OMNISHORE

Introduction to Elastic Stack

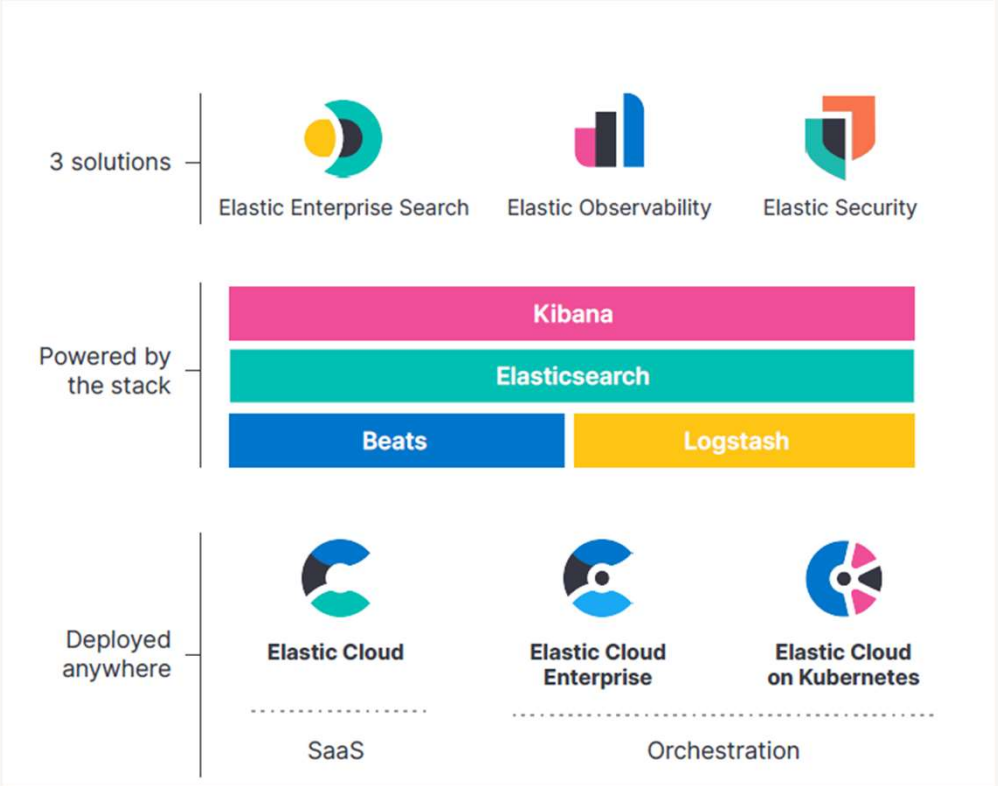
Mr. Iyanou Eraste AKANDE
Elastic Certified Engineer
Data Engineer at Synaptique Maghreb

The Beginning

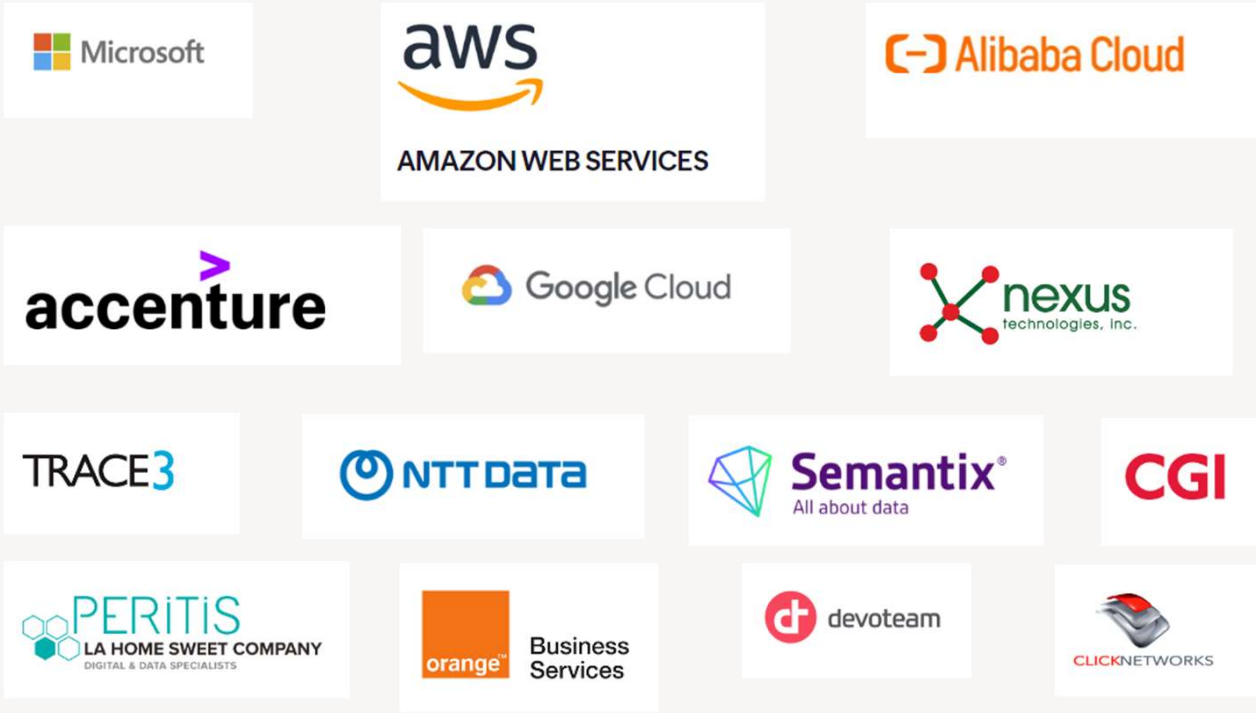
Lucene	Compass	Elasticsearch
1999 Doug Cutting Search Engine in Java	2004 Shay Banon Search Product in Java, scalability included, built on Lucene	2010 Shay Banon Search product built on Compass concepts, distributed and integrated with other language

Elastic Stack Components

- Elasticsearch : NoSQL Database
- Kibana : Analytics and Visualization platform
- Logstash : Server side data processing pipeline
- Beats : Data Shippers (Filebeat, Metricbeat, Heartbeat, Elastic Agents)



Elastic Partners



https://partners.elastic.co/findapartner/

Elastic Stack Use Cases

- **Log Management and Analysis**: One of the primary use cases for Elastic Stack is log management and analysis. Organizations use it to collect, parse, index, and analyze logs generated by various systems, applications, and services.
- **Security Information and Event Management (SIEM)**: Elastic Stack can be used as a SIEM solution to collect, analyze, and visualize security-related data such as logs, network traffic, and system events.
- **Real-time Monitoring and Alerting**: Elastic Stack can be utilized for real-time monitoring of infrastructure, applications, and services. By collecting metrics, logs, and events in real-time, organizations can set up alerts and notifications.
- **Application Performance Monitoring (APM)**: Elastic APM, an extension of the Elastic Stack, is used for monitoring the performance of applications and microservices.

Elastic Stack Use Cases

- **Business Analytics and Insights**: Elastic Stack can be leveraged for business analytics and data visualization purposes. By indexing and analyzing large volumes of data from different sources, organizations can gain valuable insights into customer behavior, market trends, and operational performance.

- **Search and Information Retrieval**: Elasticsearch supports full-text search, fuzzy search, geospatial search, and faceted navigation, making it suitable for building search-driven applications and platforms.

- **Data Exploration and Discovery**: Elastic Stack can be used for data exploration and discovery in various domains such as scientific research, e-commerce, and content management.

Elasticsearch

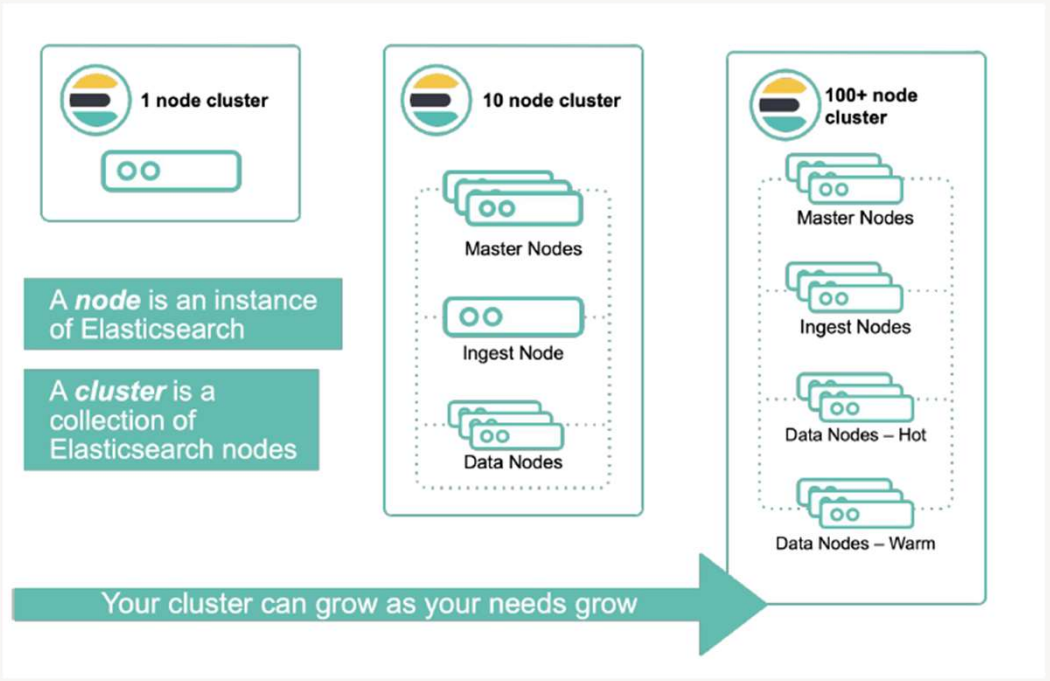
NoSQL Database

first_ame	name	gender	city	country
Ali	KOZNI	M	Rabat	Morocco
Achille	BADRE	F	Tetouan	Morocco


```
{
“first_name” : “Ali”,
“name” : “KOZNI”,
“gender” : “M”,
“city” : “Rabat”,
“country” : “Morocco”
},
{
“first_name” : “Achille”,
“name” : “BADRE”,
“gender” : “F”,
“city” : “Tetouan”,
“country” : “Morocco”
}
```

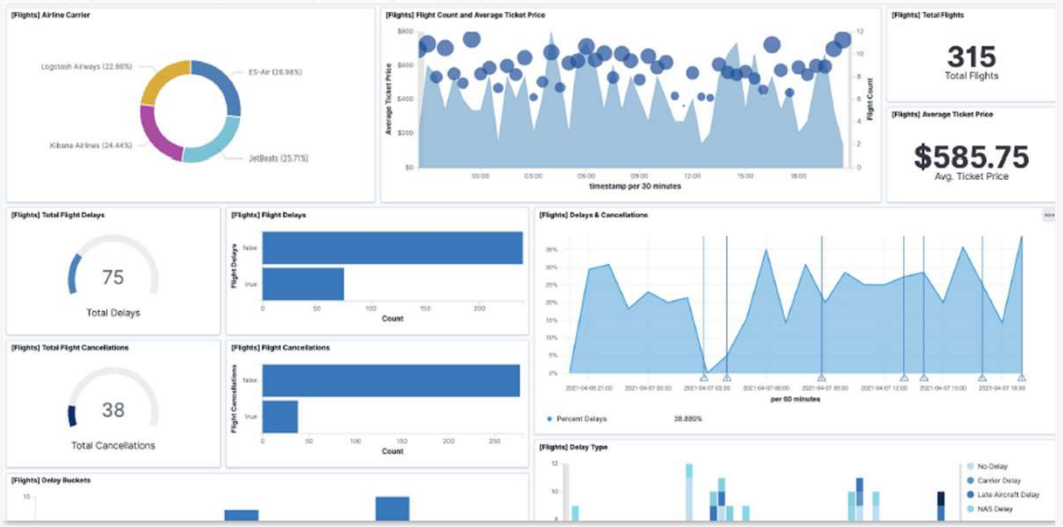
Elasticsearch

Distributed and Horizontally Scalable



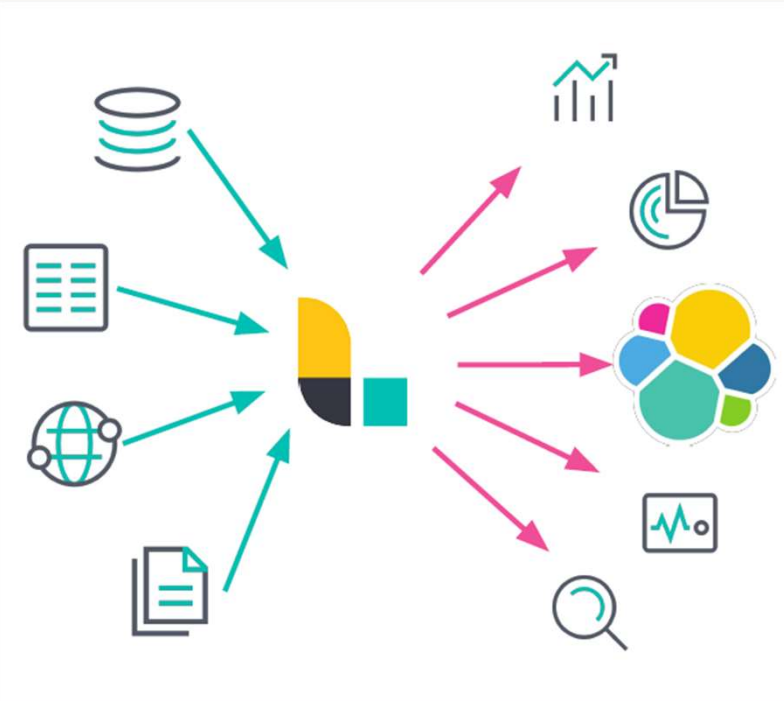
Kibana

Analytics and Visualization Platform



Logstash

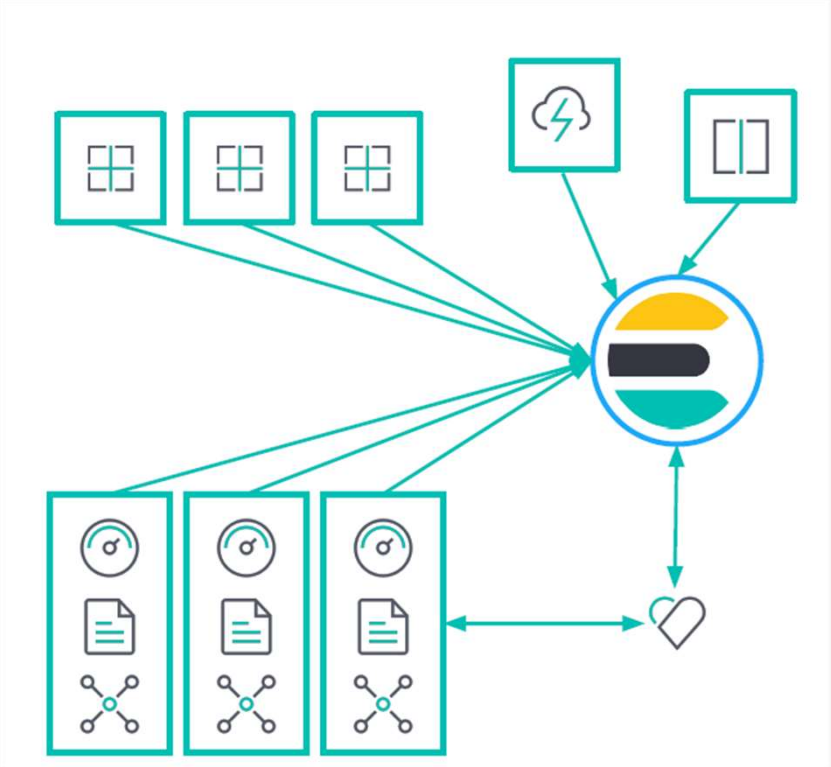
Data Processing Pipeline



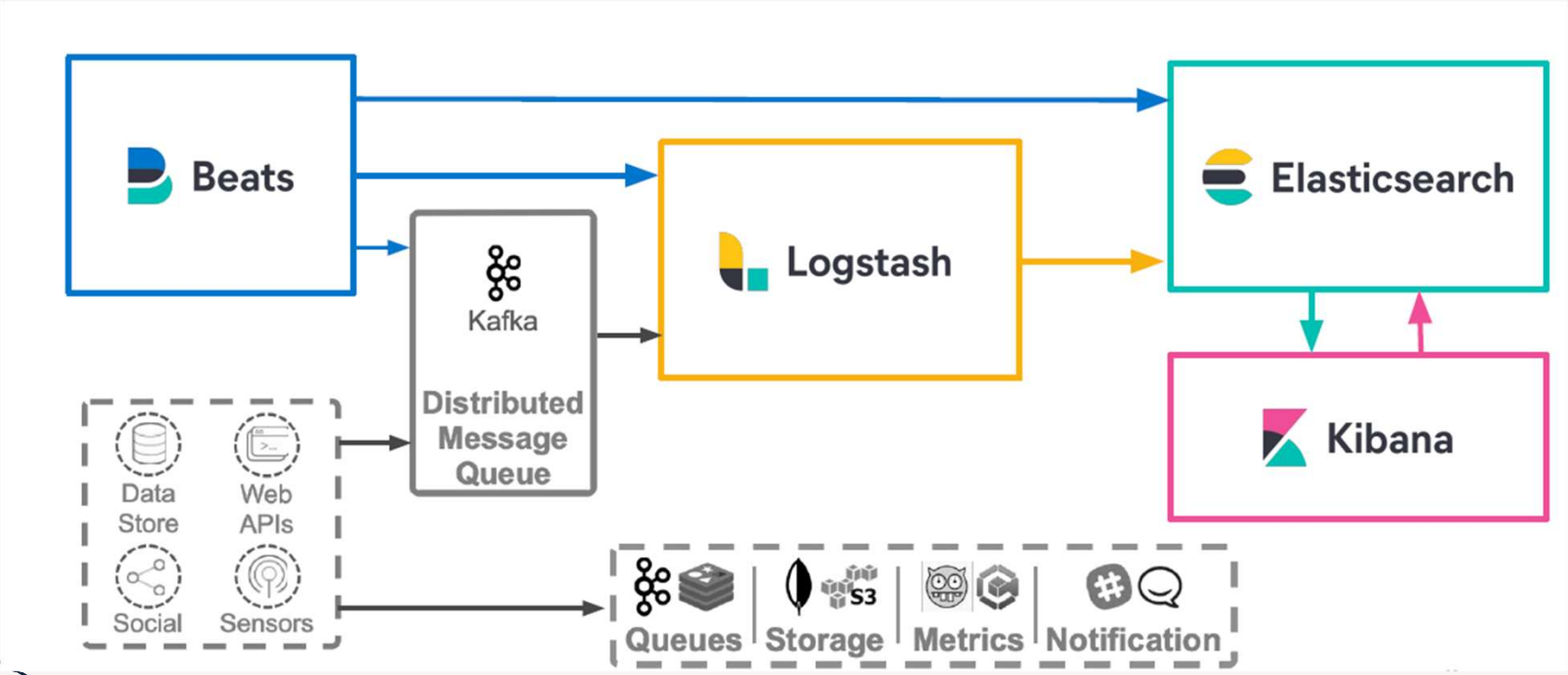
Beats

Open Source Data Shippers

Filebeat, Metricbeat, Heartbeat,
Packetbeat, Winlogbeat, etc.



Elastic Stack Architecture



Questions

Thank you

Mr. Iyanou Eraste AKANDE

eraste.akande@gmail.com

OMNISHORE

Introduction to Filebeat

Mr. Iyanou Eraste AKANDE
Elastic Certified Engineer
Data Engineer at Synaptique Maghreb

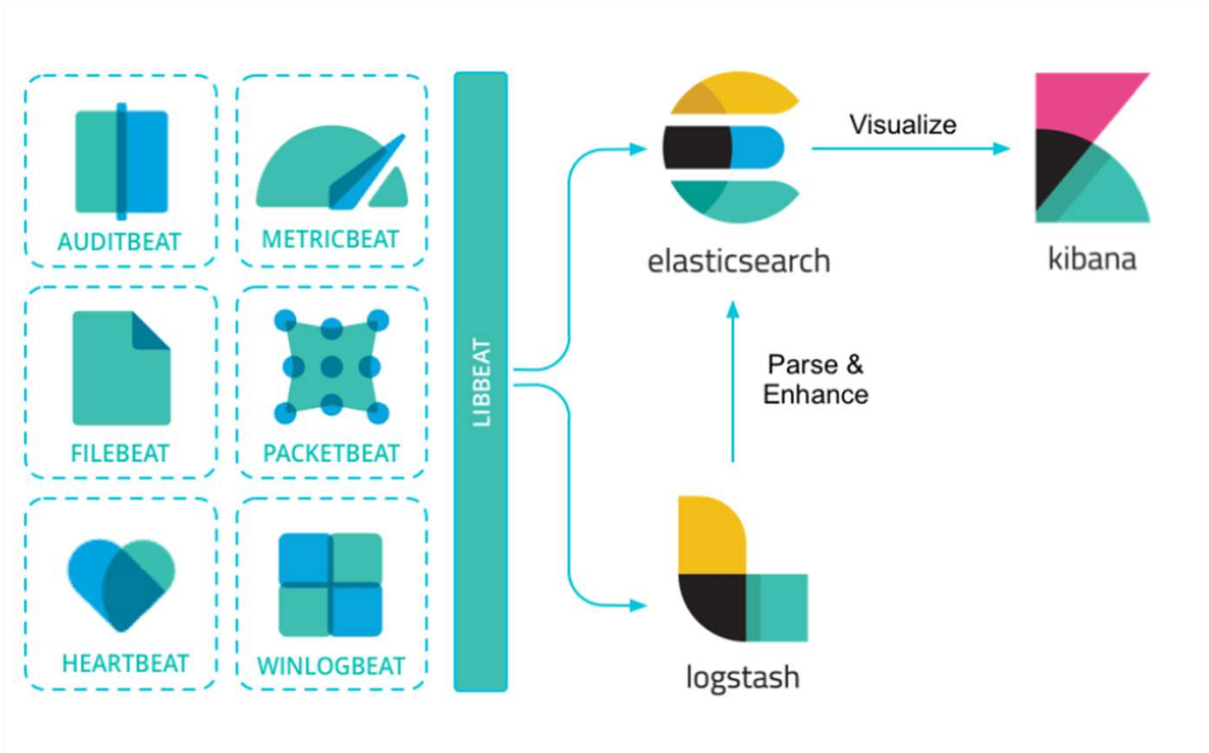
Beats

Use Case

Auditbeat	Audit data
Filebeat	Log files and journal
Functionbeat	Cloud data
Heartbeat	Availability of servers and services
Metricbeat	Metrics and statistics of instances
Packetbeat	Network Traffic
Winlogbeat	Window event logs

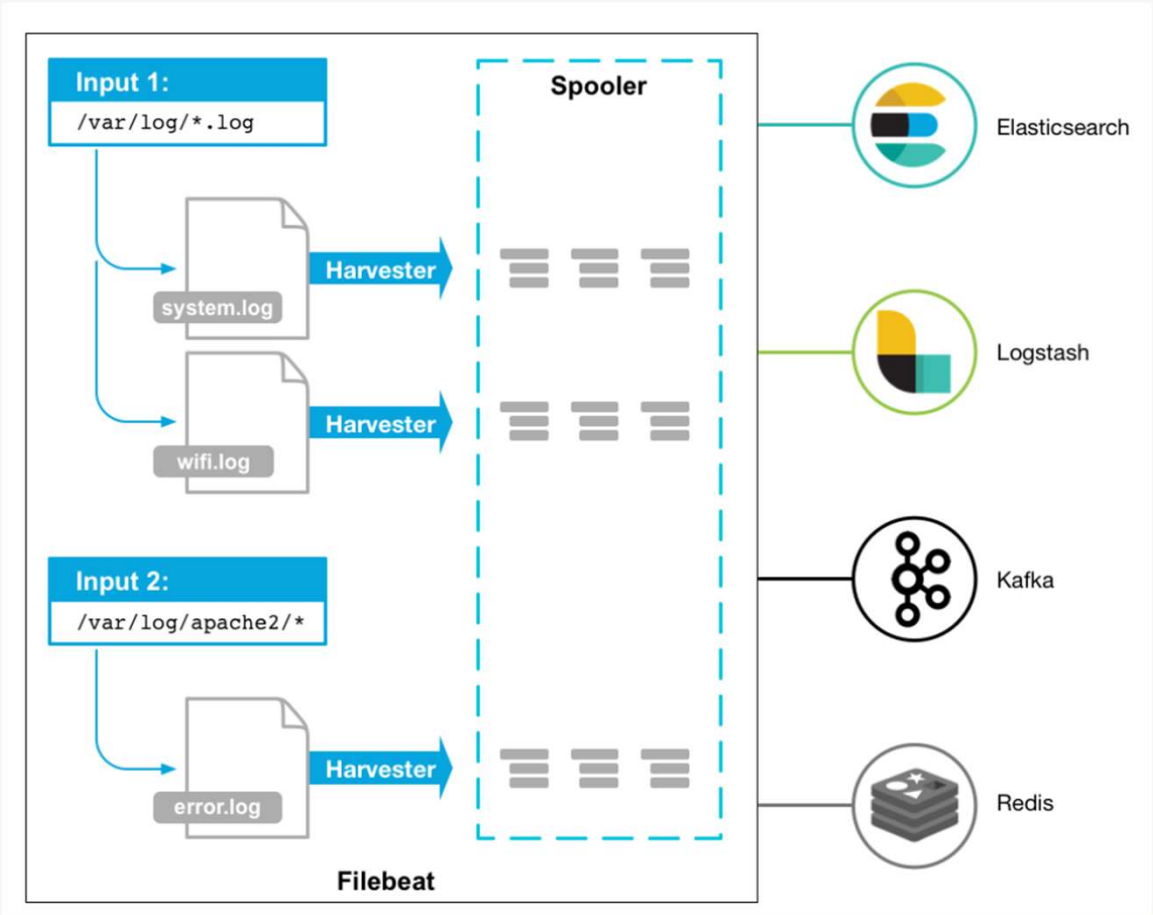
Beats

Architecture



Filebeat

Architecture



Filebeat

Filebeat.yml >> Input

Input Types

- AWS CloudWatch
- AWS S3
- Azure Event Hub
- Azure Blob Storage
- CEL
- Cloud Foundry
- CometD
- Container
- Entity Analytics
- filestream
- GCP Pub/Sub
- HTTP Endpoint
- HTTP JSON
- journald
- Kafka
- Log (deprecated in 7.16.0, use filestream)
- MQTT
- NetFlow
- Office 365 Management Activity API
- Redis
- Stdin
- Syslog
- TCP
- UDP
- Google Cloud Storage

Filebeat

Filebeat.yml >> Input

Filestream Input

https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-input-filestream.html

```
filebeat.inputs:
- type: filestream
  id: my-filestream-id
  paths:
    - /var/log/messages
    - /var/log/*.log
```

```
filebeat.inputs:
- type: filestream (1)
  id: my-filestream-id
  paths:
    - /var/log/system.log
    - /var/log/wifi.log
- type: filestream (2)
  id: apache-filestream-id
  paths:
    - "/var/log/apache2/*"
  fields:
    apache: true
```

```
filebeat.inputs:
- type: filestream
  ...
  parsers:
    - ndjson:
        target: ""
        message_key: msg
```

Filebeat

Filebeat.yml >> Input

Filestream Input

https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-input-filestream.html

```
filebeat.inputs:
- type: filestream
  ...
  prospector.scanner.exclude_files: ['\.gz$']
  prospector.scanner.include_files: ['^/var/log/.*']
```

```
filebeat.inputs:
- type: filestream
  ...
  include_lines: ['^ERR', '^WARN']
  exclude_lines: ['^DBG']
```

Filebeat

Filebeat.yml >> Input

Filestream Input

https://www.elastic.co/guide/en/beats/filebeat/current/multiline-examples.html

Problem

```
[beat-logstash-some-name-832-2015.11.28] IndexNotFoundException[no such index]
    at
org.elasticsearch.cluster.metadata.IndexNameExpressionResolver$WildcardExpressionResolver.resolve(IndexNameExpressionResolver
.java:566)
    at org.elasticsearch.cluster.metadata.IndexNameExpressionResolver.concreteIndices(IndexNameExpressionResolver.java:133)
    at org.elasticsearch.cluster.metadata.IndexNameExpressionResolver.concreteIndices(IndexNameExpressionResolver.java:77)
    at org.elasticsearch.action.admin.indices.delete.TransportDeleteIndexAction.checkBlock(TransportDeleteIndexAction.java:75)
```

Solution

```
parsers:
- multiline:
    type: pattern
    pattern: '^\['
    negate: true
    match: after
```

Filebeat

Filebeat.yml >> Processors

Processors Types

https://www.elastic.co/guide/en/beats/filebeat/current/defining-processors.html

- add_docker_metadata
- add_fields
- add_host_metadata
- add_id
- add_locale
- append
- convert
- copy_fields
- decode_base64_field
- decode_csv_fields
- decode_duration
- decode_json_fields
- decode_xml
- decompress_gzip_field
- dissect
- drop_event
- drop_fields
- extract_array
- fingerprint
- include_fields
- move-fields
- rename
- replace
- script
- timestamp
- translate_sid

Filebeat

Filebeat.yml >> Processors

Processors Conditions

https://www.elastic.co/guide/en/beats/filebeat/current/defining-processors.html

- equals
- contains
- regexp
- range
- network
- has_fields
- or
- and
- not

Filebeat

Filebeat.yml >> Processors

Examples

```
processors:
  - add_fields:
      target: project
      fields:
        name: myproject
        id: '5747348851209524 59'
```

```
processors:
  - convert:
      fields:
        - {from: "src_ip", to: "source.ip", type: "ip"}
        - {from: "src_port", to: "source.port", type: "integer"}
      ignore_missing: true
      fail_on_error: false
```

Filebeat

Filebeat.yml >> Processors

Examples

```
processors:
  - rename:
      fields:
        - from: "a.g"
          to: "e.d"
      ignore_missing: false
      fail_on_error: true
```

```
- replace:
    fields:
      - field: "file.path"
        pattern: "/usr/"
        replacement: "/usr/local/"
    ignore_missing: false
    fail_on_error: true
```

```
processors:
  - drop_event:
      when:
        equals:
          log_status: "INFO"
```

Filebeat

Filebeat.yml >> Processors

Examples

Problem

"321 - App01 - WebServer is starting"
"321 - App01 - WebServer is up and running"
"321 - App01 - WebServer is scaling 2 pods"
"789 - App02 - Database is will be restarted in 5 minutes"
"789 - App02 - Database is up and running"
"789 - App02 - Database is refreshing tables"

Solution

```
processors:
  - dissect:
      tokenizer: '"%{service.pid|integer} - %{service.name} - %{service.status}"'
      field: "message"
      target_prefix: ""
```

Filebeat

Filebeat.yml >> Output

Output Types

- Elasticsearch Service
- Elasticsearch
- Logstash
- Kafka
- Redis
- File
- Console

Filebeat

Filebeat.yml >> Output

Elasticsearch Output

```
output.elasticsearch:
  hosts: ["https://localhost:9200"]
  index: "your_index_name"
  ssl.enabled: true  # Enable SSL/TLS
  ssl.certificate_authorities: ["/path/to/your/ca.crt"]
  username: "your_username"  # Elasticsearch username
  password: "your_password"  # Elasticsearch password
```

Logstash Output

```
output.logstash:
  hosts: ["localhost:5044"]
```

Filebeat

Modules

Modules List

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

- ActiveMQ module
- Apache module
- AWS module
- Azure module
- Cisco module
- CoreDNS module
- Elasticsearch module
- F5 module
- Fortinet module
- Google Cloud module
- Google Workspace module
- IBM MQ module
- Kafka module
- Kibana module
- Logstash module
- Microsoft module
- MongoDB module
- MSSQL module
- MySQL module
- MySQL Enterprise module
- Osquery module
- PostgreSQL module
- RabbitMQ module
- Redis module
- System module
- ZooKeeper module
- Zoom module
- etc

Filebeat

Modules

Configure modules.d file

Elasticsearch – elasticsearch.yml

```
server:
  enabled: true
  var.paths:
    - /var/log/elasticsearch/*.log
    - /var/log/elasticsearch/*_server.json
```

Kafka – kafka.yml

```
- module: kafka
  log:
    enabled: true
    var.paths:
      - "/path/to/logs/controller.log*"
      - "/path/to/logs/server.log*"
      - "/path/to/logs/state-change.log*"
      - "/path/to/logs/kafka-*.log*"
```

Filebeat

Execution

```
cd filebeat_directory

filebeat modules enable module_name

filebeat setup –e

filebeat -e
```

Questions

Thank you

Mr. Iyanou Eraste AKANDE

eraste.akande@gmail.com

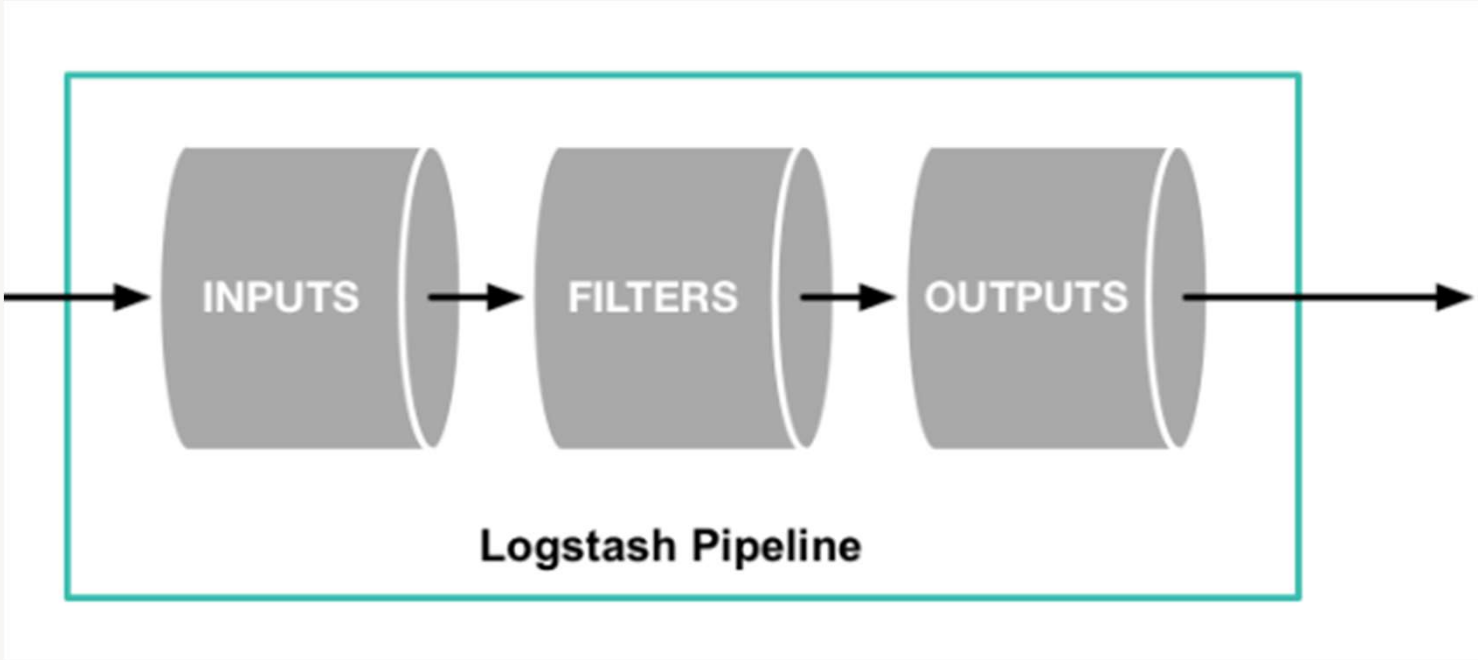
OMNISHORE

Introduction to Logstash

Mr. Iyanou Eraste AKANDE
Elastic Certified Engineer
Data Engineer at Synaptique Maghreb

Configuration

Architecture



Configuration

Sample file

```
# Sample Logstash configuration for creating a simple
# Beats -> Logstash -> Elasticsearch pipeline.
input {
        beats {    port => 5044  }
}
filter {}
output {
        elasticsearch {
                hosts => ["http://localhost:9200"]
                index => "filebeat-%{+YYYY.MM.dd}"
                user => "elastic"
                password => "changeme"
         }
}
```

Configuration

Input

```
input {
  beats {
    port => 5044
  }
}
```

https://www.elastic.co/guide/en/logstash/current/input-plugins.html

Configuration

Input

```
input {

  file {
    path => ["/path/samples/input-file.csv"]
    start_position => "beginning"
    sincedb_path => "/path/log.sincedb"
    file_completed_action => "log"
    file_completed_log_path => "/path/log.processed"
    mode => read
  }

}
```

https://www.elastic.co/guide/en/logstash/current/input-plugins.html

Configuration

Filter

```
csv {
    source => "message"
    separator => ";"
    columns => ["field1", "field2", "field3", "field4"]
}

mutate {
  add_field => { "country" => "Morocco" }
}

mutate {
  convert => { "money" => "float" }
}

date {
      match => ["@timestamp", "yyyyMMdd HH:mm:ss"]
      timezone => "UTC"
      target => "@timestamp"
}

mutate {
  rename => {"Lieu de Naissance" => "birth_city"}
}
```

https://www.elastic.co/guide/en/logstash/current/filter-plugins.html

Configuration

Filter

```
mutate { remove_field => [ "message", "@version", "host", "path", "event"]}

if ([code] =~ /.+/ ) {
translate {
    source => "[code]"
    target => "[value]"
    dictionary_path => "/path/dicts/code.yml"
    exact => true
    override => true
    regex => false
  }
}

translate {
        source => "[code]"
        target => "[code]"
        override => true
        exact => true
        dictionary => {
            "0" => "Normal"
            "1" => "Free"
        }
}
```

https://www.elastic.co/guide/en/logstash/current/filter-plugins.html

Configuration

Filter

```
ruby {
    code => "
      event.set('salary', event.get('salary') / event.get('total_number'));
      event.set('id', '2024' + event.get('id'));
    "
}
```

```
# location : "region=Rabat,country=Morocco"
kv {
    source => "location"
    field_split => ","
    value_split => "="
}

if [loglevel] == "debug" {
    drop { }
}
```

https://www.elastic.co/guide/en/logstash/current/filter-plugins.html

Configuration

Filter

```
mutate {
    add_field => { "hostname" => "LENOVO" }
    #description : "The//name//of//the city"
    gsub => [
          "description", "//", " "
        ]
    uppercase => [ "fieldname" ]
    lowerercase => [ "fieldname" ]
    copy => { "source_field" => "dest_field" }
}

# message : "55.3.244.1 GET /index.html 15824 0.043"
grok {
  match => { "message" => "%{IP:client} %{WORD:method} %{URIPATHPARAM:request} %{NUMBER:bytes} %{NUMBER:duration}" }
}
```

https://www.elastic.co/guide/en/logstash/current/filter-plugins.html

Configuration

Output

```
elasticsearch {
    hosts                           => "server-address"
    user                            => "elastic"
    password                        => "changemee"
    cacert                          => "/path/cert/elk.crt"
    ssl                             => "true"
    ssl_certificate_verification    => "true"
    index                           => "logstash-test"
    action                          => "index"
}
```

https://www.elastic.co/guide/en/logstash/current/output-plugins.html

Configuration

Output

```
kafka {
    codec => json
    bootstrap_servers => "server_address:9092"
    topic_id => "topic-name"
    compression_type => "gzip"
}
```

https://www.elastic.co/guide/en/logstash/current/output-plugins.html

Questions

Thank you

Mr. Iyanou Eraste AKANDE

eraste.akande@gmail.com

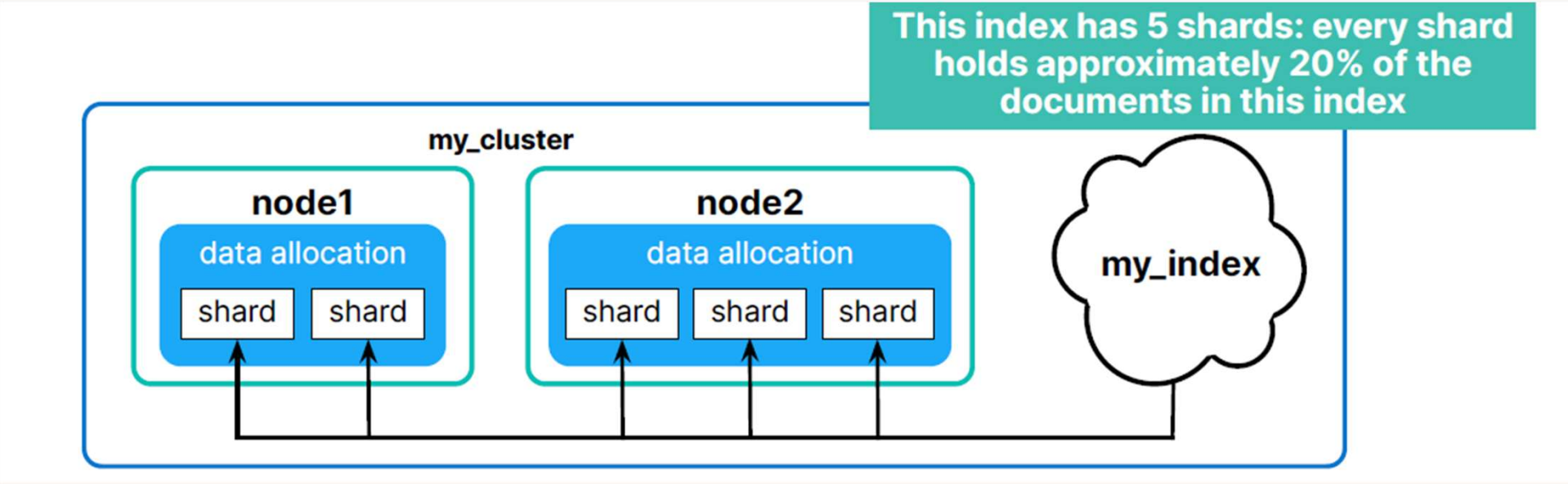
OMNISHORE

Introduction to Elasticsearch

Mr. Iyanou Eraste AKANDE
Elastic Certified Engineer
Data Engineer at Synaptique Maghreb

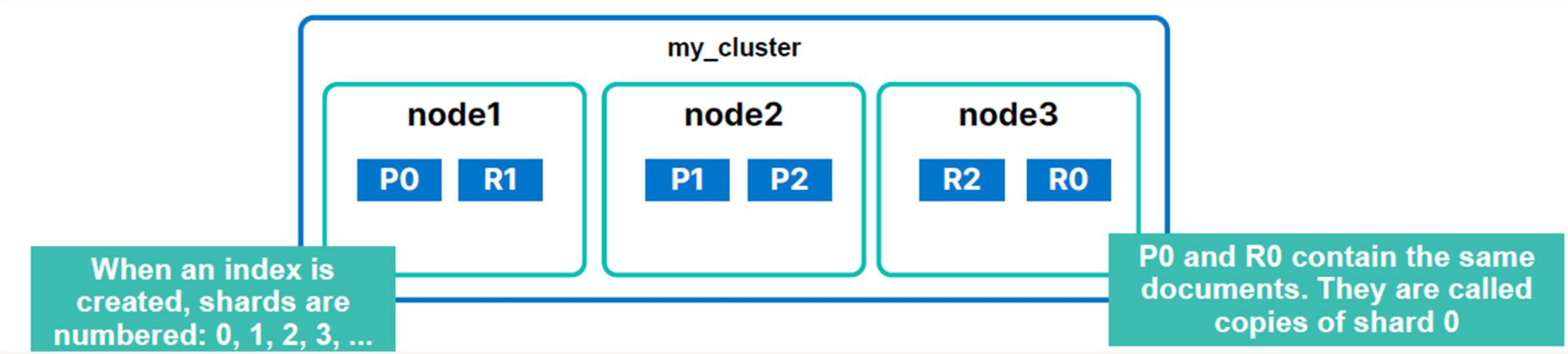
Architecture

Shards



Architecture

Primary Vs Replica

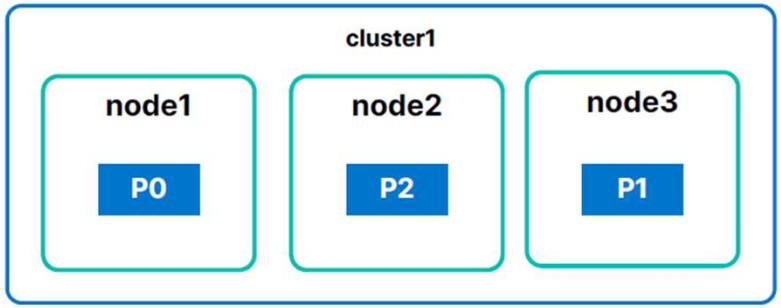


Architecture

Shards Configuration

request

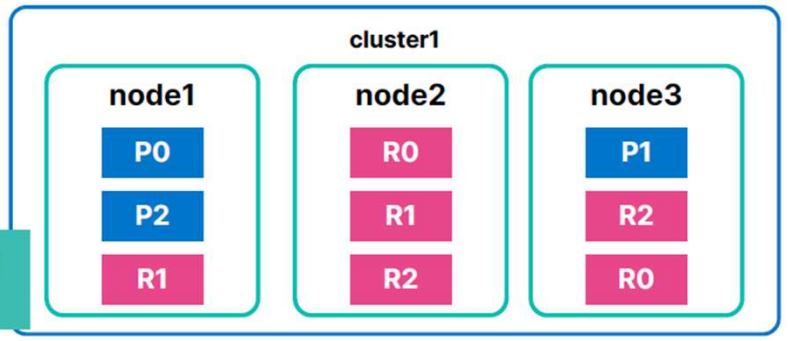
```
PUT my_new_index
{
  "settings": {
    "number_of_shards": 3
  }
}
```



request

```
PUT my_new_index/_settings
{
  "number_of_replicas": 2
}
```

3 primaries + 2 replica sets =
9 total shards



Mapping

Default Mapping

- Slow index speed
- Non optimized storage space
- Not improved searches

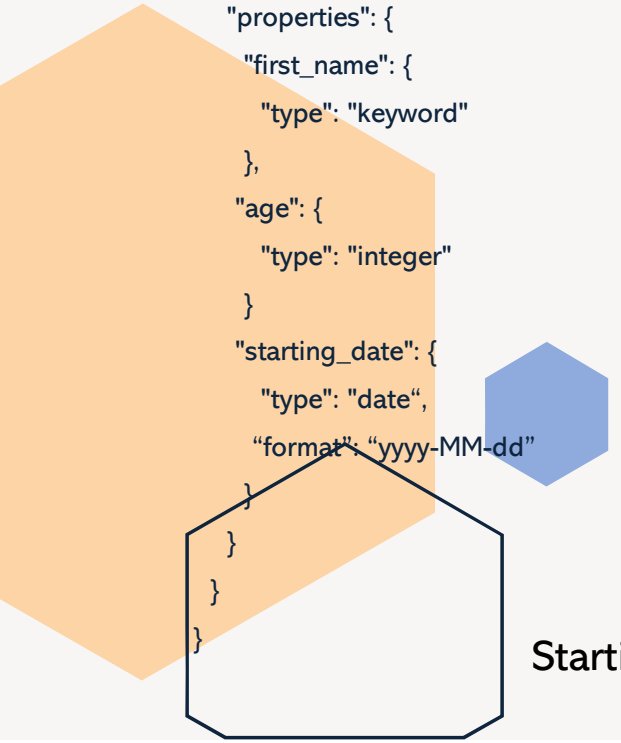
```
{
“first_name” : “Ali”,
“name” : “KOZNI”,
“gender” : “M”,
“city” : “Rabat”,
“country” : “Morocco”,
“age” : 25,
“company” : “INTEL”,
“starting_date” : “2022-04-23”
}
```

```
“first_name” : text
“first_name.keyword” : keyword
“name” : text
“name.keyword” : keyword
“gender” : text
“gender.keyword” : keyword
“city” : text
“city.keyword” : keyword
“country” : text
“country.keyword” : keyword
“age” : long
“company” : text
“company.keyword” : keyword
“starting_date” : text
“starting_date.keyword” : keyword
```

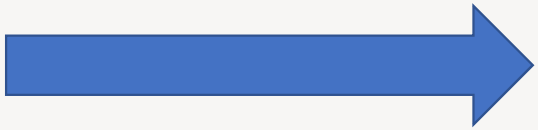
Mapping

Explicit Mapping

```
PUT employees_data
{
  "mappings": {
    "properties": {
      "first_name": {
        "type": "keyword"
      },
      "age": {
        "type": "integer"
      }
      "starting_date": {
        "type": "date“,
        “format”: “yyyy-MM-dd”
      }
    }
  }
}
```



- **Fast index speed**
- **Optimized storage space**
- **Improved searches**



“first_name” : keyword
“name” : keyword

“gender” : keyword

“city” : keyword
“country” : keyword

“age” : integer

“company” : keyword
“starting_date” : date

Mapping

Data Types

- **text**: for full-text (analyzed) strings
- **keyword**: for exact value strings and aggregations
- **date** and **date_nanos**: string formatted as dates, or numeric dates
- integer types: **byte**, **short**, **integer**, **long**
- floating-point numbers: **float**, **double**, **half_float**
- **boolean**

Mapping

Properties

- **Mappings** are Elasticsearch's data schema
- Mappings are defined **per index**
- If you do not define an explicit mapping, Elasticsearch will **dynamically** map the fields in your documents
- You cannot change the mapping of a field after the index has been created, but you can add new fields to a mapping
- Creating a custom mapping will produce savings in search and index speed, as well as memory and storage requirements

Component Template

- Reusable building blocks that can contain settings, mappings or aliases

- Reused across multiple templates

```
PUT _component_template/component_template1
{
  "template": {
    "mappings": {
      "properties": {
        "@timestamp": {
          "type": "date"
        }
      }
    }
  }
}
```

Index Template

- Multiple indices with the same settings and mappings

- Index template can contain
  - **settings**
  - **mappings**
  - **aliases**
  - **component templates**

```
PUT _index_template/template_1
{
  "index_patterns": ["spring-*"],
  "template": {
    "settings": {
      "number_of_shards": 1
    },
    "mappings": {
      "properties": {
        "host_name": {
          "type": "keyword"
        }
      }
    }
  },
  "priority": 500,
  "composed_of": ["component_template1"],
}
```

Analysers

How to analyse text fields ?

Example: A quick brown fox jumps over the lazy dog

- Character filters : Transform the stream by adding, removing, or changing characters (Ex: mapping, html_strip)
- Tokenizers : Breaking a text down into smaller chunks (Ex: whitespace)
- Token filters : A *token filter* receives the token stream and may add, remove, or change tokens (Ex: lowercase, stop, synonym)

Built in Analysers

https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-analyzers.html

Built in Tokenizers

https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-tokenizers.html

Built in Token filters

https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-tokenfilters.html

Analysers

Create a custom analyser

```
PUT /my-index-000001
{
  "settings": {
    "analysis": {
      "analyzer": {
        "my_analyzer": {
          "tokenizer": "whitespace",
          "filter": [ "stop" ]
        }
      }
    }
  }
}
```

```
PUT trim_example
{
  "settings": {
    "analysis": {
      "analyzer": {
        "keyword_trim": {
          "tokenizer": "keyword",
          "filter": [ "trim" ]
        }
      }
    }
  }
}
```

Analysers

Apply an analyser to a field

```
PUT my-index-000001
{
  "mappings": {
    "properties": {
      "title": {
        "type": "text",
        "analyzer": "whitespace"
      }
    }
  }
}
```

Query DSL

Search

```
GET index_name/_search
{
    "query": {
       "match": {
          "title": "lifestyle"
        }
    }
}
```

```
GET index_name/_search
{
    "query": {
       "term": {
          "name": "Alain"
        }
    }
}
```

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

Query DSL

Aggregations

```
GET index_name/_search
{
  "aggs": {
    "my-agg-name": {
      "terms": {
        "field": "my-field"
      }
    }
  }
}
```

```
GET index_name/_search
{
  "aggs": {
    "my-second-agg-name": {
      "avg": {
        "field": "my-other-field"
      }
    }
  }
}
```

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

Reindex API

Local

```
POST _reindex
{
    "source": {
        "index": " source_index_name",
        "query": {...}
    },
    "dest": {
        "index": " destination_index_name"
    }
}
```

Reindex API

Remote

```
POST _reindex
{
    "source": {
        "remote": {
            "host": "http://otherhost:9200",
            "username": "user",
            "password": "pass"
        },
        "index": "remote_index",
    },
    "dest": {
        "index": "local_index"
    }
}
```

Update By Query

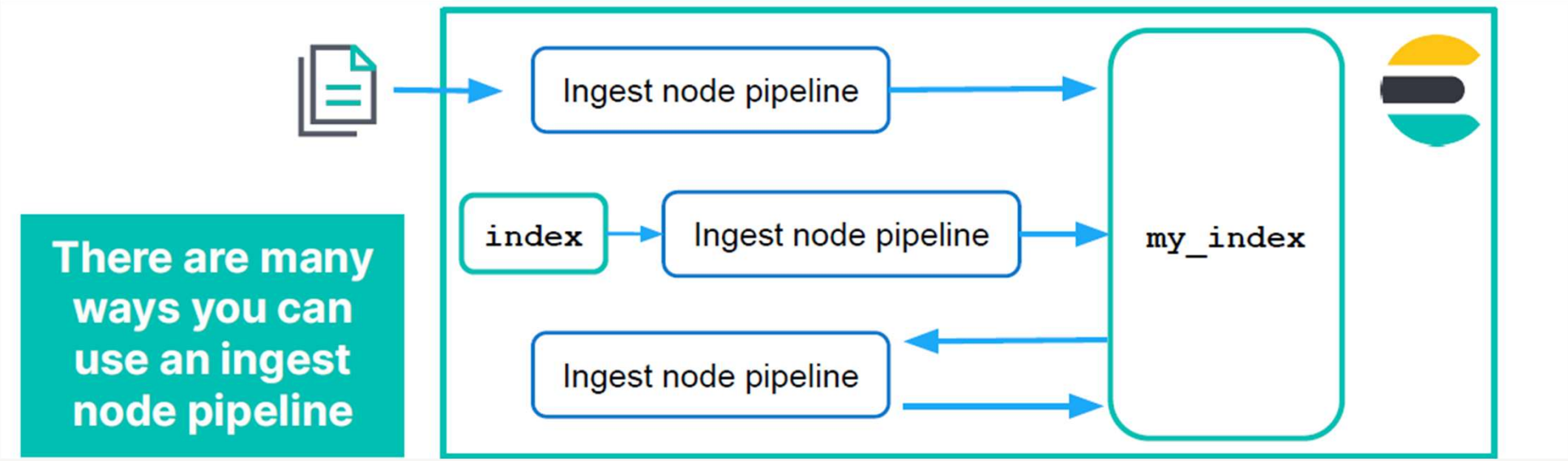
```
POST index_name/_update_by_query
{
    "query": {
        …..
    },
    "script": {
          "source": "ctx._source. name = Ali"
    }
}
```

Delete By Query

```
POST index_name/_delete_by_query
{
    "query": {
        "match": {
            “name": “Malton Jack"
        }
    }
}
```

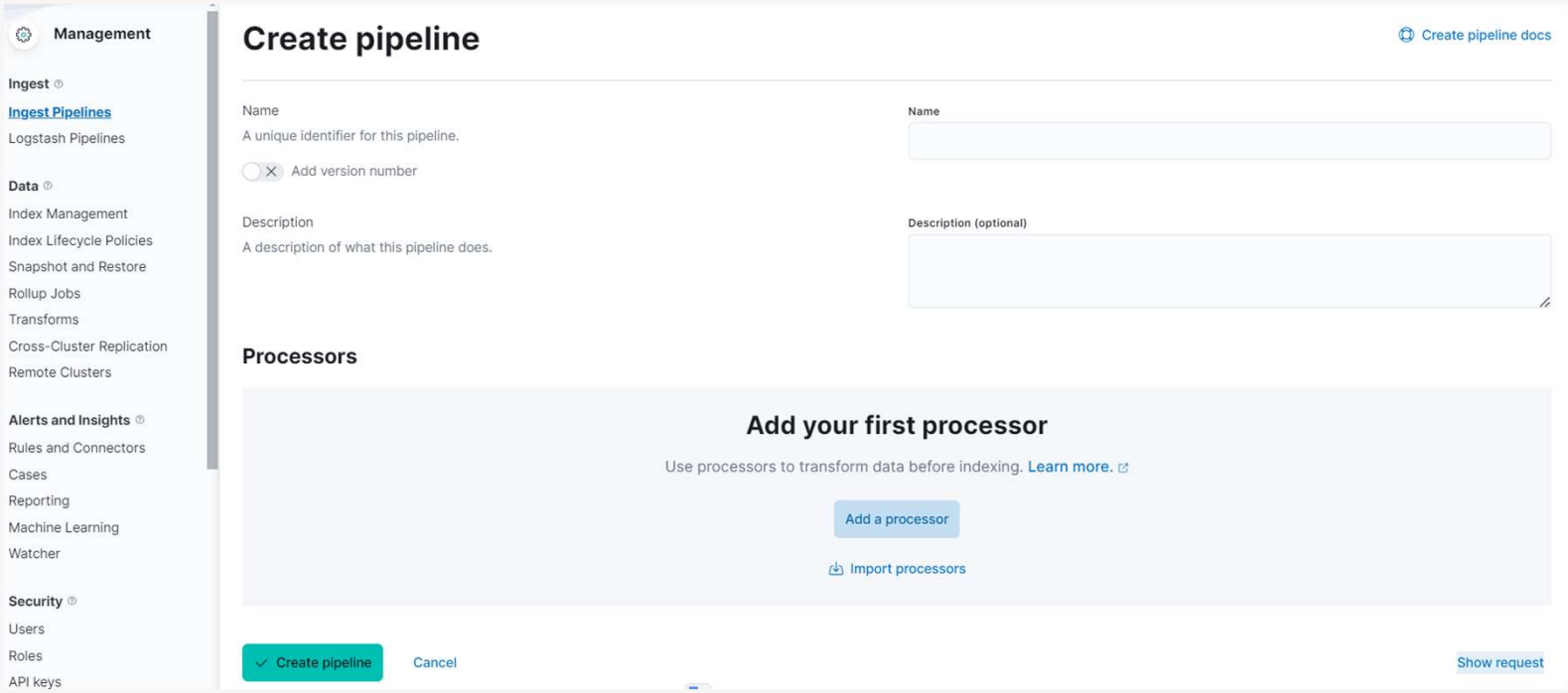
Ingest Node Pipelines

How it works ?



Ingest Node Pipelines

Pipeline Creation



Ingest Node Pipelines

Processors

Manipulate Fields

- set
- remove
- rename
- dot_expander
- etc

Manipulate Values

- split/join
- grok
- dissect
- gsub
- etc

Special Operations

- csv/json
- geoip
- user_agent
- script
- pipeline
- etc

Ingest Node Pipelines

Usage

- **POST** my_index/_update_by_query? **pipeline**=my_pipeline

- **PUT** my_index

```
{
    "settings": {
        "default_pipeline": "my_pipeline"
    }
}
```

- **POST** _reindex

```
{
    "source": {
        "index": "my_index"
    },
    "dest": {
         "index": "new_index",
         "pipeline": "my_pipeline"
}
```

Enrich Data

Explanation

stock		
id	product	price
1	rice	40
2	bag	30
3	hat	15
4	dog	200
5	desk	300
6	chair	150
7	table	250
8	computer	1000
9	phone	750
10	oil	60

provenance	
id	origin
1	Benin
2	Togo
3	Morocco
4	Congo
5	Mali
6	Niger
7	Soudan
8	Angola
9	Namibie
10	Rwanda

Enrich Data

Enrich Policy

```
PUT _enrich/policy/ product_mapper
{
    "match": {
        "indices": "provenance",
        "match_field": "id",
        "enrich_fields": ["country"]
    }
}
```

Enrich Data

Create Enrich Index

```
POST _enrich/policy/product_mapper/_execute
```

Enrich Data

Ingest Pipeline

```
PUT /_ingest/pipeline/product_mapper_pipeline
{
    "processors" : [
        {
            "enrich" : {
                "policy_name": "product_mapper",
                "field" : "id",
                "target_field": "country"
            }
        }
    ]
}
```

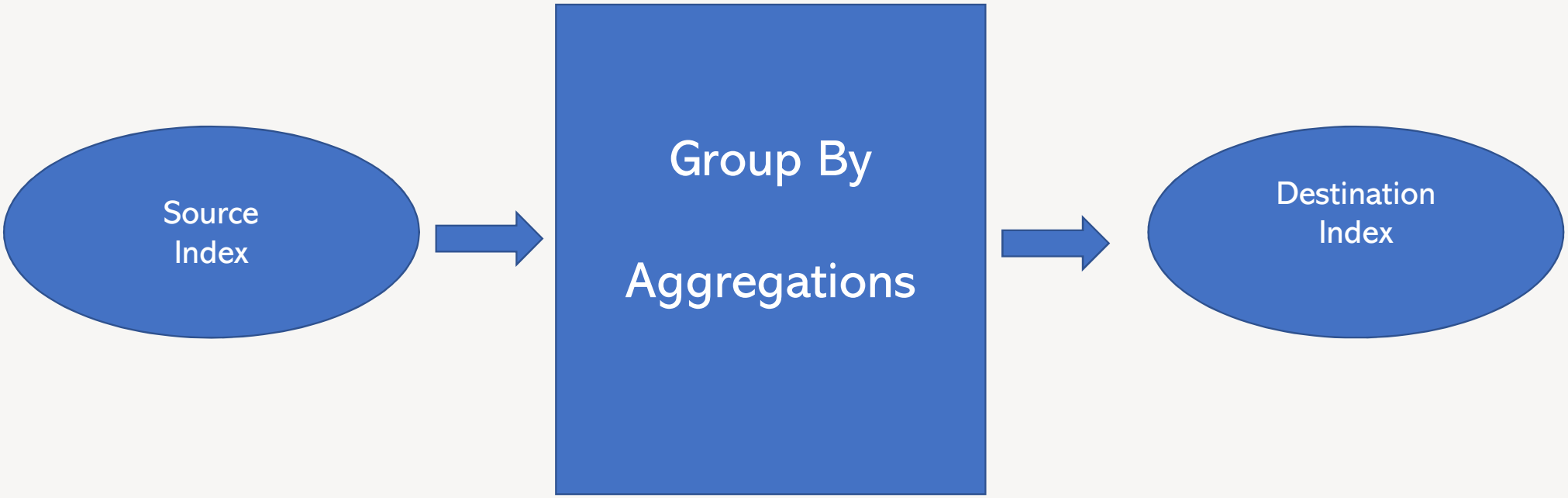
Enrich Data

Use the pipeline

```
POST stock/_update_by_query?pipeline=product_mapper_pipeline
```

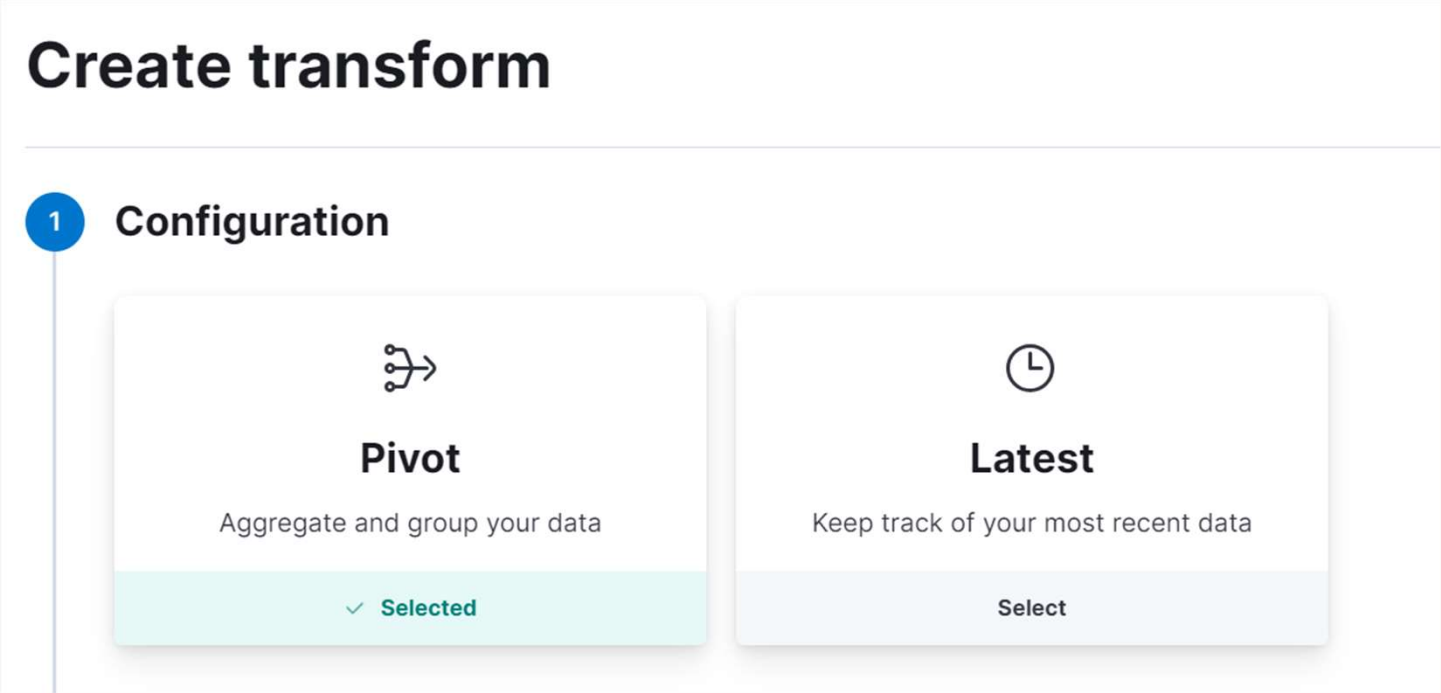
Transforms

Explanation



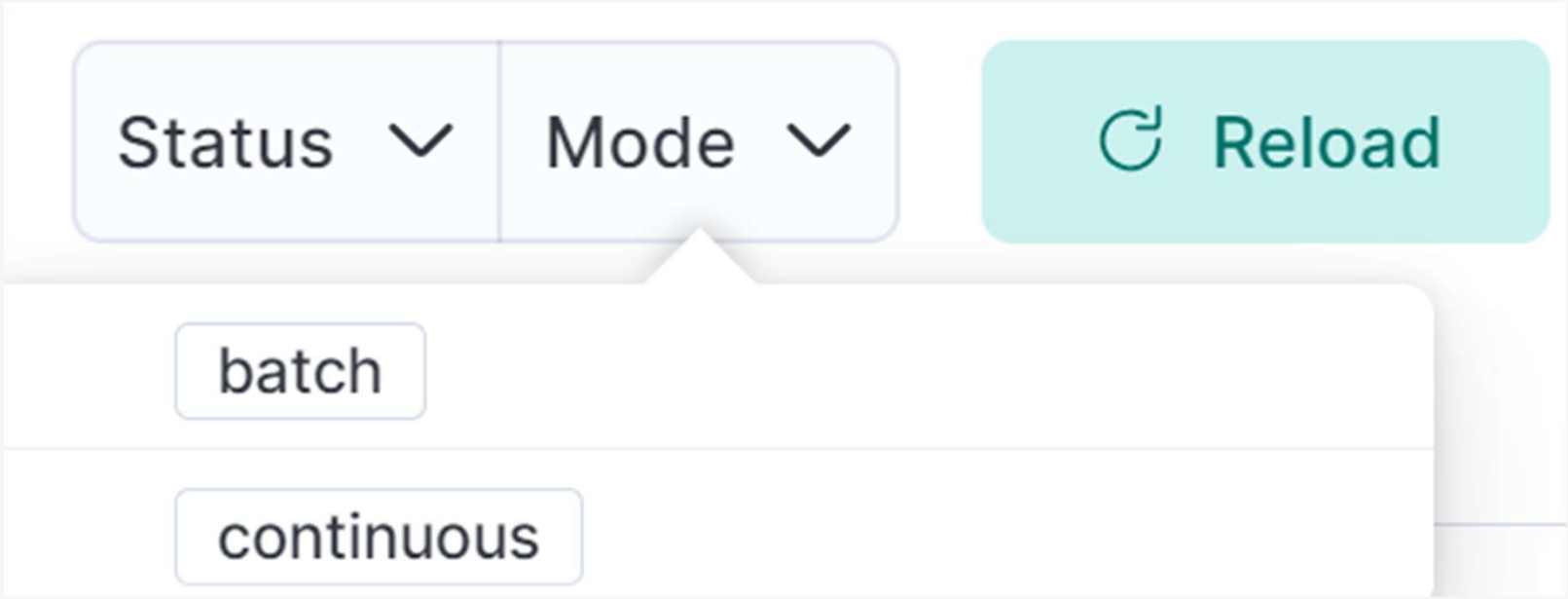
Transforms

Types



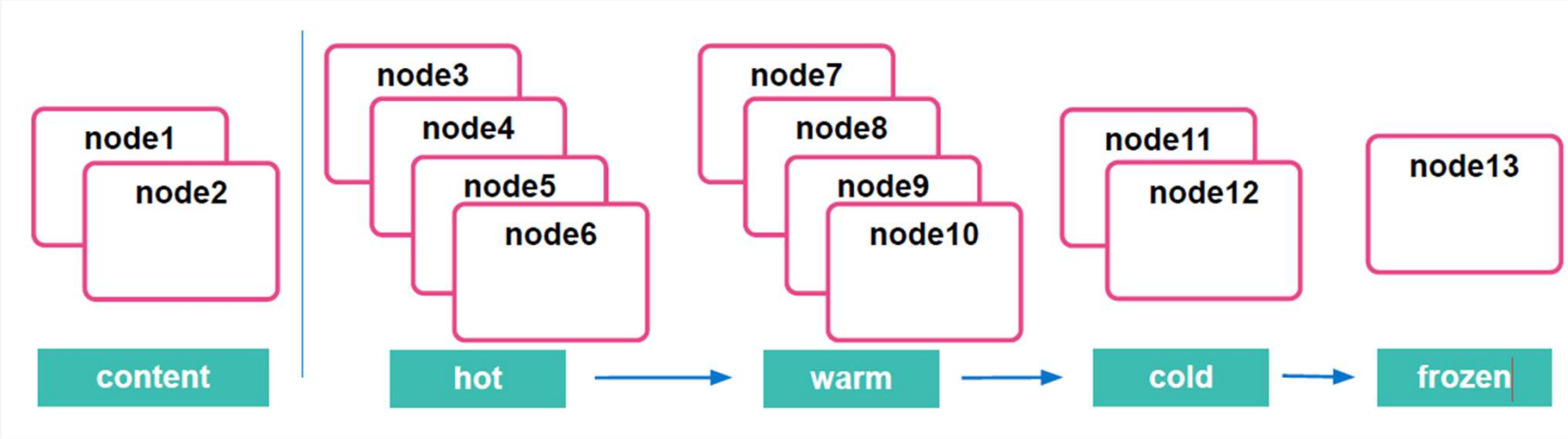
Transforms

Mode



Index Lifecycle Management

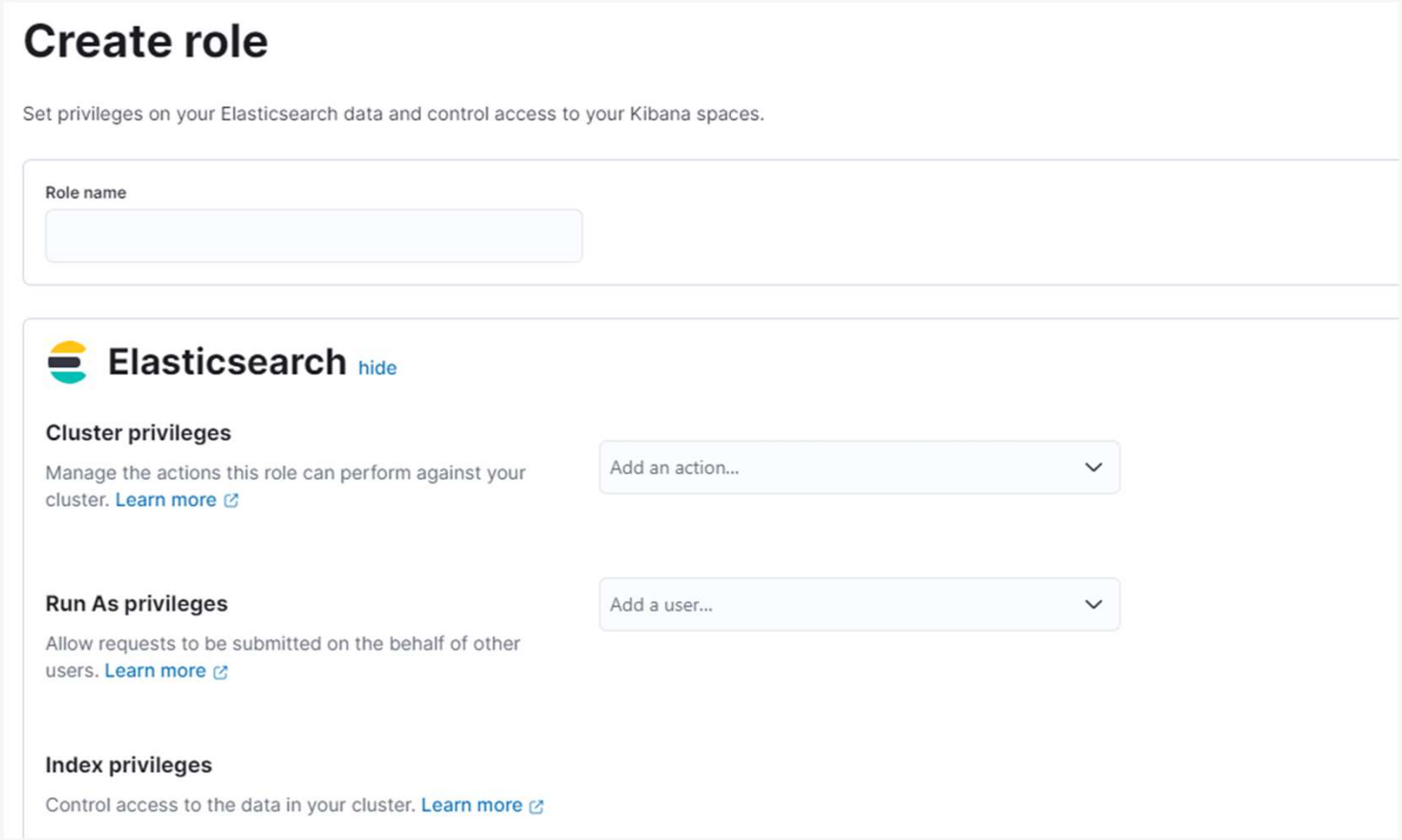
Data Tiers



node.roles: ["data_warm", "data_content"]

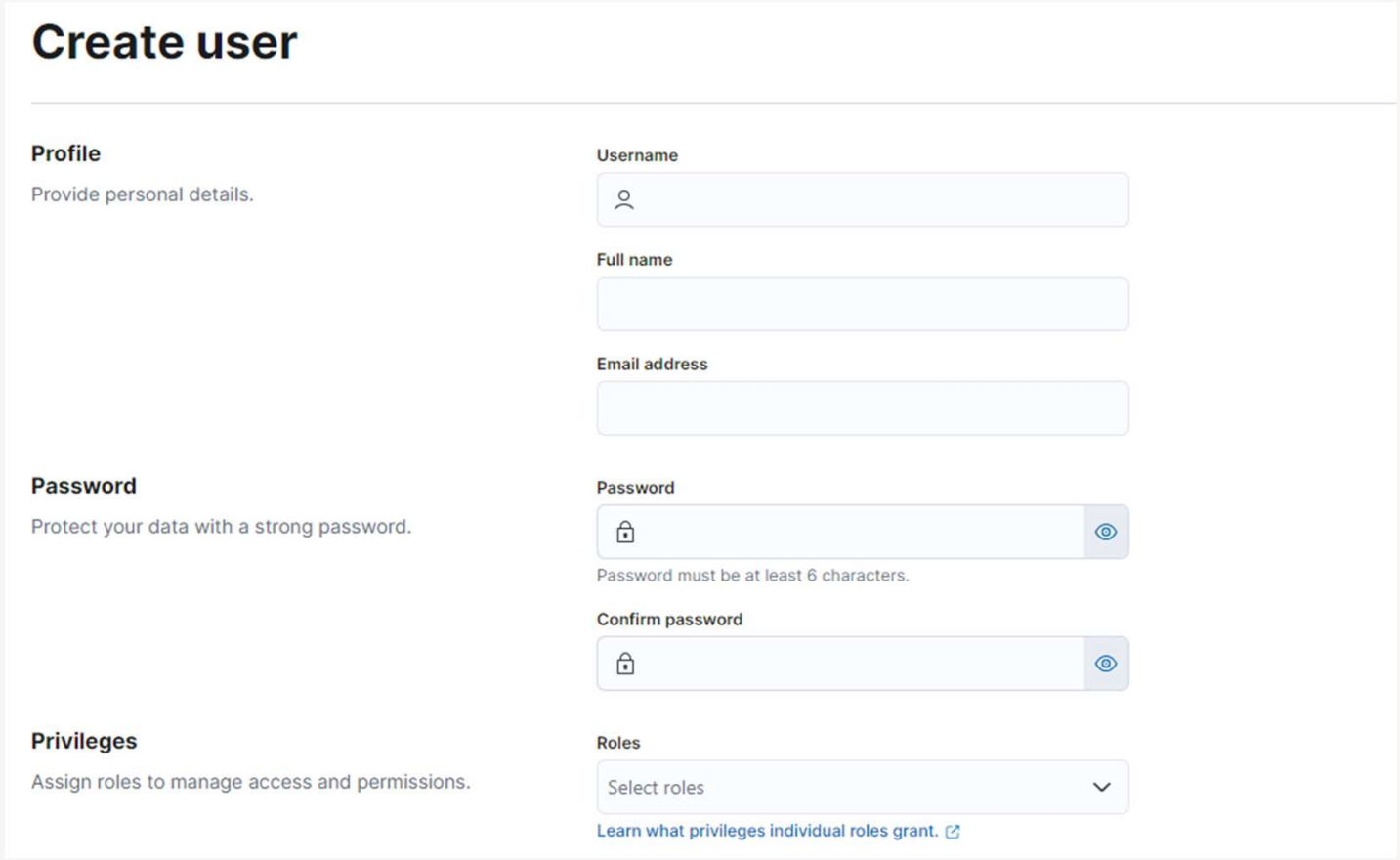
Security

Role



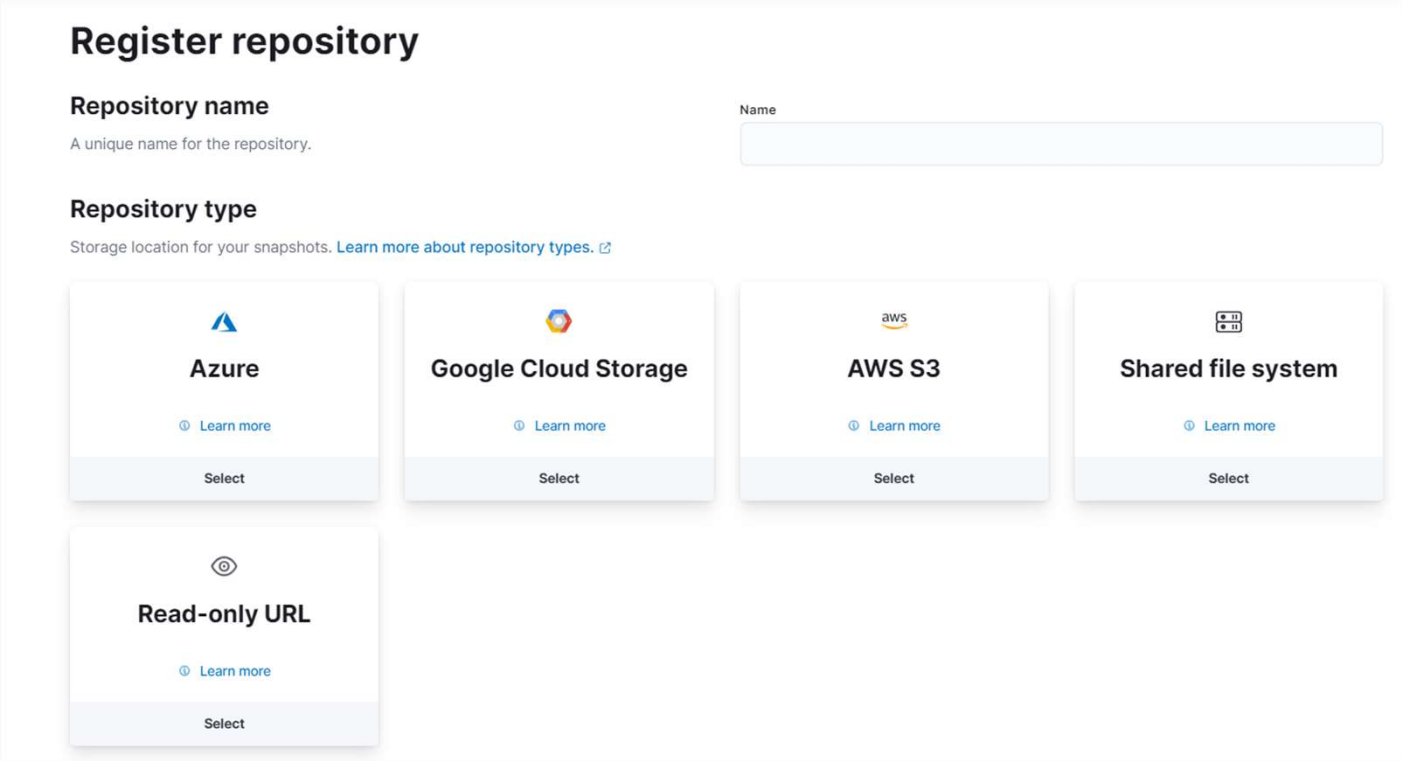
Security

User



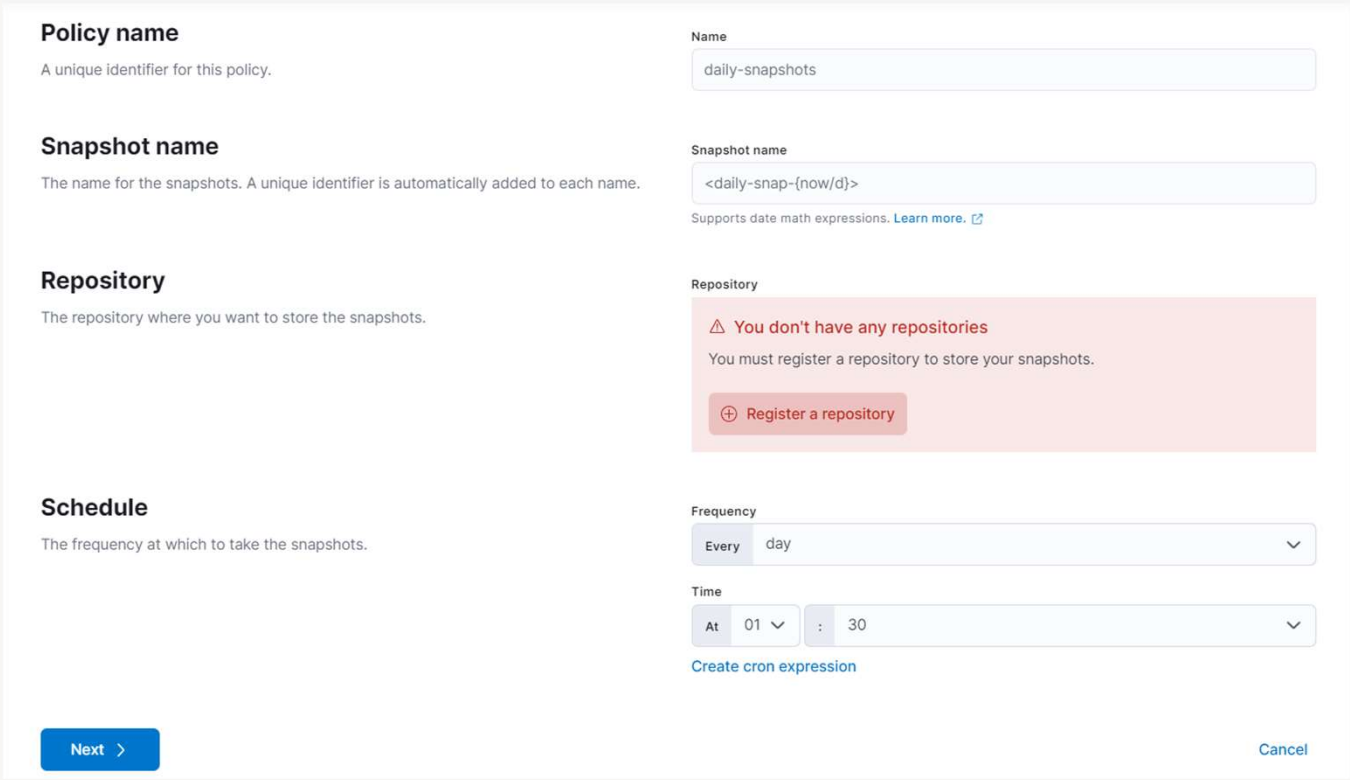
Snapshots & Restore

Repository



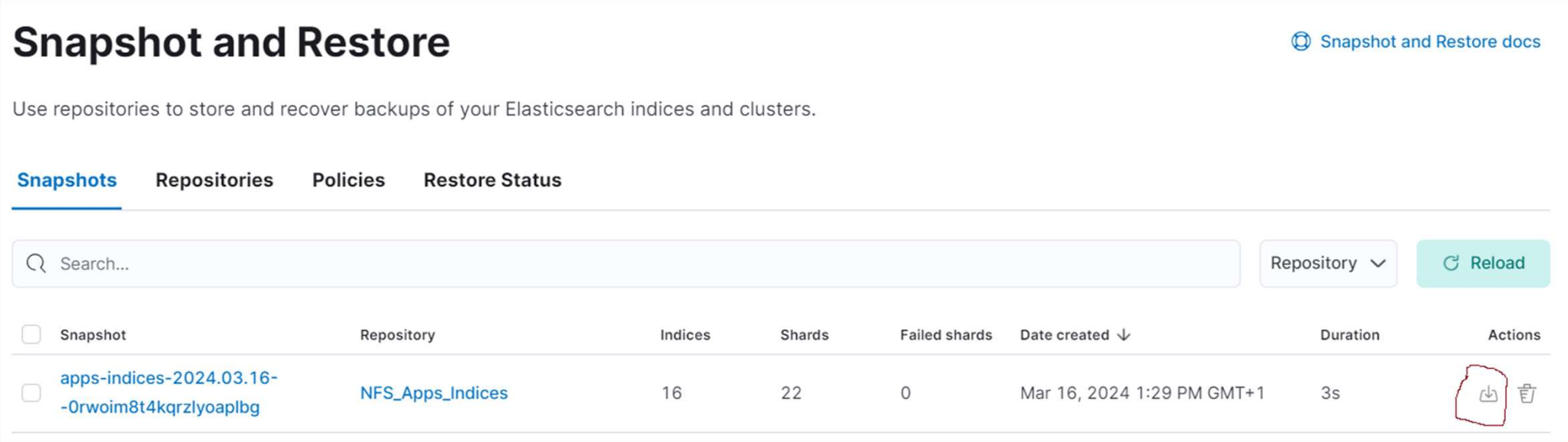
Snapshots & Restore

Policy



Snapshots & Restore

Restore

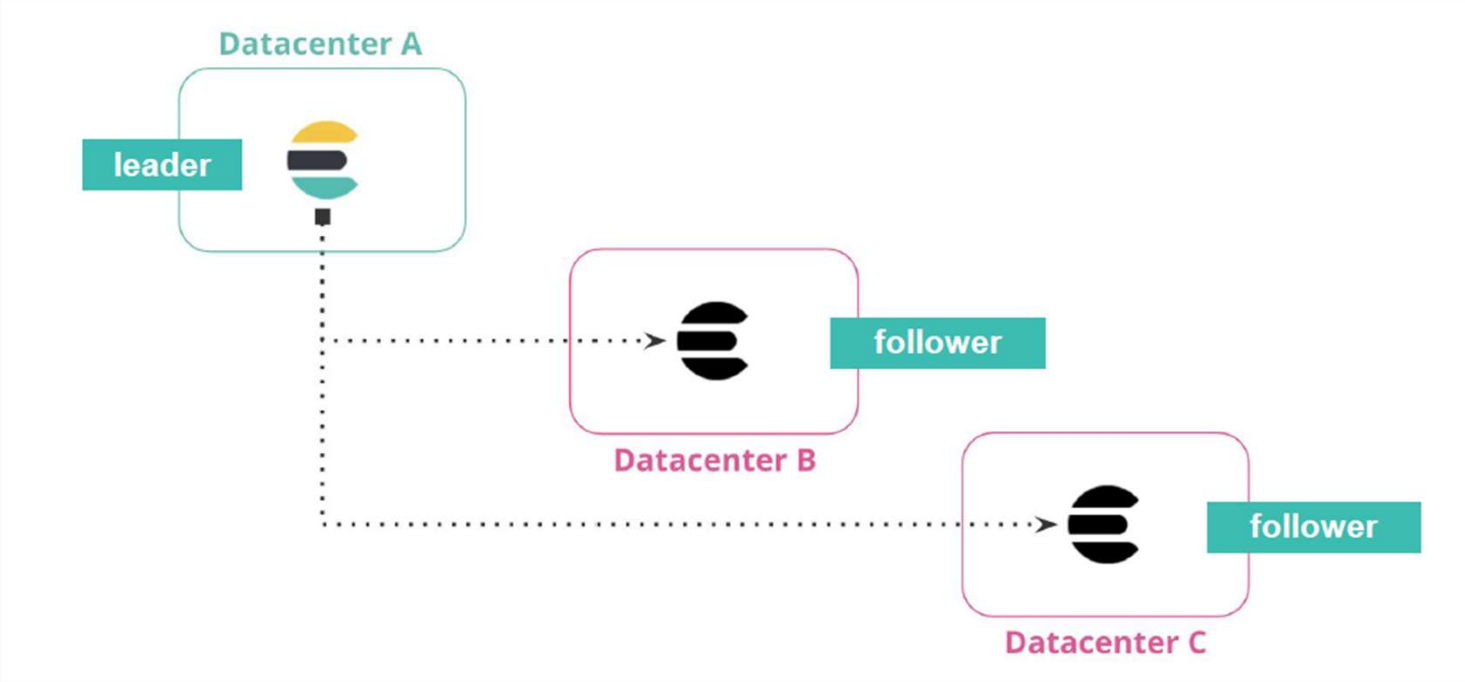


Troubleshooting

Command	Response
GET _tasks	the running tasks on the cluster
GET _cluster/pending_tasks	any cluster-level changes that have not yet executed
GET _nodes/hot_threads	threads using high CPU volume and executing for a long time
GET _cat/shards	the statistics of shards
GET _cat/health	the health of the cluster
GET _cluster/allocation/explain	explain the shards allocation

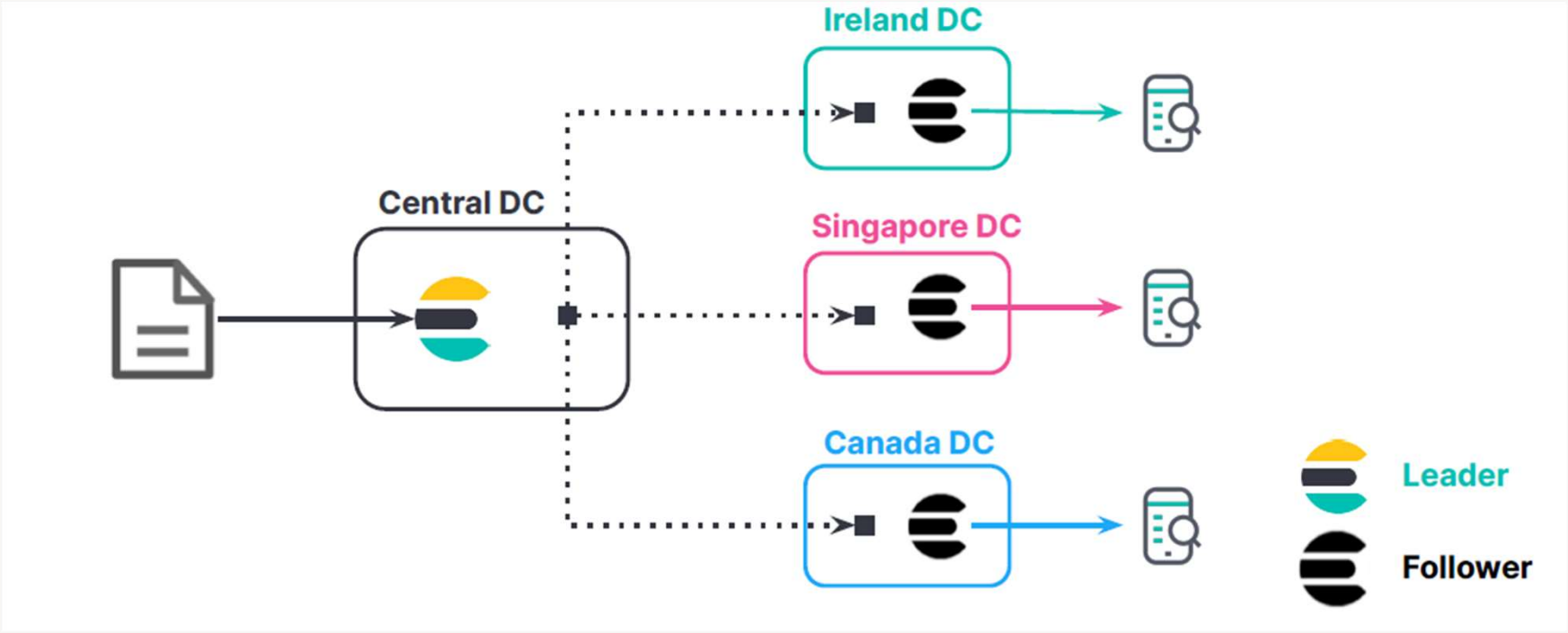
Cross Cluster Replication

Disaster recovery and high availability



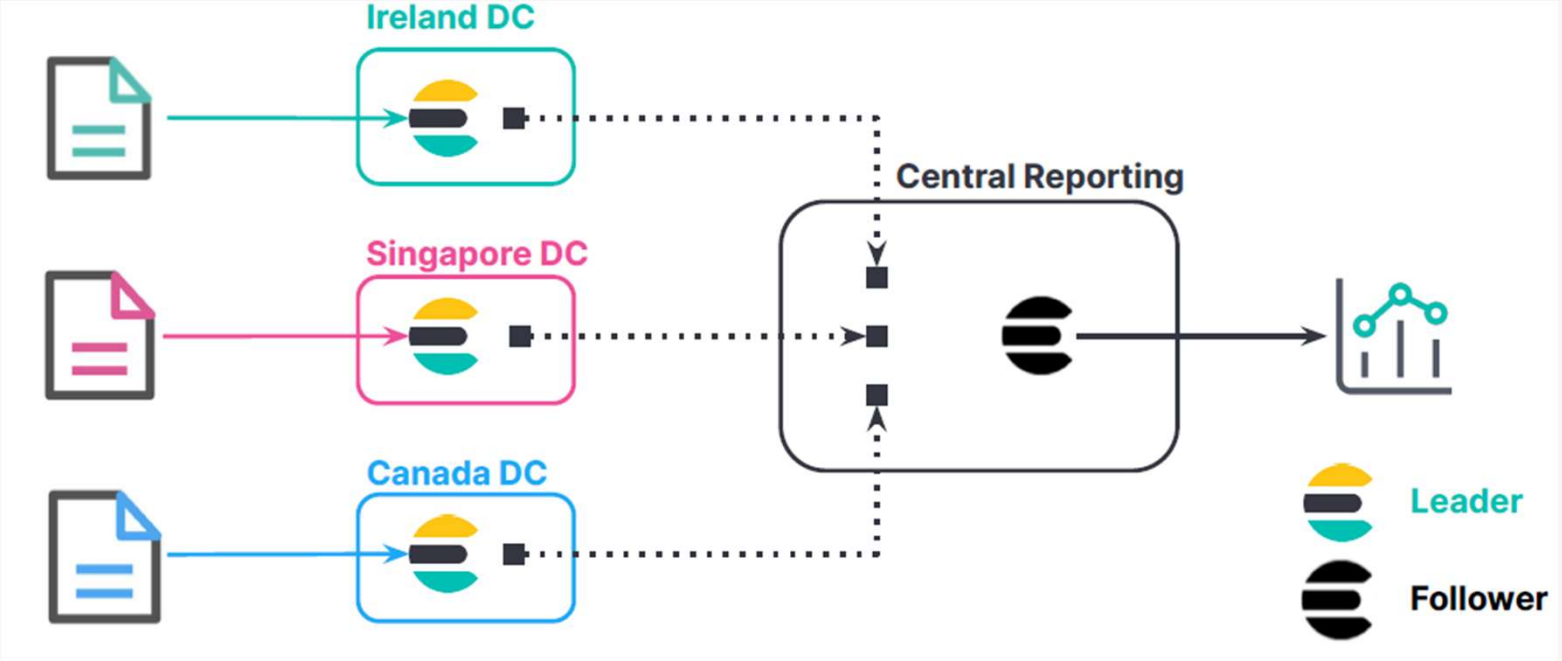
Cross Cluster Replication

Data locality



Cross Cluster Replication

Centralized reporting



Questions

Thank you

Mr. Iyanou Eraste AKANDE

eraste.akande@gmail.com