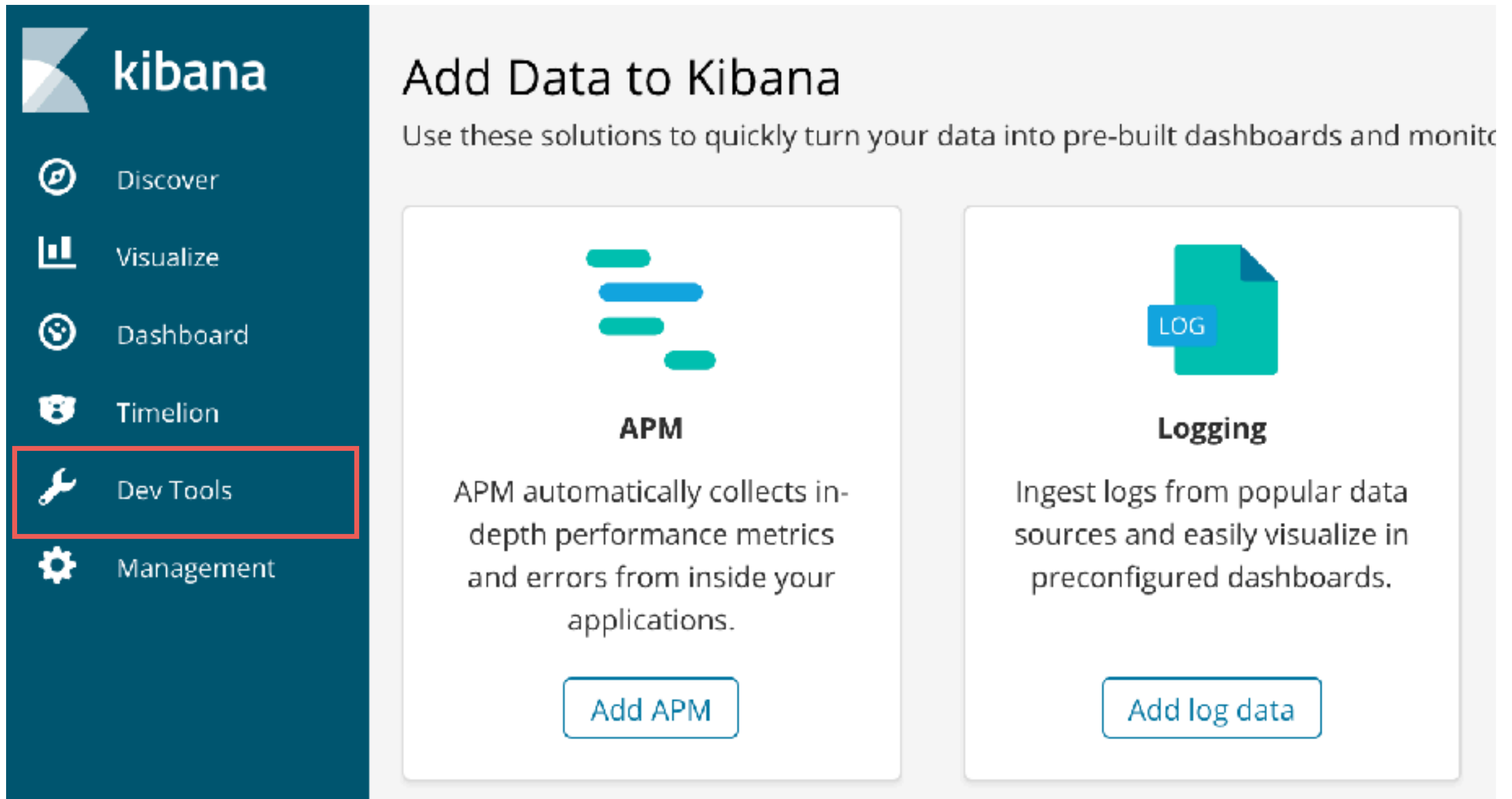
- Dashboard**
Display and share a collection of visualizations and saved searches.
- Discover**
Interactively explore your data by querying and filtering raw documents.
- Timelion**
Use an expression language to analyze time series data.
- Visualize**
Create visualizations and aggregate data stores in your

Manage and Administer the Elastic Stack

- Console**
Skip cURL and use this JSON interface to work with your data directly.
- Index Patterns**
Manage the index patterns that help retrieve your data from Elasticsearch.
- Saved Objects**
Import, export, and manage your saved searches.



Using Dev Tools




The image shows the Kibana user interface. On the left is a dark teal sidebar with the Kibana logo and a list of navigation items: Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The 'Dev Tools' item, which has a wrench icon, is highlighted with a red rectangular border. The main content area is light gray and titled 'Add Data to Kibana'. Below the title is a subtitle: 'Use these solutions to quickly turn your data into pre-built dashboards and monitor your applications.' There are two white cards with rounded corners. The first card is for 'APM' (Application Performance Monitoring), featuring a blue icon of three horizontal bars of increasing length. The text on the card says 'APM automatically collects in-depth performance metrics and errors from inside your applications.' and has a blue button labeled 'Add APM'. The second card is for 'Logging', featuring a blue icon of a document with a 'LOG' label. The text on the card says 'Ingest logs from popular data sources and easily visualize in preconfigured dashboards.' and has a blue button labeled 'Add log data'.

kibana

- Discover
- Visualize
- Dashboard
- Timelion
- Dev Tools**
- Management

Add Data to Kibana


Use these solutions to quickly turn your data into pre-built dashboards and monitor your applications.



APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Add APM



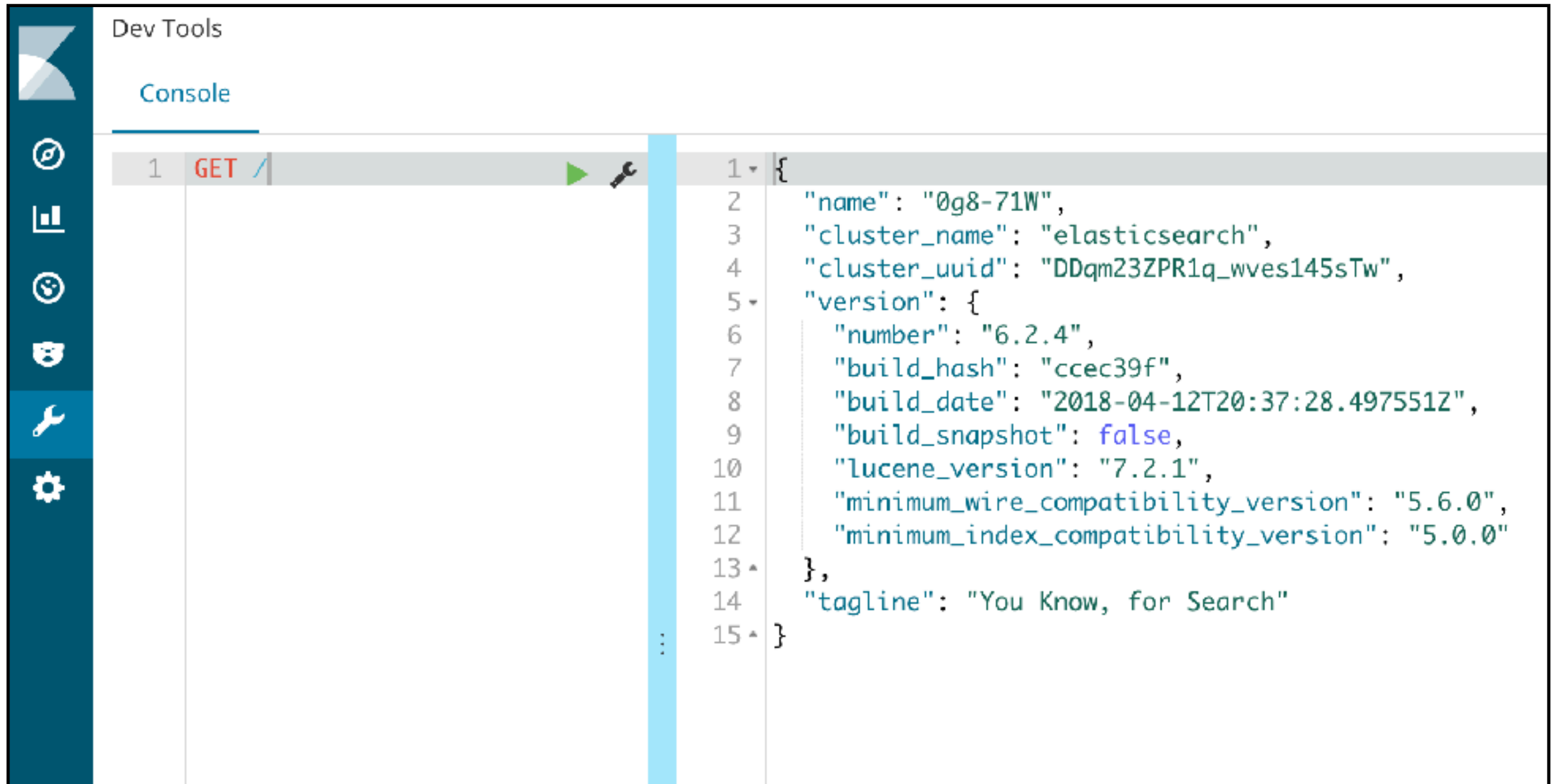
Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Add log data



Ready to start



CRUD with Elasticsearch

Create document

Read document

Update document

Delete document



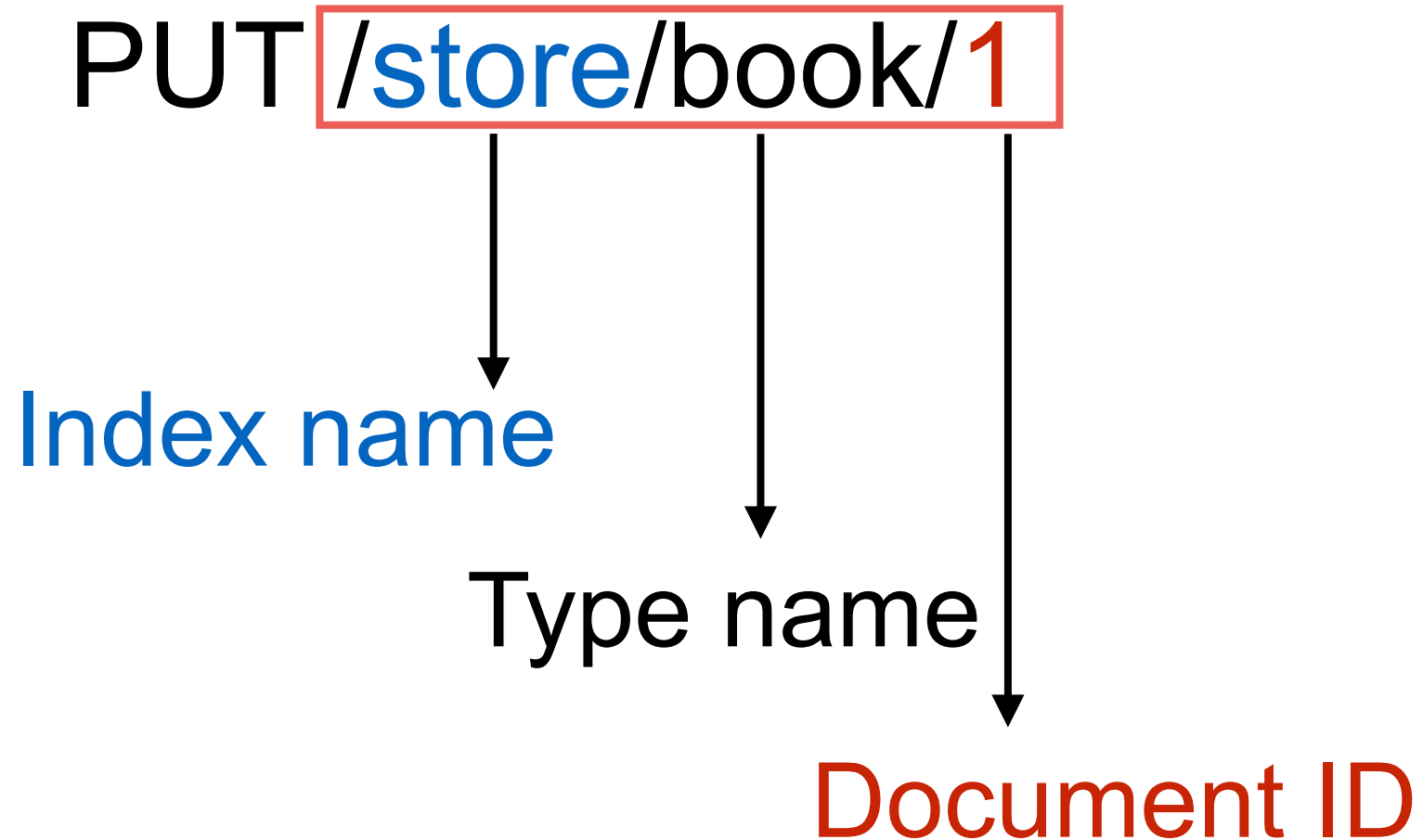
Create a document

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>



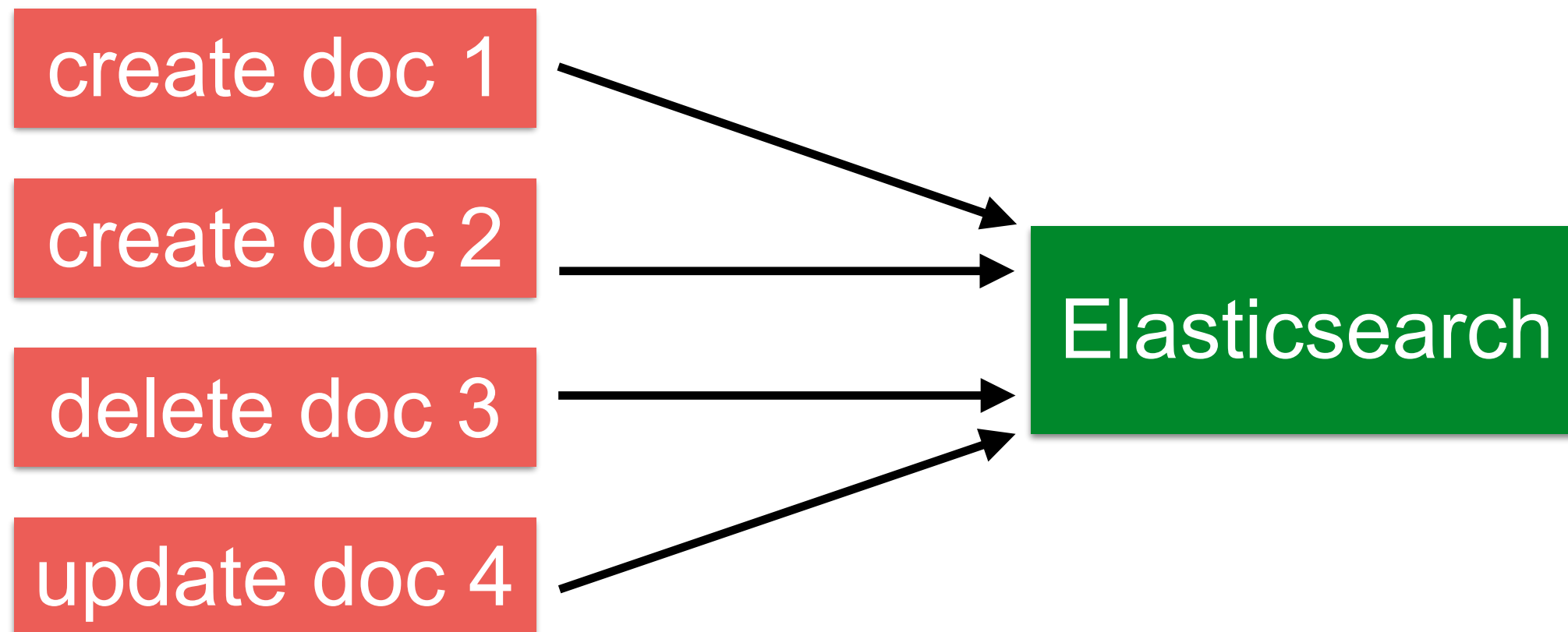
Bulk API

Perform many index/delete operation in
single API call

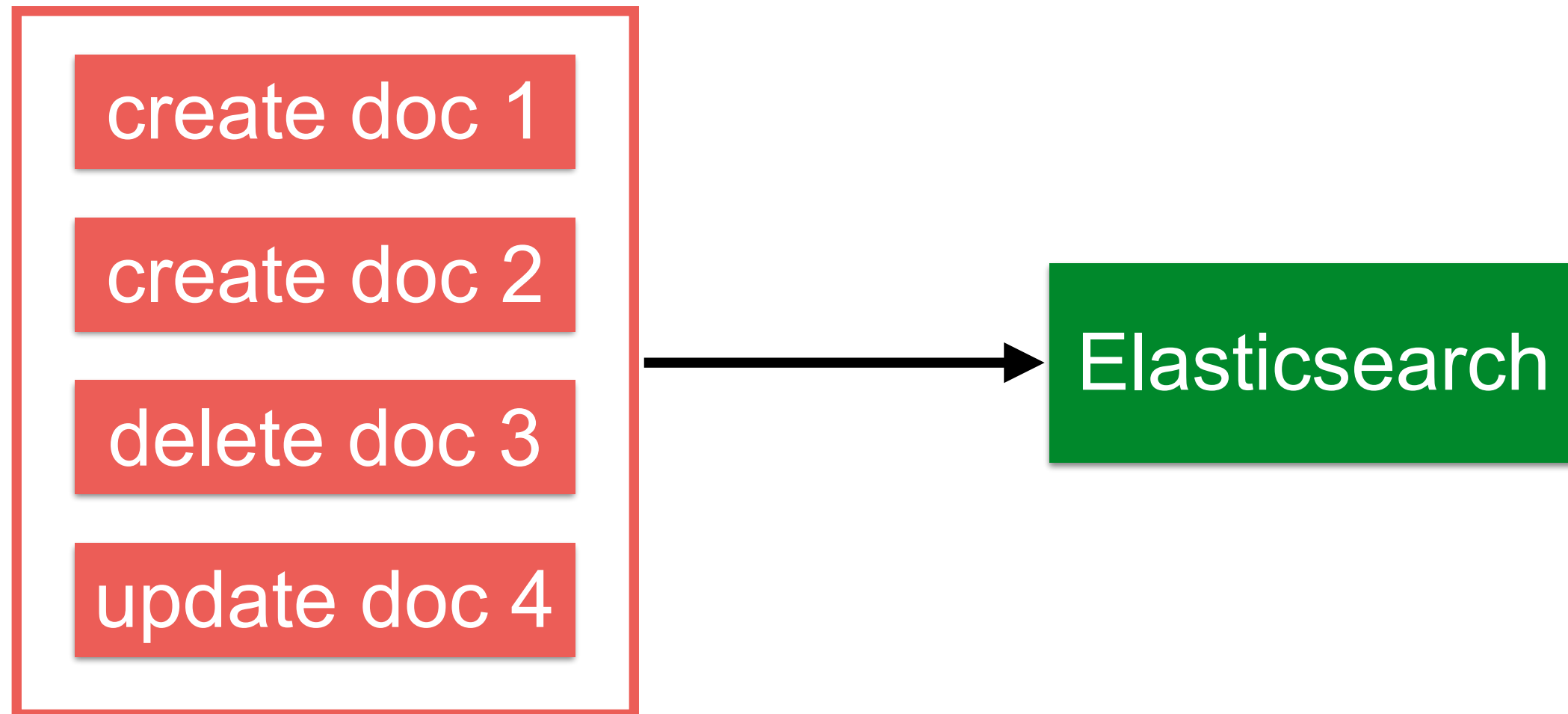
Increase indexing speed



Without Bulk API



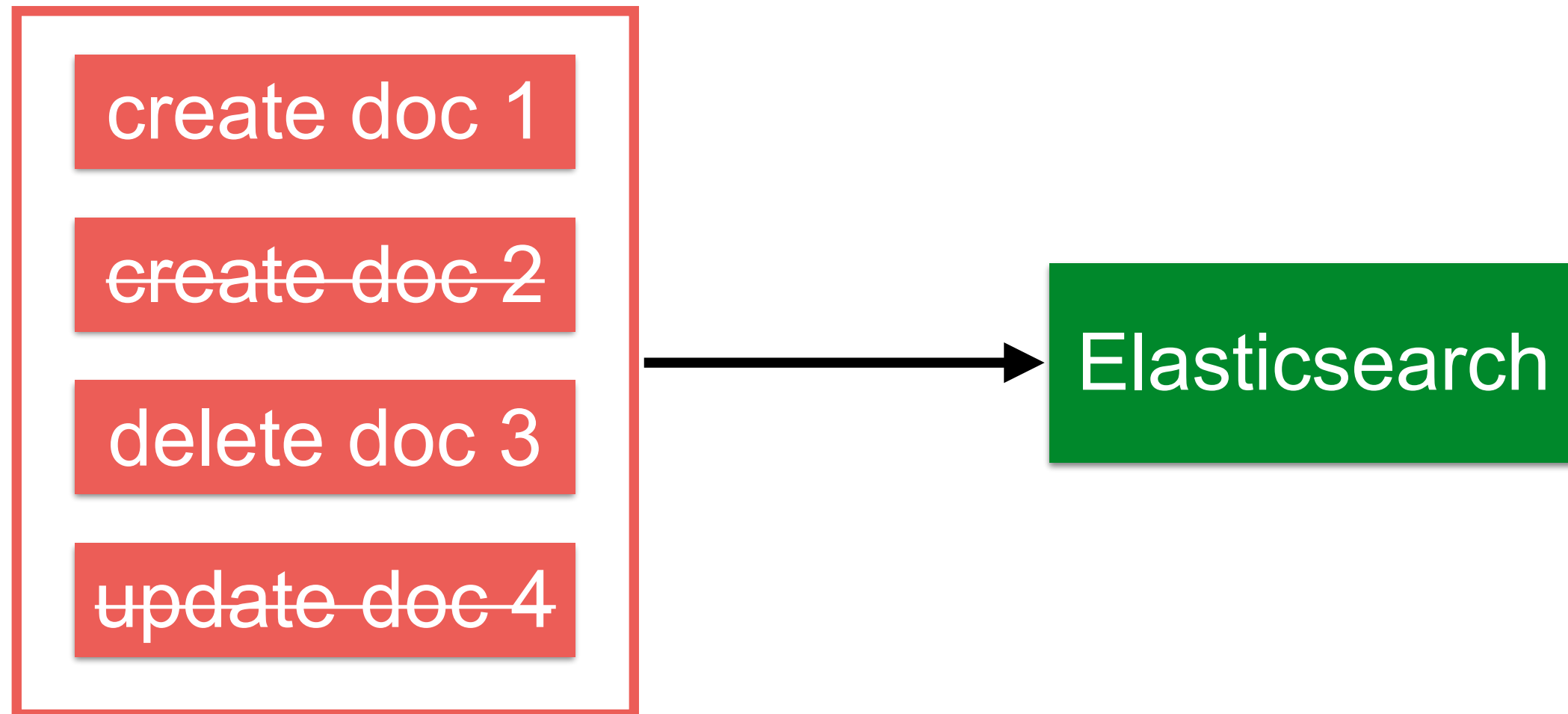
With Bulk API



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

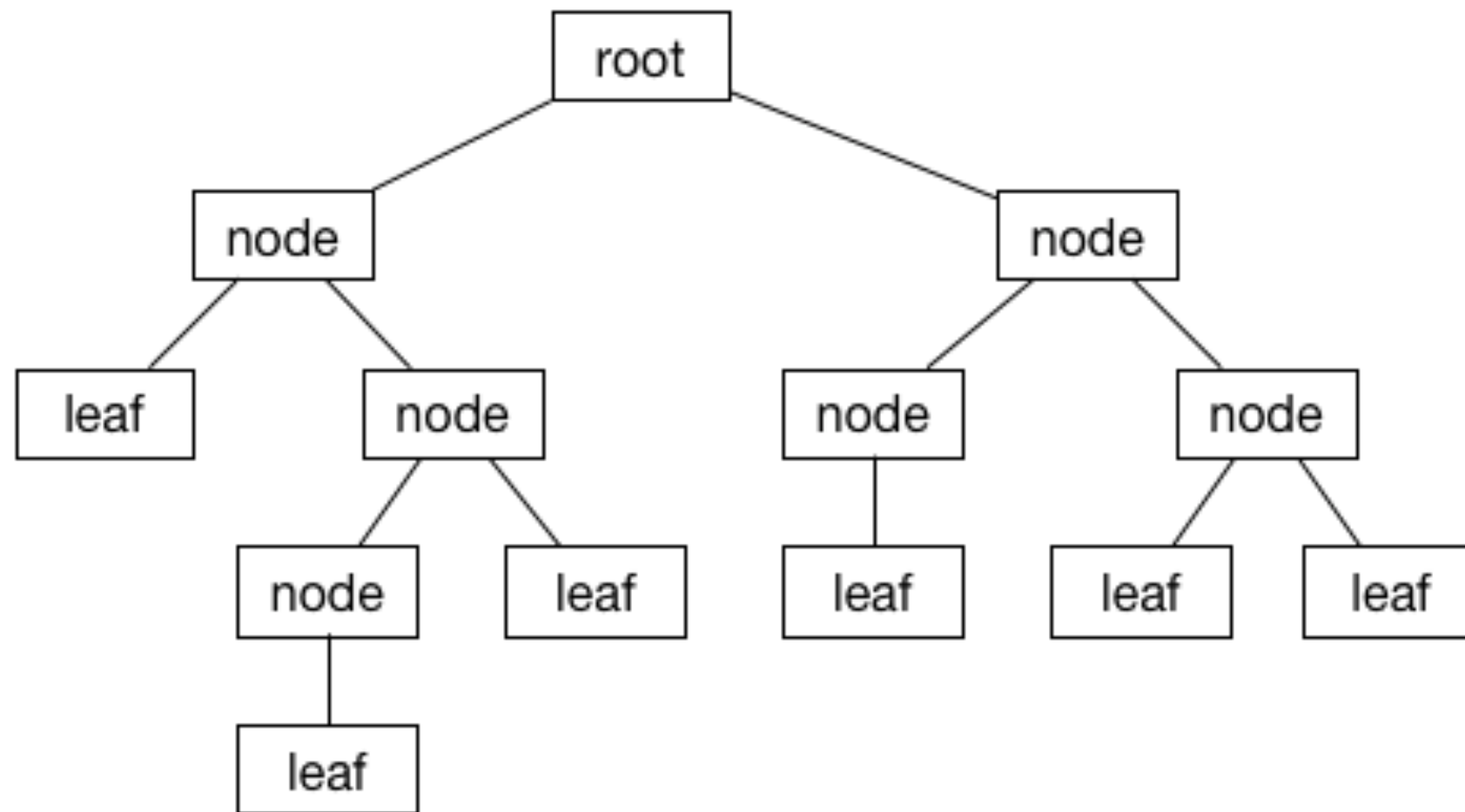
Domain Specific Language for query data
Flexible query language
Based on JSON format

<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

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

Search result



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, Set search result size = 0  
  "aggs": {  
    "all_book_title": {  
      "terms": {  
        "field": "category.keyword"  
      }  
    }  
  }  
}
```



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

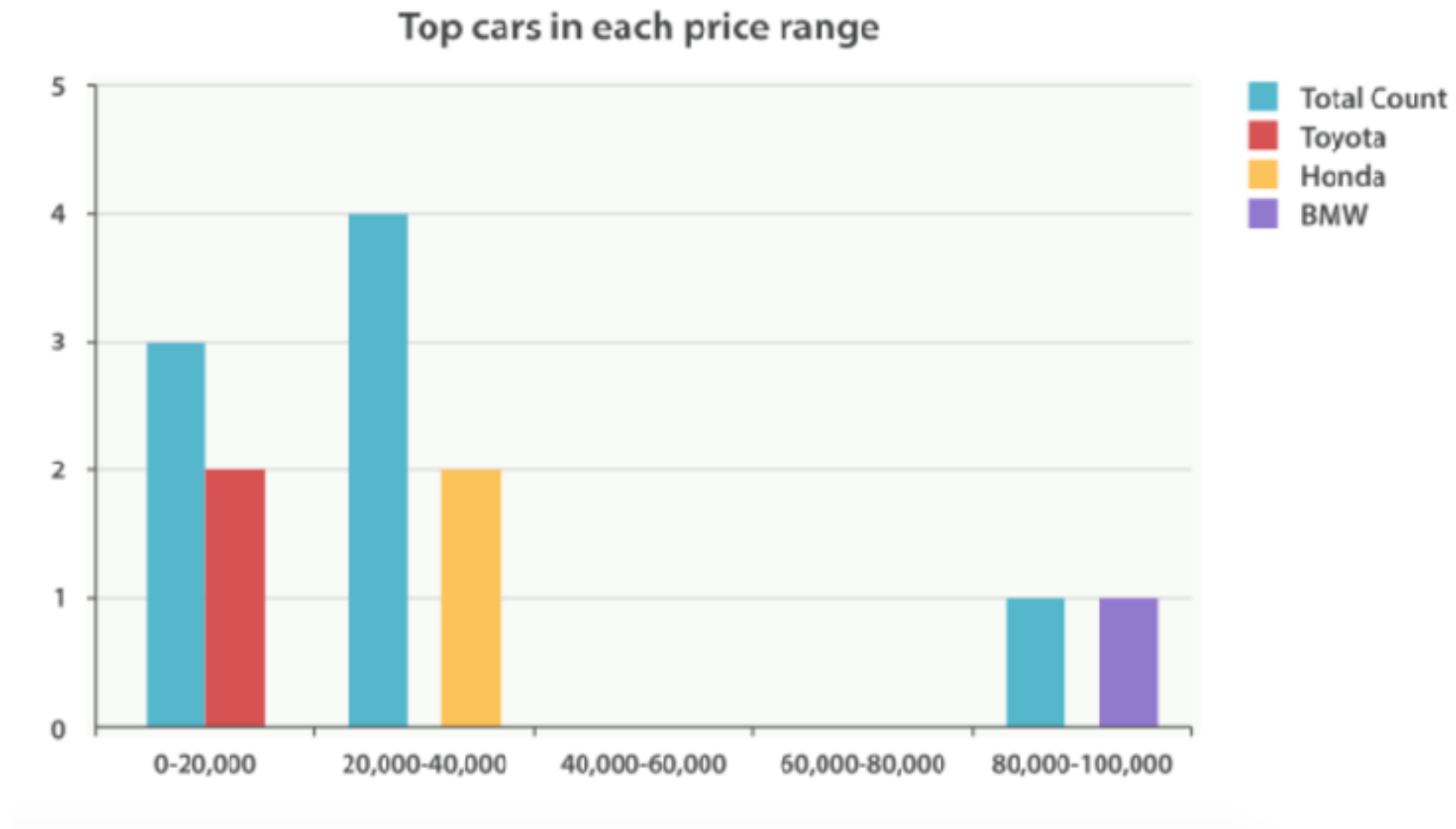
Statistic of best seller by color

Detail of car in each color

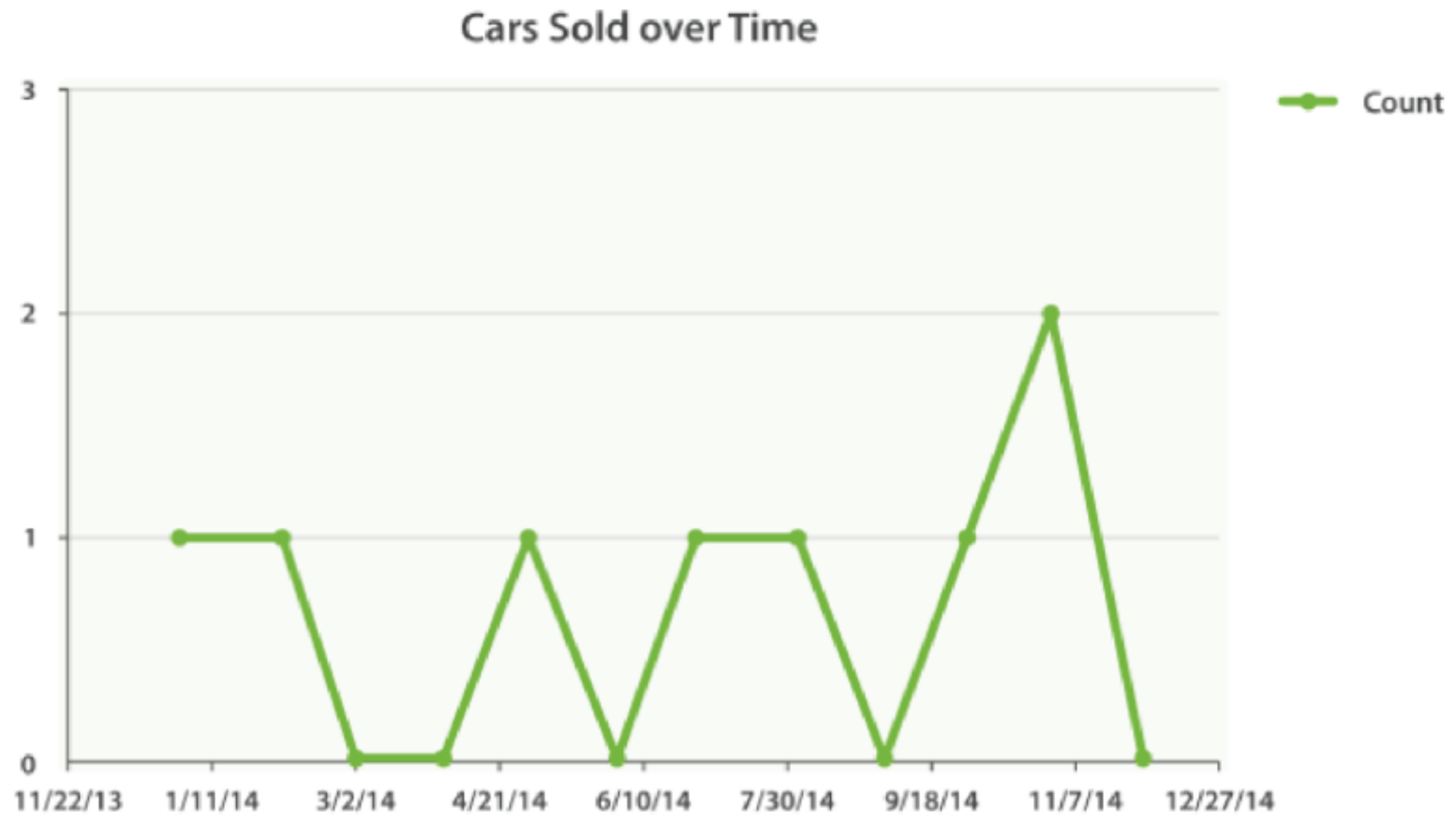
Min/max of price by make



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



Geo location

