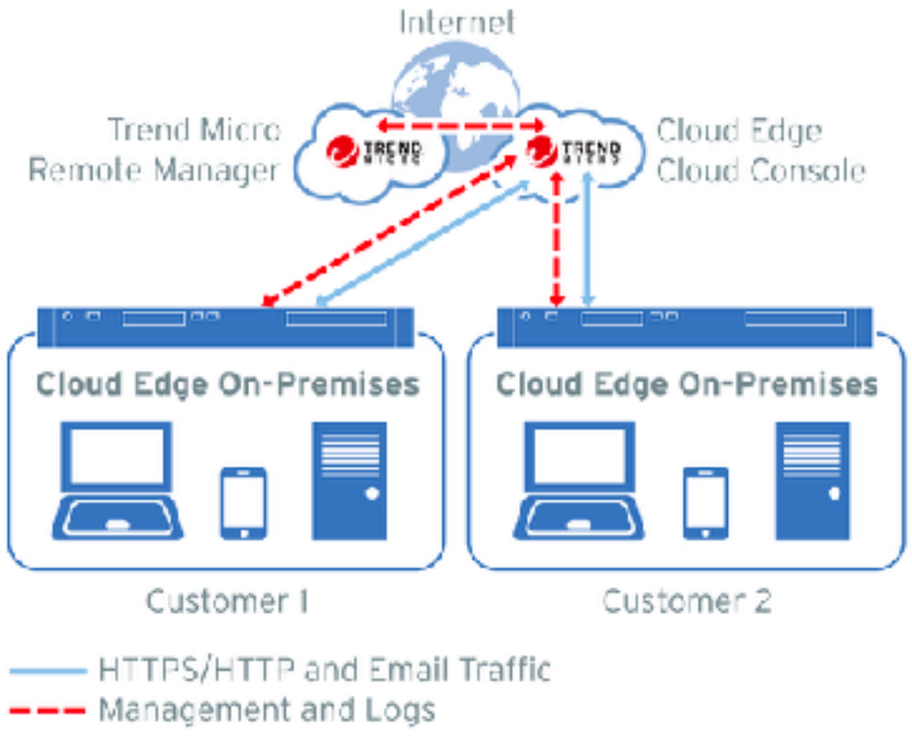
ElasticLog with ES in CloudEdge

Bruce Zhao

CloudEdge

- A hybrid Unified Threat Management (UTM) appliance.
  - Firewall, IPS, URL filter, web security, email security, application control, bandwidth control, user VPN, etc.
  - Combination of on-premise appliance and cloud service.



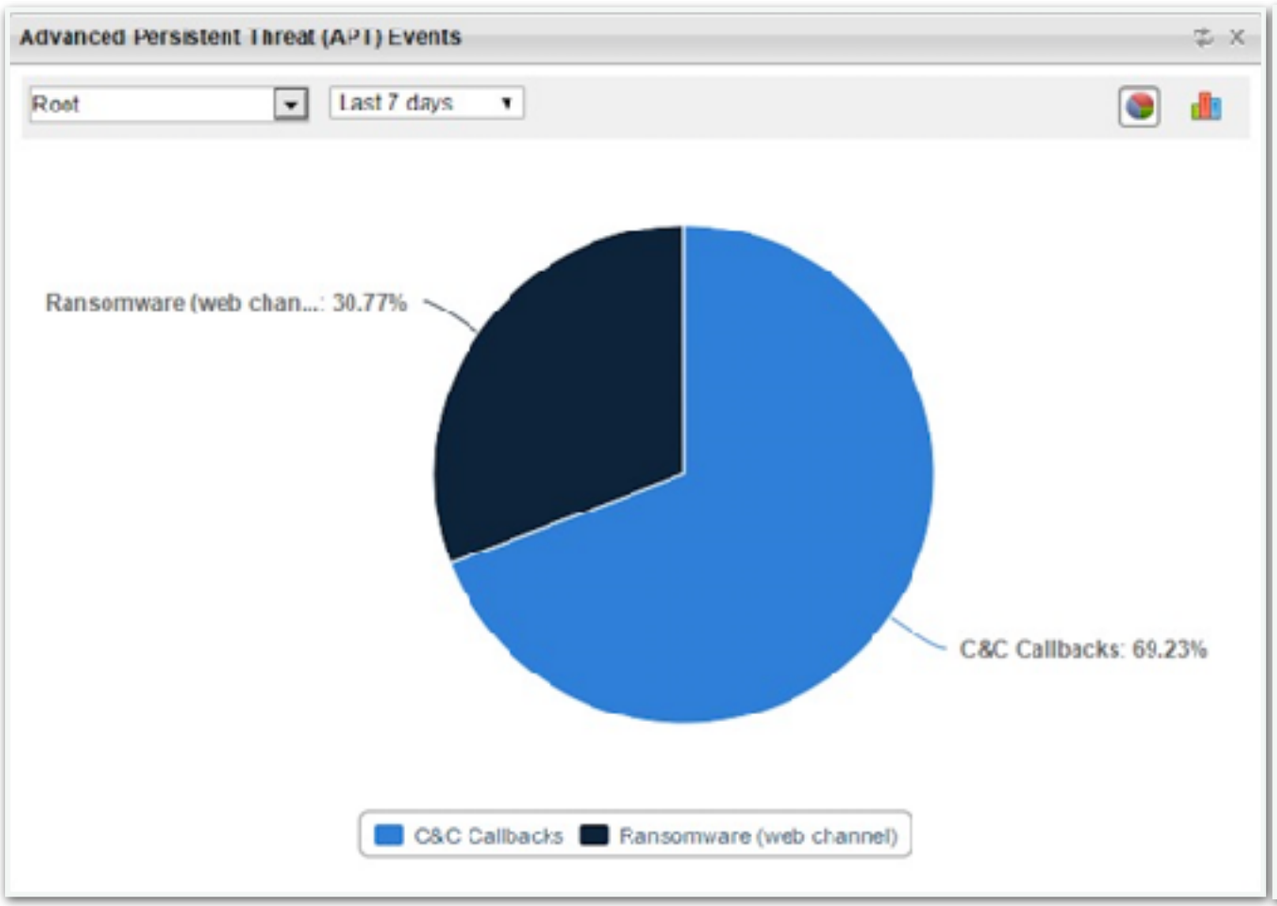
Project ElasticLog

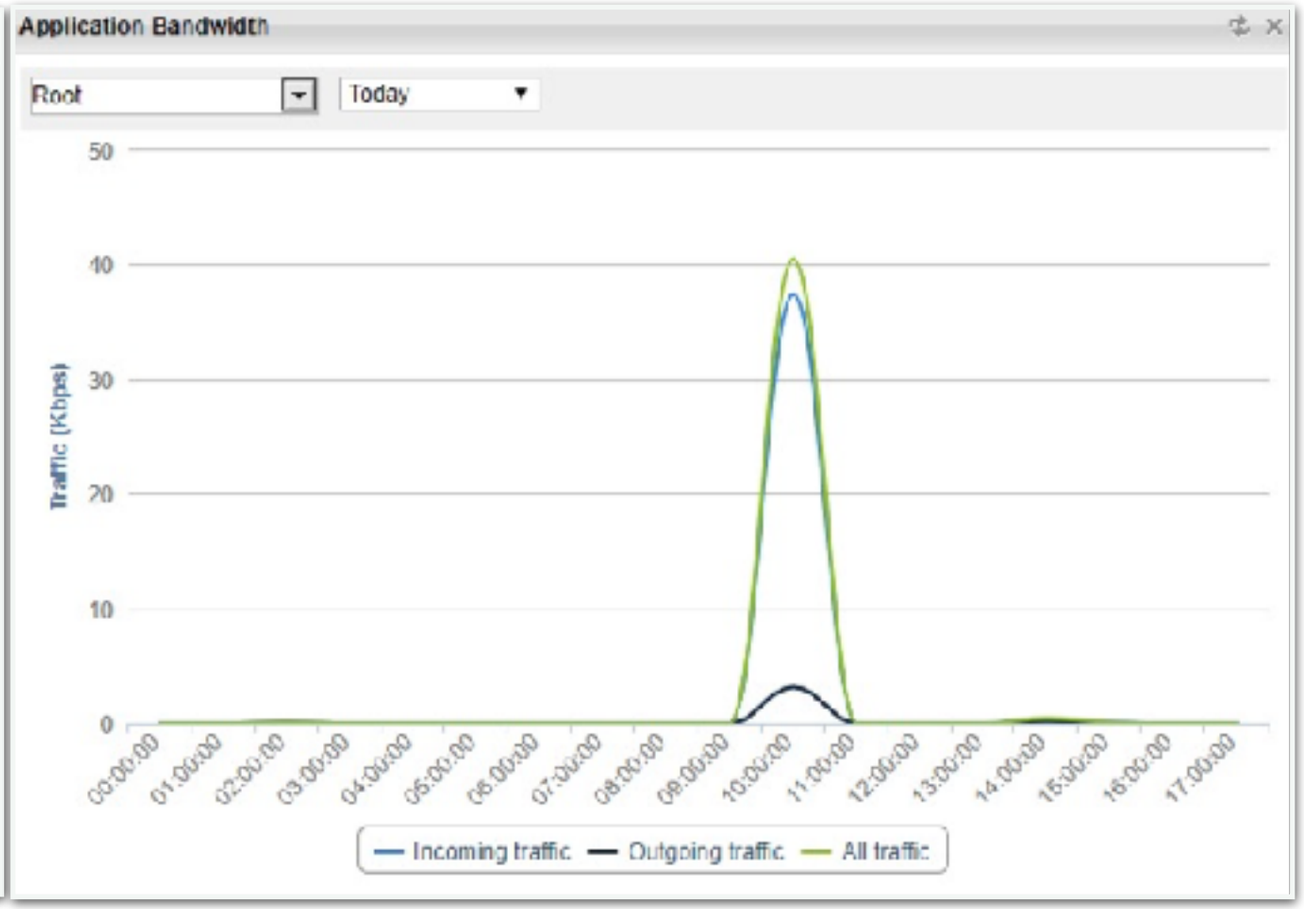
Internet | Bandwidth | Security | Policy

Continuous Structured Data

```
2018-06-27 16:04:45 +0800       nj-host-ken     202.45.67.178   Allen Fang (RD-CN)      102.34.23.43
80      6       1504    0       3       qq-policy       4       1051149 ken.java        Virus007
www.google.com  /find.php?c=bcda        81      60,58,33,35             1357    0

2018-06-27 16:04:45 +0800       nj-host-ethan   10.64.69.252    Ning Chen (QA-CN)       67.32.45.67
21      6       1504    0       1       Default 9       1055938 mfile.mp3       Virus007        www.sina
.com    /find.php?c=bcda        21      21,18,31,45             9892    0

2018-06-27 16:04:45 +0800       nj-host-james   10.204.16.66    Ethan She (RD-CN)       10.204.1.1
2543    17      2412    0       1       allow-web       13      1054202 ken.java        KidVirus002
www.baidu.com   /index.html     81      60,58,33,35             9892    0

2018-06-27 16:04:45 +0800       nj-host-ken     64.67.32.142    Allen Fang (RD-CN)      102.34.23.43
22      17      1170    0       2       qq-policy       5       1055988 ken.java        KidVirus002
www.sina.com    /index.html     60      21,18,31,45             25436   1

2018-06-27 16:04:45 +0800       nj-host-ken     202.45.67.178   Ethan She (RD-CN)       90.32.24.159
21      17      1521    0       1       qq-policy       15      1055988 jamesc.conf     Virus008
www.sina.com    /search.htm     81      21,18,31,45             6789    1
```

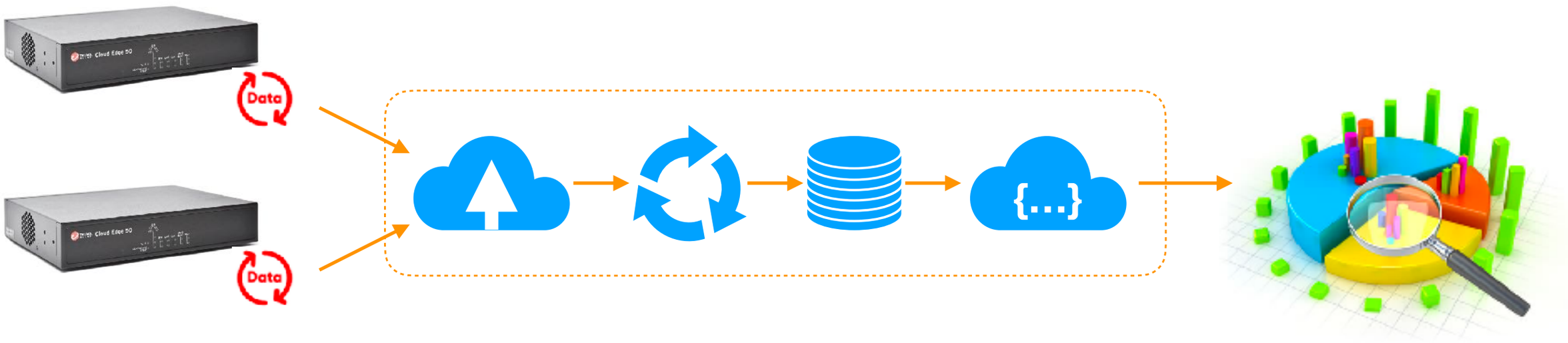
Project ElasticLog





Data Analytics

Project ElasticLog

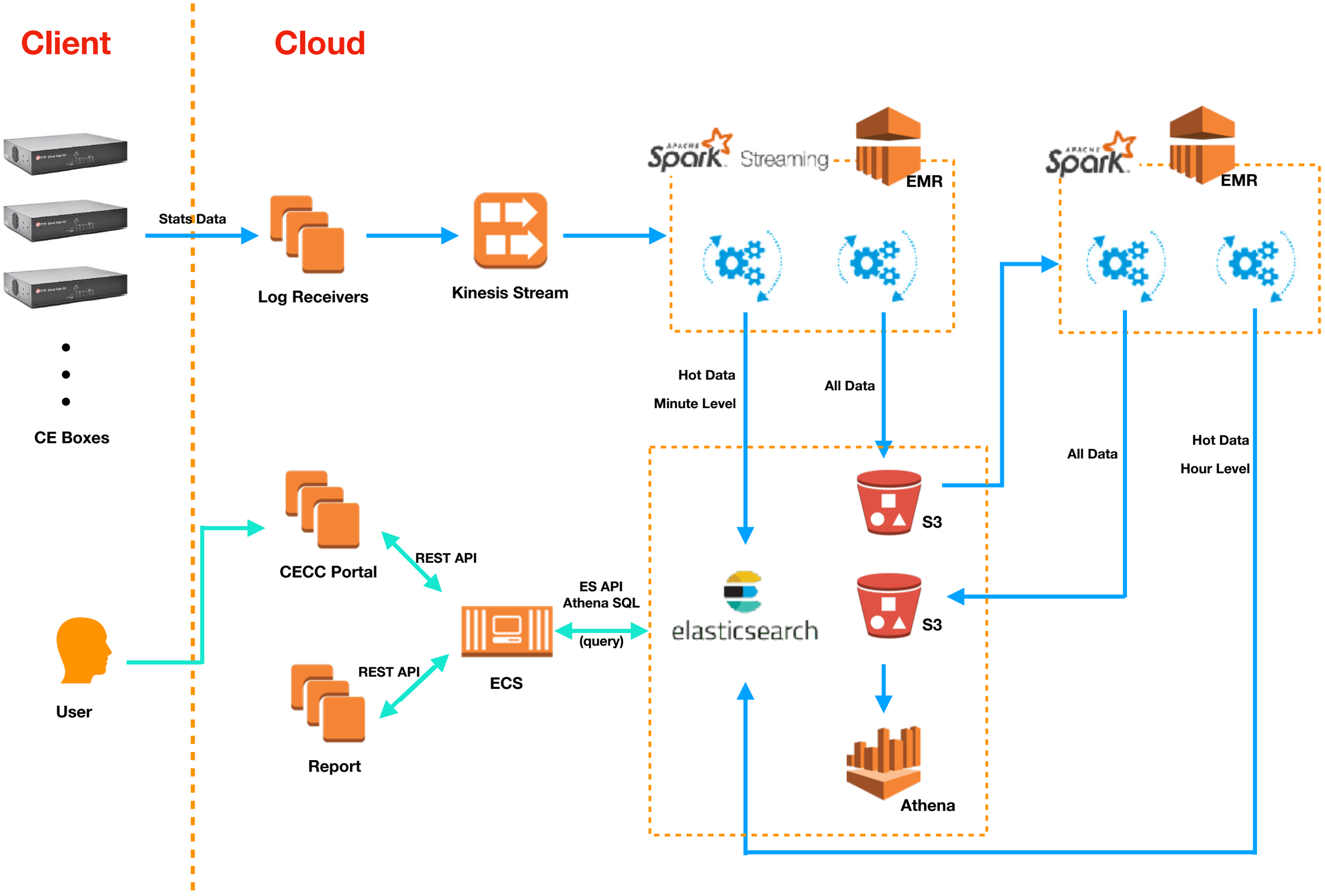


...

A scalable big data system built on AWS

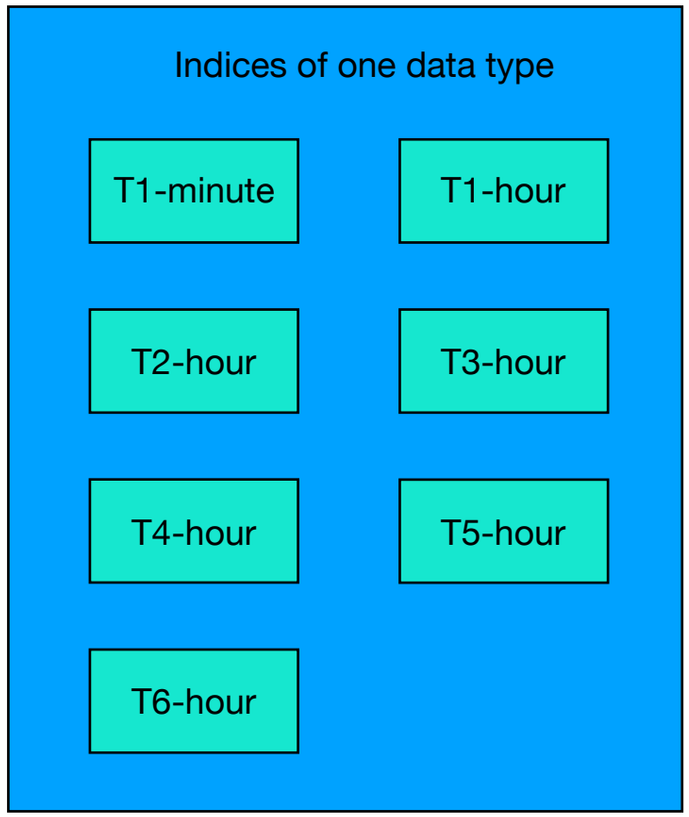
Project ElasticLog

- **1,300,000** data per minute, will increase to **5** * 1,300,000 at the end of this year.
- Provide second-level query for **6 month** data, most queries will hit the **1st month** data.
- Data MUST be **accurate**. No less no more.

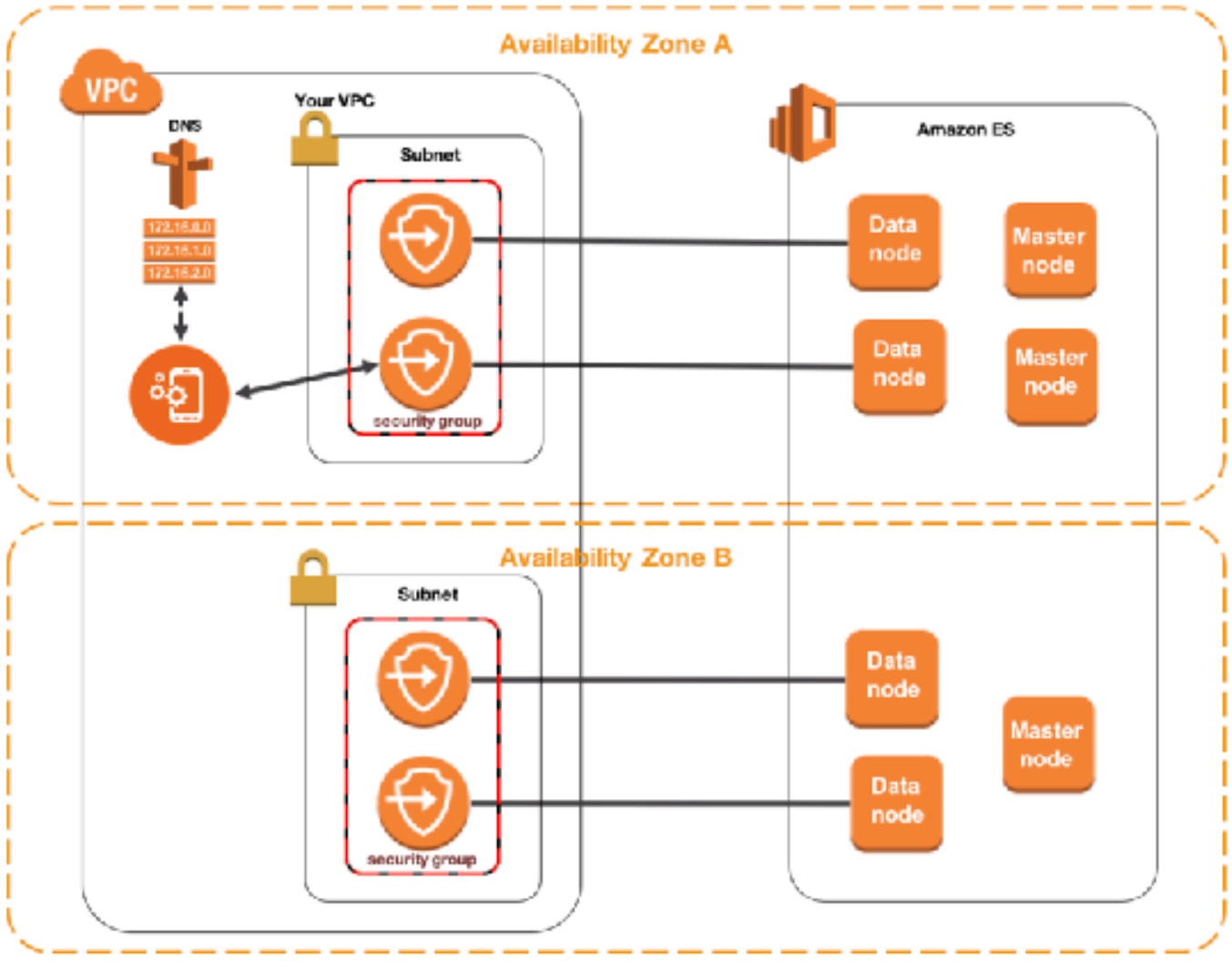


ES in ElasticLog

- Hot data storage and query
  - 1,300,000 data injection per minute
  - 1 month data
  - Second-level query
- One cluster
  - Build on AWS
  - 3 dedicated master nodes + 10 data nodes
  - Master node: M4.large, Data node: C4.2Xlarge
  - Cost: $7622 per month
- Roll index by week
  - 28 indices at most
  - 200 primary shards, 1 replica



ES on AWS



- Why is AWS ES?
  - All cloud services are on AWS
  - AWS ES is a managed service
- AWS ES
  - Easy to use
  - Zone awareness
  - VPC + Security Group
  - Blue/Green deployment
  - Monitor + Auto-Snapshot

Best Practices in ES

- Index Design
- Shard Allocation
- Fast Injection in Spark
- Data Deduplication

Index Design

One Index

- Unable to update some mapping settings, e.g., primary shard number
- Unable to scale out flexibly and rapidly
- Better for fixed/small data set

VS

Time-based Index

- How to determine the interval?
  - Data amount
  - Change frequency
  - Try weekly-split by default
- How to implement?
  - Index template

```
{
    "facet_internet_access_minute": {
        "template": "ce-index-access-v1-*",
        "order": 0,
        "settings": {
            "number_of_shards": 5
        },
        "aliases": {
            "{index}-query": {}
        },
        "mappings": {
            "es_doc": {
                "dynamic": "strict",
                "_all": {
                    "enabled": false
                },
                "_source": {
                    "enabled": false
                },
                "properties": {
                    "CLF_Timestamp": {
                        "type": "long"
                    },
                    "CLF_CustomerID": {
                        "type": "keyword"
                    },
                    "CLF_ClientIP": {
                        "type": "ip",
                        "ignore_malformed": true
                    }
                }
            }
        }
    }
}
```

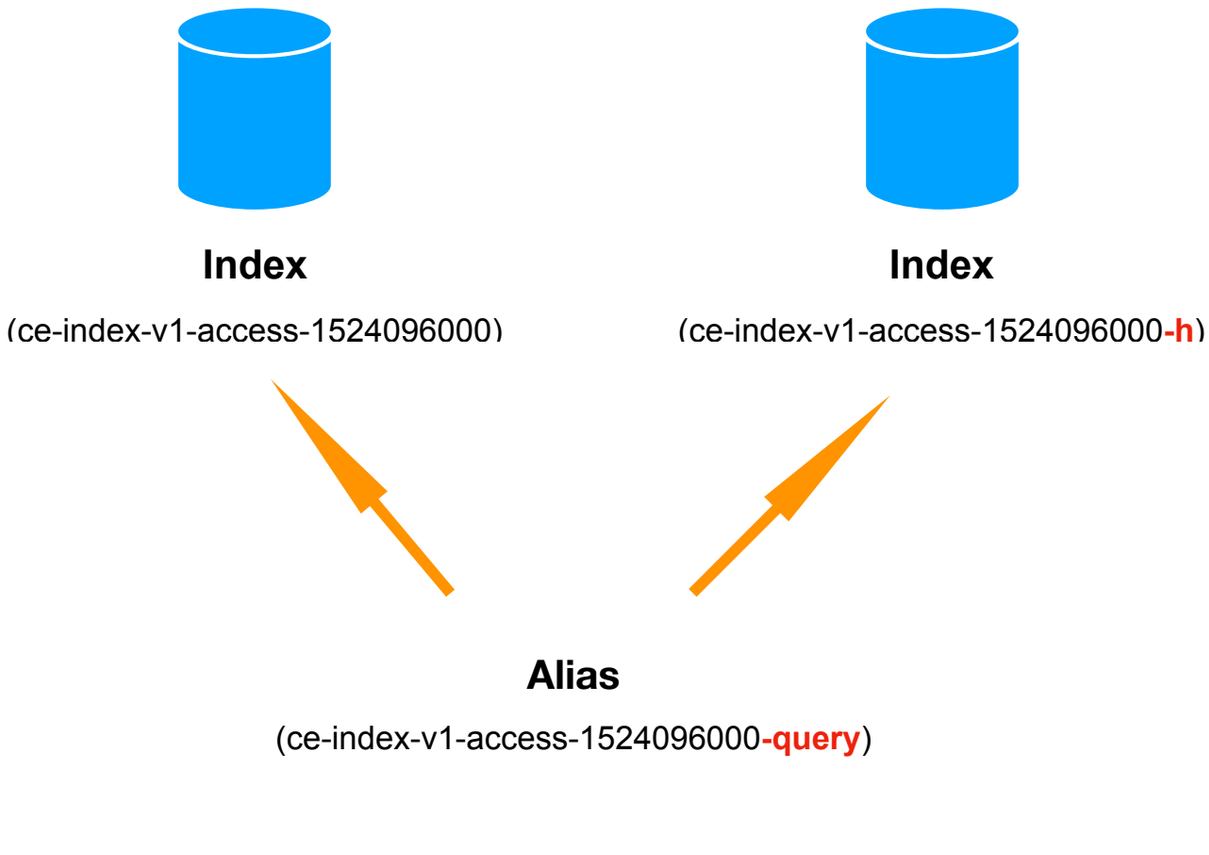
- DO NOT use multiple doc_type in one index
  - Fields that have the same names in different types must have the same mapping definition
  - Not support in 6.0
- Set _source=false
  - Suppose you only care about metric results, not raw documents content
  - Will save disk space and reduce IO
- Set _all=false
  - Suppose you know exactly what fields you want to query

```
{
    "facet_internet_access_minute": {
        "template": "ce-index-access-v1-*",
        "order": 0,
        "settings": {
            "number_of_shards": 5
        },
        "aliases": {
            "{index}-query": {}
        },
        "mappings": {
            "es_doc": {
                "dynamic": "strict",
                "_all": {
                    "enabled": false
                },
                "_source": {
                    "enabled": false
                },
                "properties": {
                    "CLF_Timestamp": {
                        "type": "long"
                    },
                    "CLF_CustomerID": {
                        "type": "keyword"
                    },
                    "CLF_ClientIP": {
                        "type": "ip",
                        "ignore_malformed": true
                    }
                }
            }
        }
    }
}
```

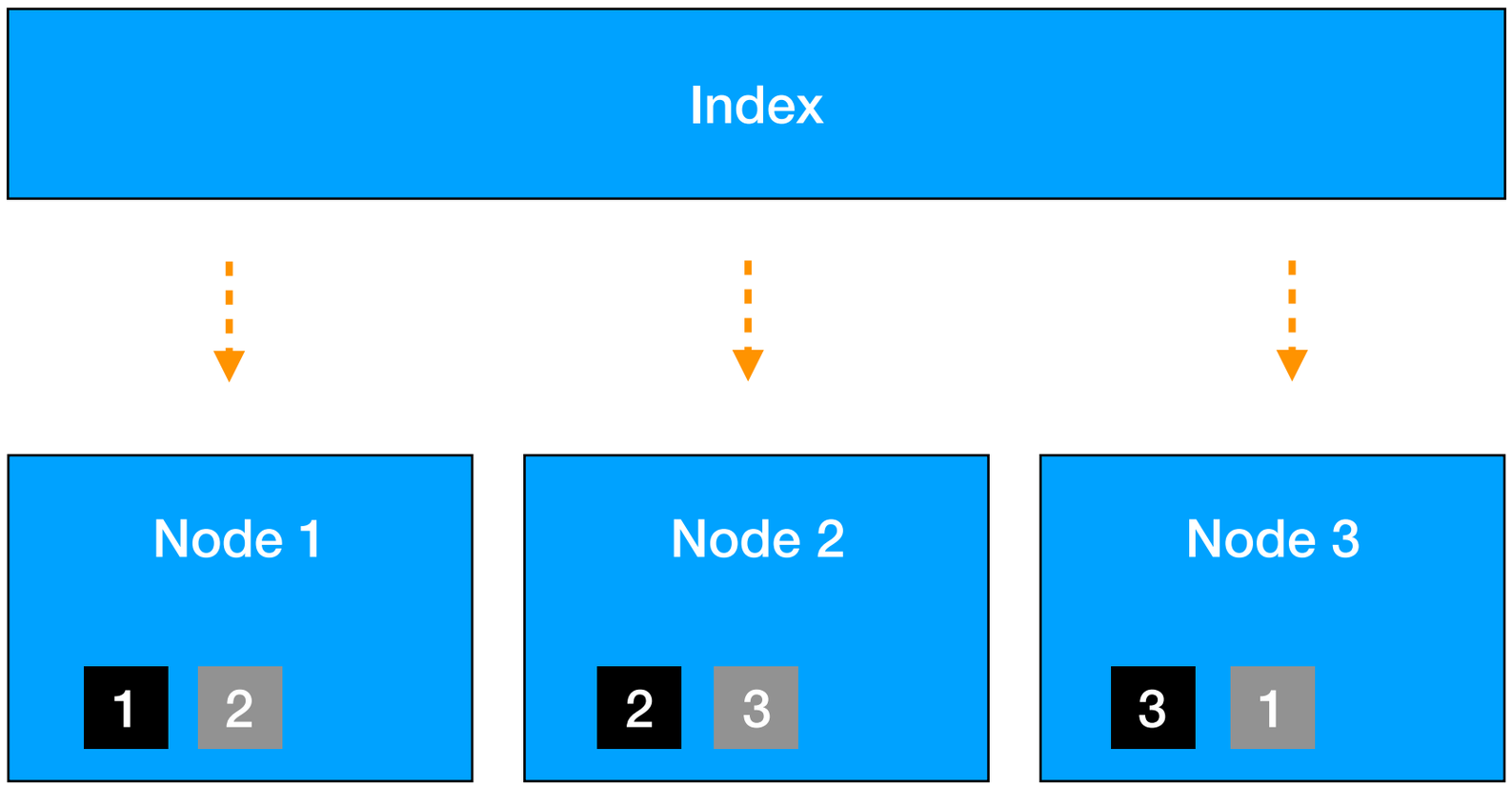
- Set dynamic=strict
  - Suppose your data is structured-data
  - Avoid dirty-data injection
- Set not_analyzed for String
  - Suppose you only care about full match
  - Improve injection performance and disk space
  - Use “keyword” in 5.x

```
{
    "facet_internet_access_minute": {
        "template": "ce-index-access-v1-*",
        "order": 0,
        "settings": {
            "number_of_shards": 5
        },
        "aliases": {
            "{index}-query": {}
        },
        "mappings": {
            "es_doc": {
                "dynamic": "strict",
                "_all": {
                    "enabled": false
                },
                "_source": {
                    "enabled": false
                },
                "properties": {
                    "CLF_Timestamp": {
                        "type": "long"
                    },
                    "CLF_CustomerID": {
                        "type": "keyword"
                    },
                    "CLF_ClientIP": {
                        "type": "ip",
                        "ignore_malformed": true
                    }
                }
            }
        }
    }
}
```

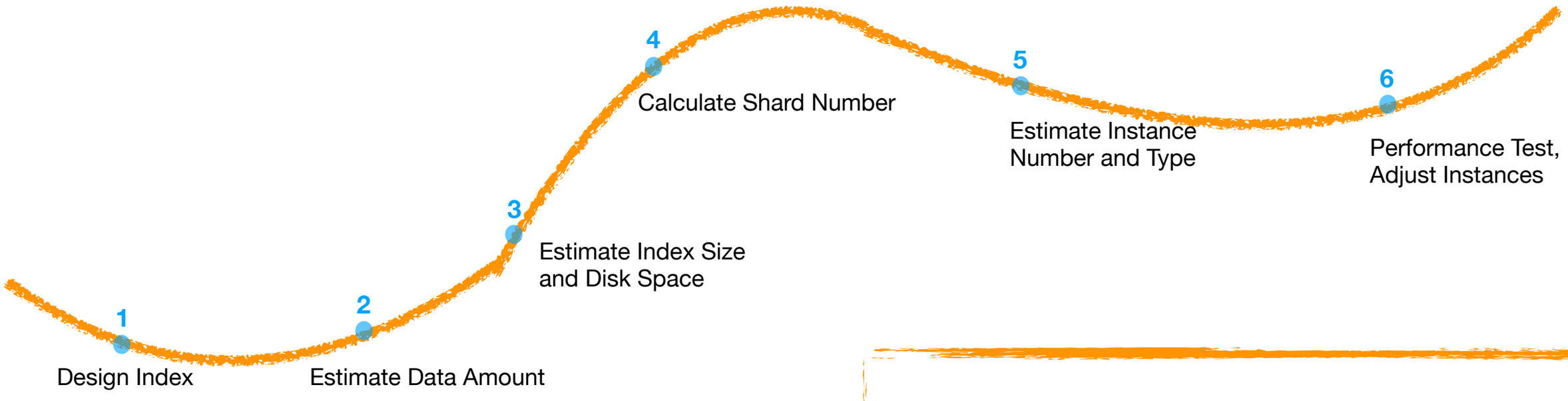
How to switch from one index to the other one without downtime?



Shard Allocation

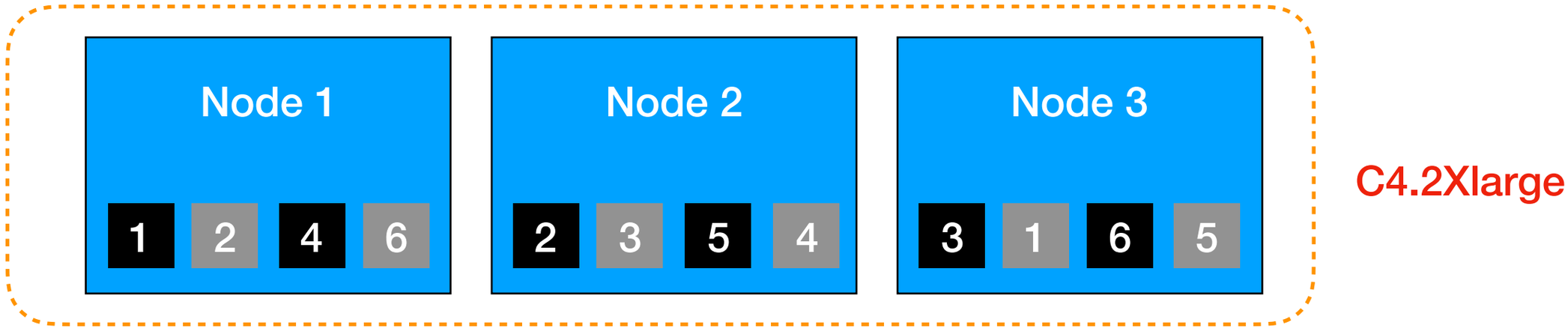


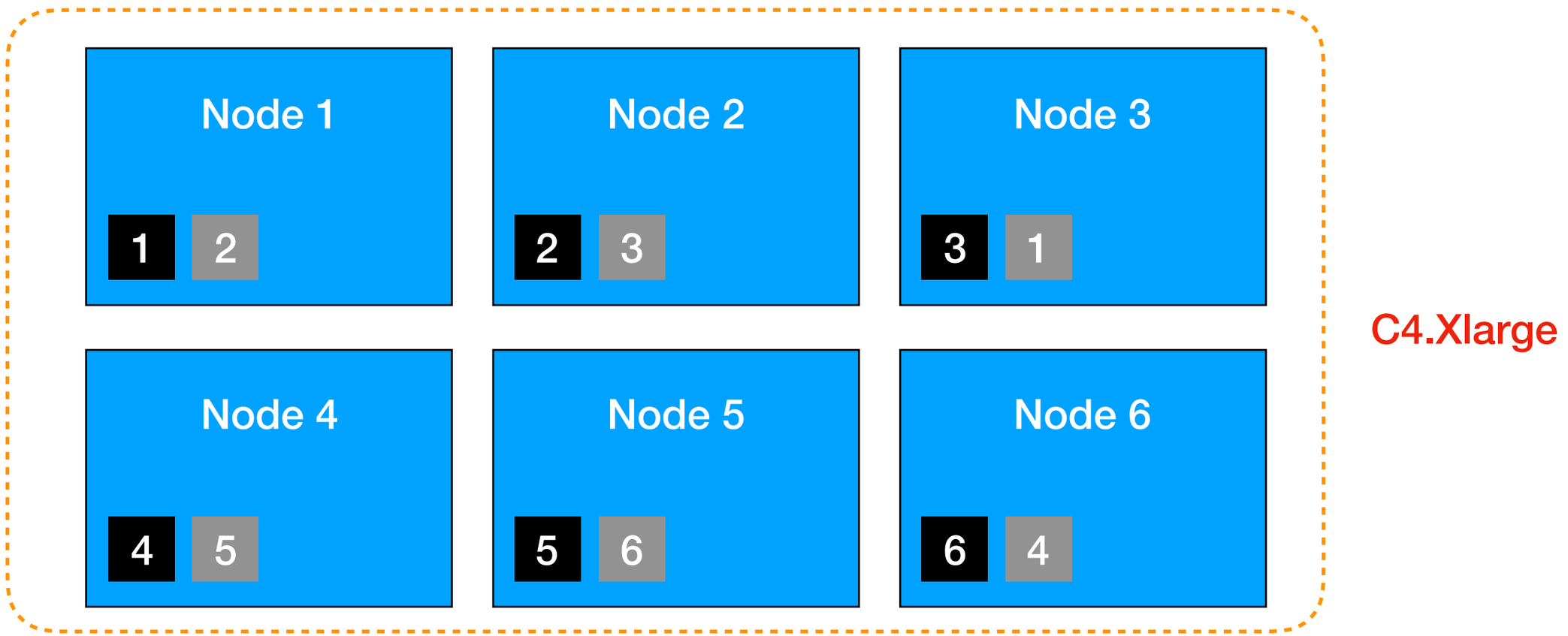
Shard Allocation



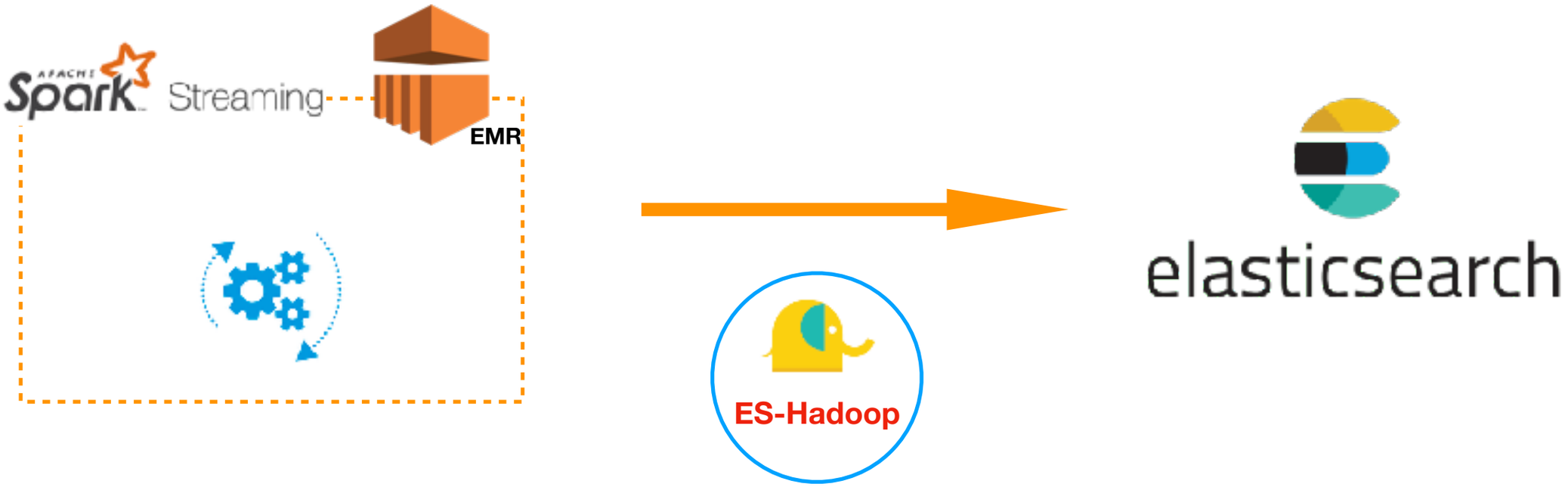
- Each shard size should be less than 30GB
- Shard Number = k * Data Nodes Number (k = a small integer)
- Suppose you have an small index, and you have enough instances in the cluster, try to use default shard number

Shard Allocation

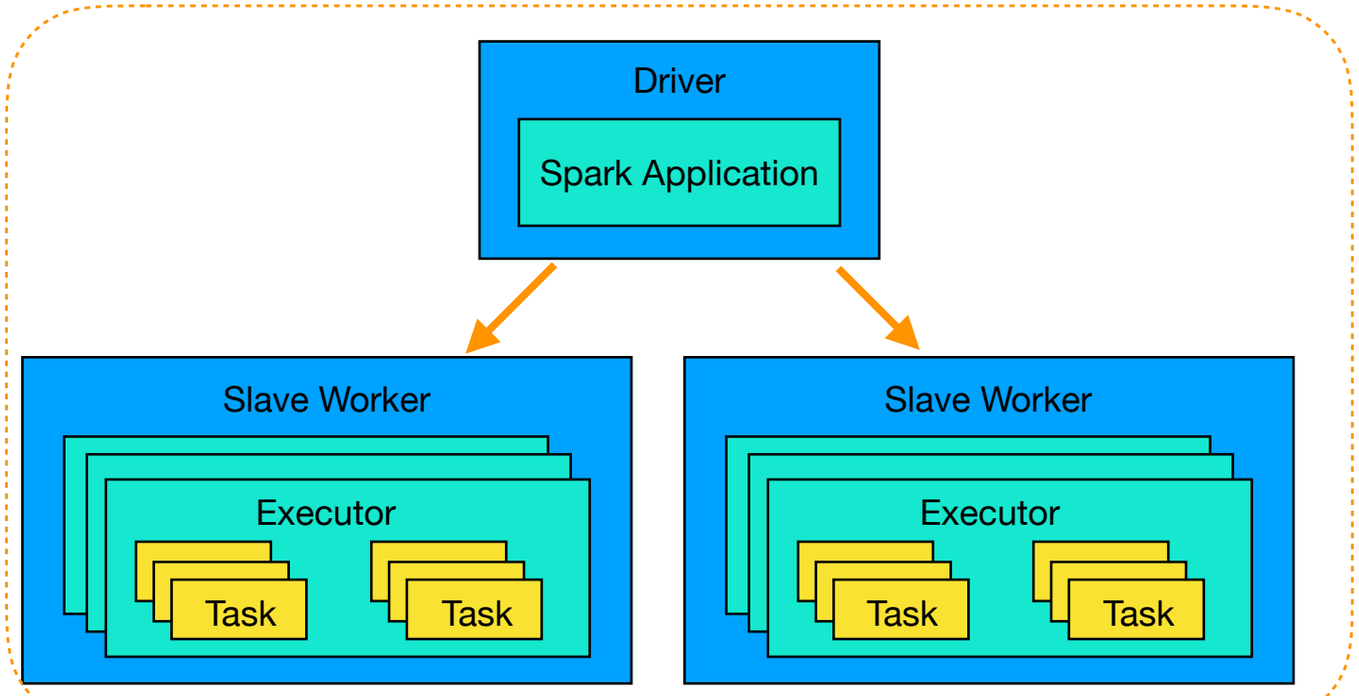




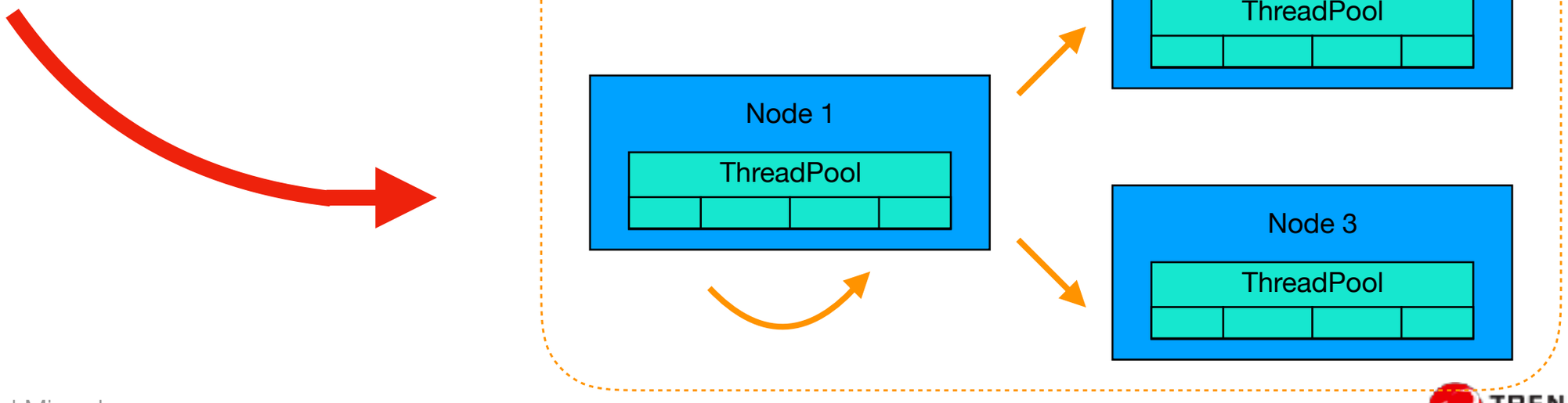
Fast Injection in Spark



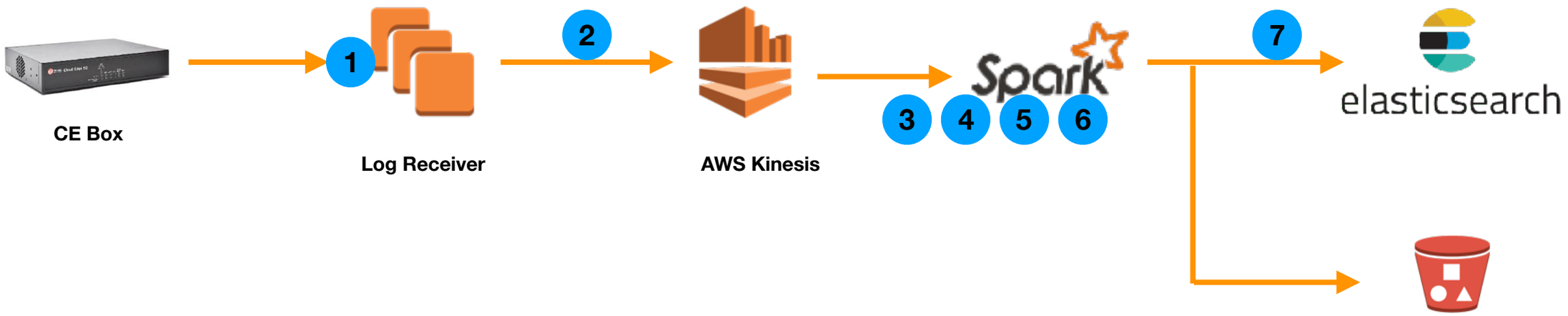
Fast Injection in Spark

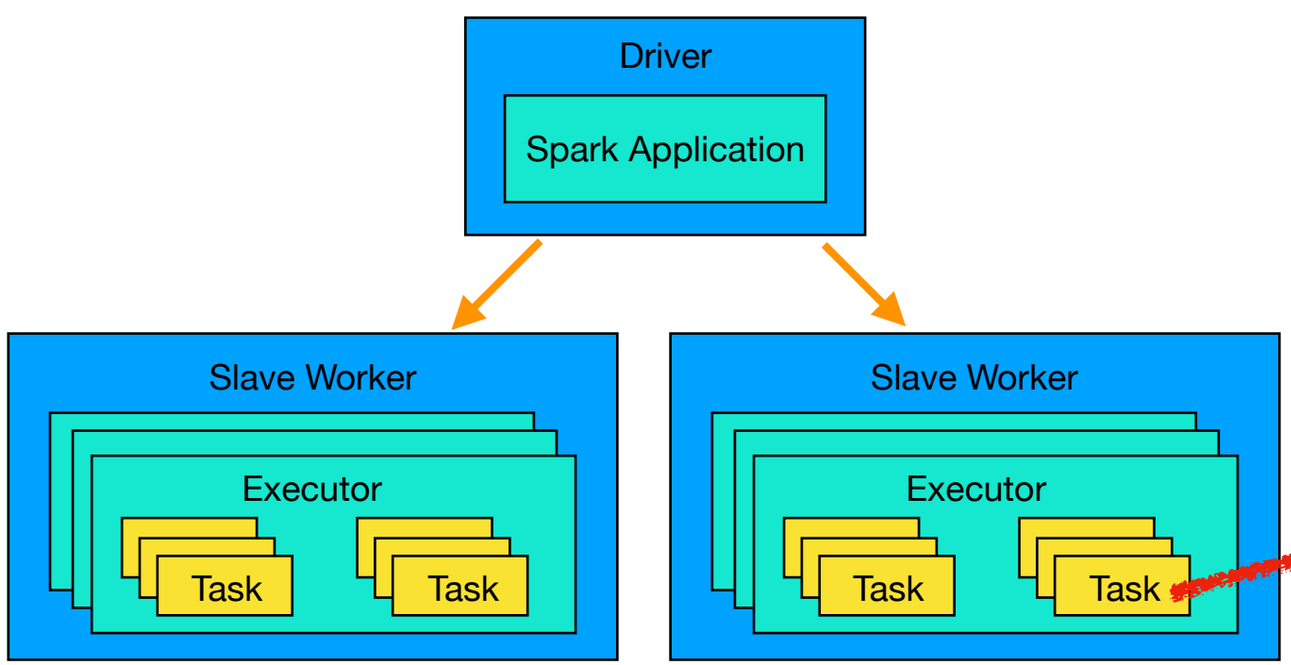


Number of Spark Task
=
Number of vCores



Data Deduplication





Task will retry for 3 times for any failure

Data Deduplication

- Use a custom unique ID
- Use aggregation to find out duplication and delete those documents
- Do distinct query

Data Deduplication

- Step 1: Add a “hash” field in all documents
- Step 2: Check duplication

```
curl -XGET http://stg-elasticlog.ap-northeast-1.es.amazonaws.com/ce-index-v1-access-1524096000/_search?pretty -d '
{
    "size": 0,
    "aggs": {
        "duplicate": {
            "terms": {
                "field": "hash",
                "min_doc_count": 2,
                "size": 5000
            },
            "aggs": {
                "documents": {
                    "top_hits": {
                        "size": 2
                    }
                }
            }
        }
    }
}
`
```

- Step 3: Bulk delete

- Do not affect injection
- Can be asynchronous

Data Deduplication

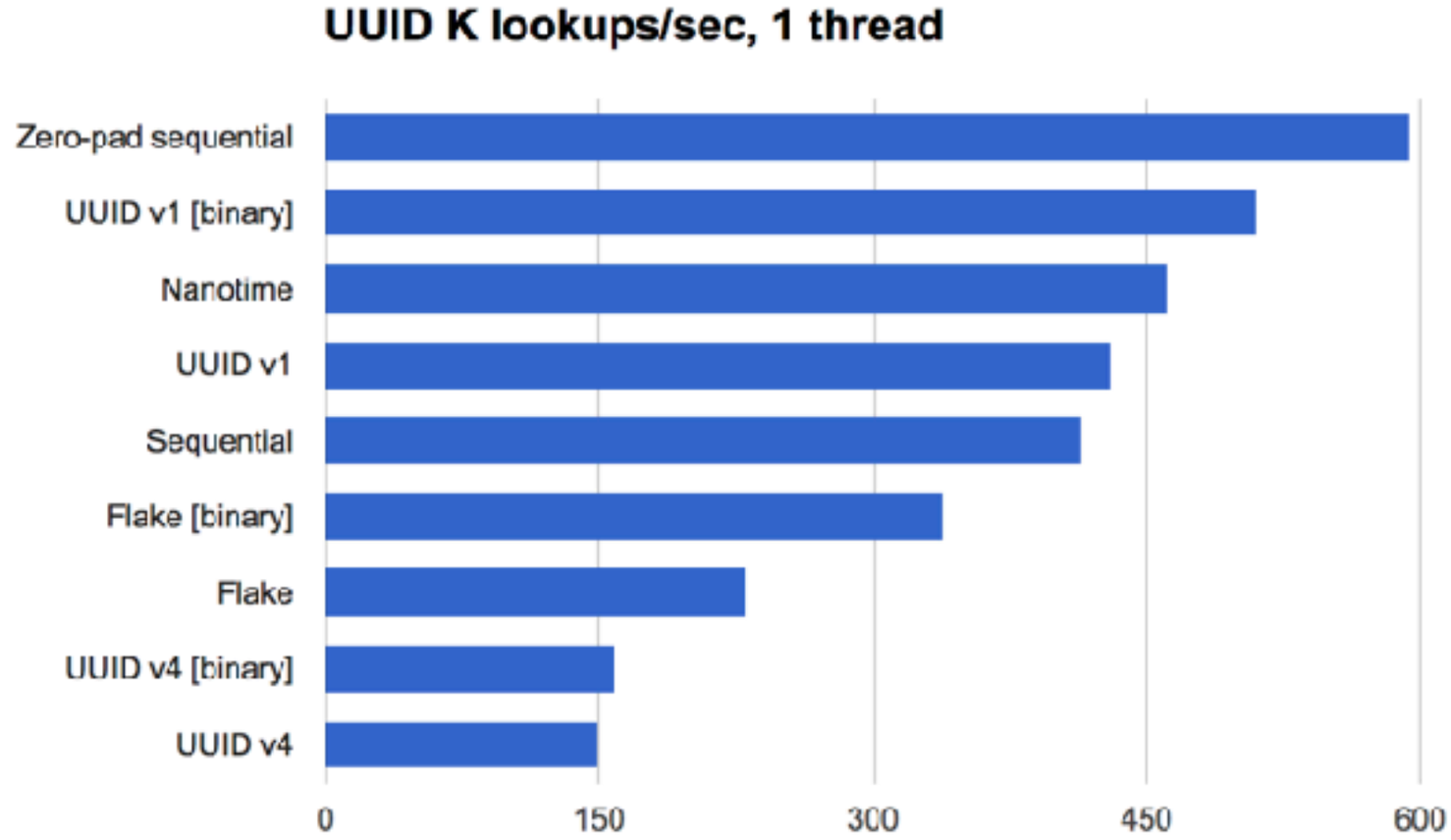
- Storage size increased heavily
- Will be slower or even failed to do aggregation when data amount is more than 0.3 billion

Field “hash” is unique!

- Unfriendly to compression
- High-Cardinality

Data Deduplication

- Auto-generated IDs are 20 character long, URL-safe, Base64-encoded GUID strings
- For custom ID, try to pick up an ID that is friendly to Lucene



(http://blog.mikemccandless.com/2014/05/choosing-fast-unique-identifier-uuid.html)

Q&A

elastic 中文社区

专业、垂直、纯粹的 Elastic 开源技术交流社区

https://elasticsearch.cn/