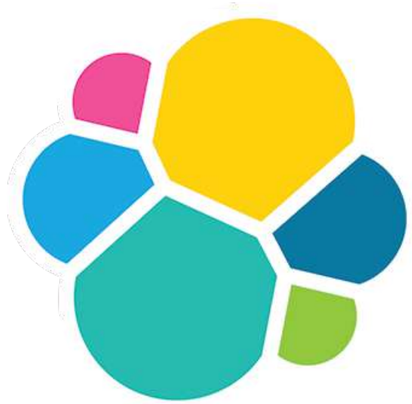
Enriching Data

Elastic Stack

Philipp Krenn **@xeraa**

Ceci n'est pas David



elastic

Developer 🥑

"Enriching"

Not 💎 👑 💰

When?

Ingest vs **Runtime**

Tradeoffs

Correctness

Where?

Edge vs **Central** vs **In-Cluster**

How?

Logstash vs Beats vs Elastic Agent vs OpenTelemetry Collector vs Elasticsearch Ingest Pipeline vs Elasticsearch Runtime Field

Demo

Filter your data using KQL syntax

Last 7 days 15 m Refresh

Sample Logs Data

This dashboard contains sample data for you to play with. You can view it, search it, and interact with the visualizations. For more information about Kibana, check our docs.

Visits

1.81K

Unique Visitors

866

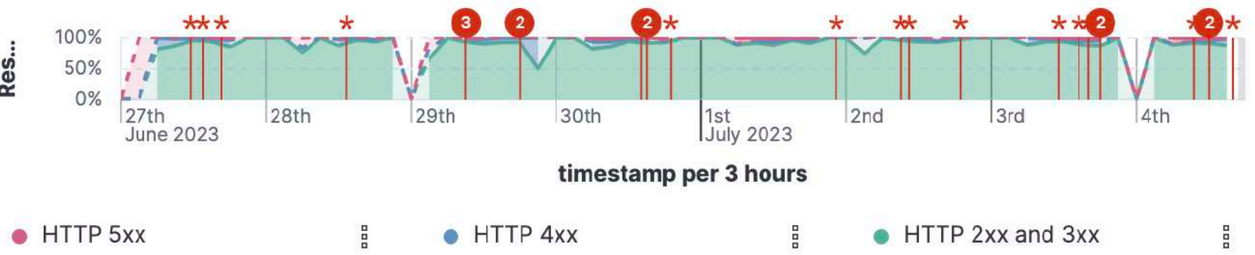
HTTP 4xx

5.09%

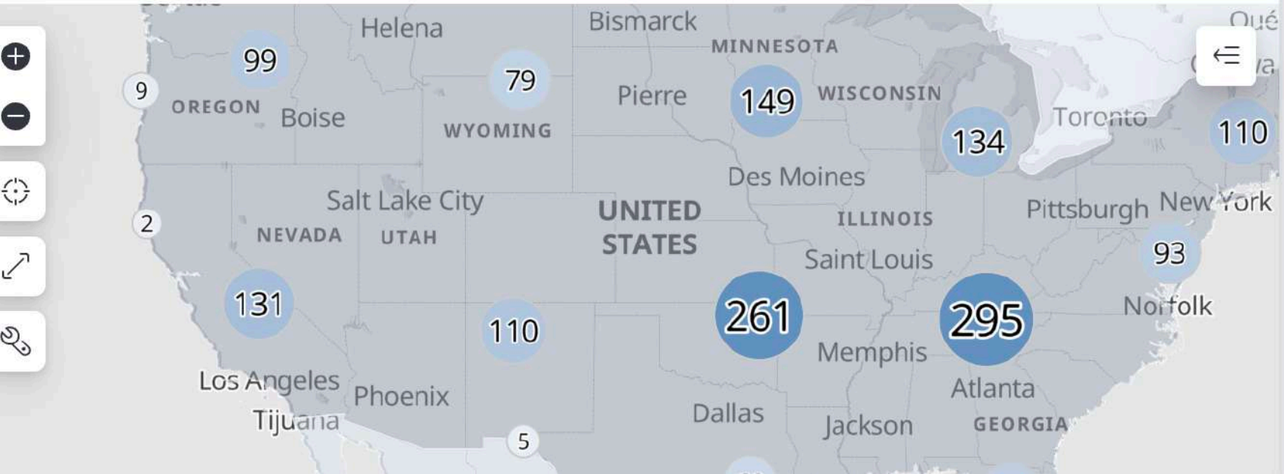
HTTP 5xx

2.77%

ogs] Response Codes Over Time + Annotations



ogs] Total Requests and Bytes



[Logs] Machine OS and Destination Sankey Chart

Elasticsearch Ingest Pipeline

```
POST _ingest/pipeline/_simulate
{
  "docs": [
    {
      "_source": {
        "clientip": "164.85.94.243"
      }
    }
  ],
  "pipeline": {
    "processors": [{
      "enrich": {
        "policy_name": "vip-policy",
        "field": "clientip",
        "target_field": "enriched"
      }
    }]
  }
}
```

Elasticsearch Runtime Field

```
GET kibana_sample_data_logs/_search
{
  "query": {
    "match": {
      "clientip": "164.85.94.243"
    }
  },
  "runtime_mappings": {
    "enriched": {
        "type": "lookup",
        "target_index": "vip",
        "input_field": "clientip",
        "target_field": "ip",
        "fetch_fields": ["name", "vip"]
    }
  },
  "fields": [
    "clientip",
    "enriched"
  ]
}
```

Logstash

```
input {
  elasticsearch {
    hosts => ["${ELASTICSEARCH_URL}"]
    user => "elastic"
    password => "${ELASTIC_PASSWORD}"
    index => "kibana_sample_data_logs"
    docinfo => true
    ecs_compatibility => "disabled"
  }
}
```

```
filter {
  elasticsearch {
    hosts => ["${ELASTICSEARCH_URL}"]
    user => "elastic"
    password => "${ELASTIC_PASSWORD}"
    index => "vip"
    query => "ip:%{[clientip]}"
    sort => "ip:desc"
    fields => {
      "[name]" => "[name]"
      "[vip]" => "[vip]"
    }
  }
  mutate {
    remove_field => ["@version", "@timestamp"]
  }
}
```

```
output {
  if [name] {
    # Write all modified documents to Elasticsearch
    elasticsearch {
      manage_template => false
      hosts => ["${ELASTICSEARCH_URL}"]
      user => "elastic"
      password => "${ELASTIC_PASSWORD}"
      index => "%{[@metadata][_index]}"
      document_id => "%{[@metadata][_id]}"
    }
  }
}
```

Naming

Elastic Common Schema (ECS)

https://www.elastic.co/what-is/ecs

```
src:10.42.42.42 OR client_ip:10.42.42.42 OR apache2.access.remote_ip:10.42.42.42 OR
context.user.ip:10.42.42.42 OR src_ip:10.42.42.42
```

```
source.ip:10.42.42.42
```

Announcing the Elastic Common Schema (ECS) and OpenTelemetry Semantic Convention Convergence

By **Reiley Yang** | Monday, April 17, 2023

Today, we're very excited to make a joint announcement with Elastic about the future of Elastic Common Schema (ECS) and the OpenTelemetry Semantic Conventions.

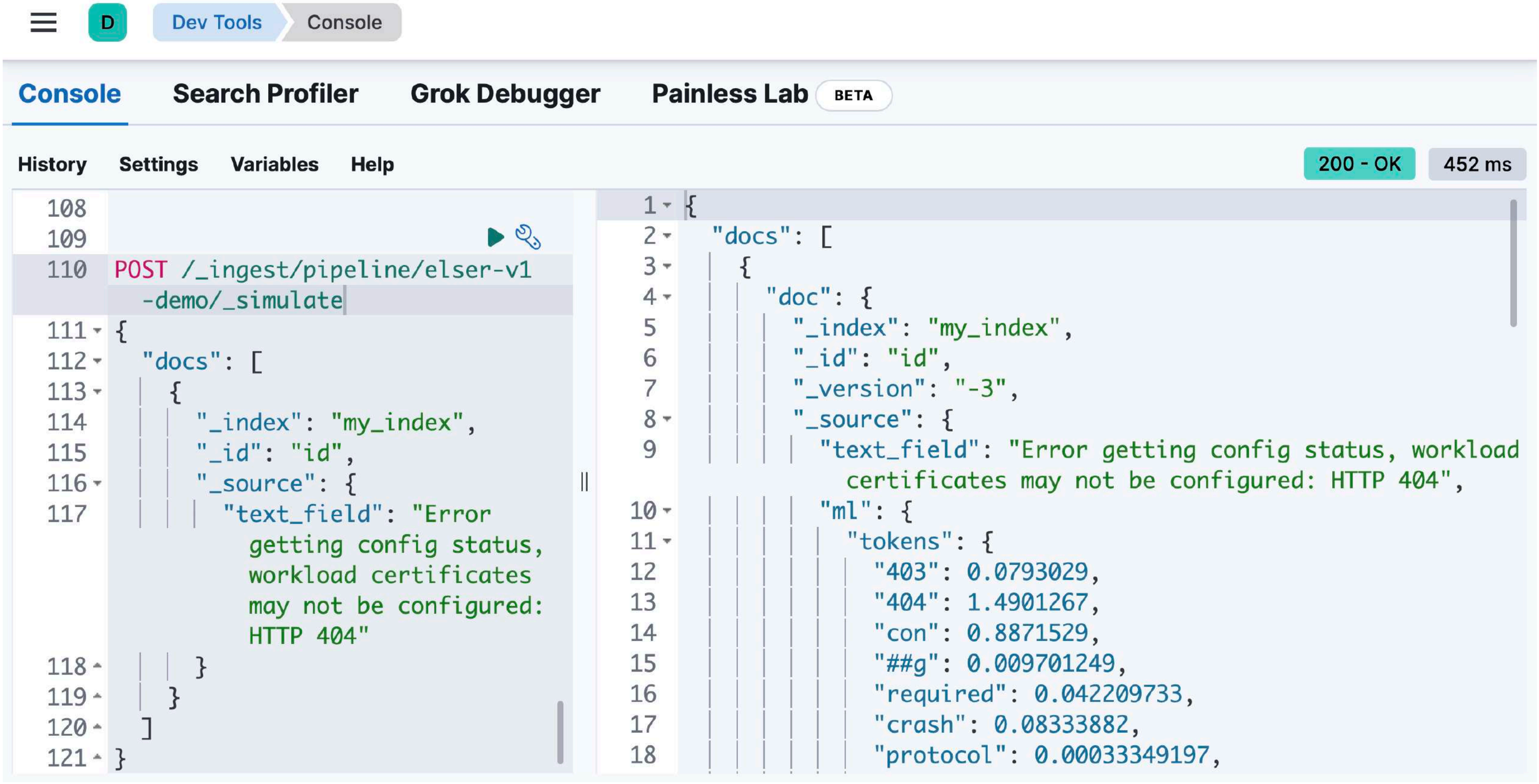
The goal is to achieve convergence of ECS and OTel Semantic Conventions into a single open schema that is maintained by OpenTelemetry, so that OpenTelemetry Semantic Conventions truly is a successor of the Elastic Common Schema. OpenTelemetry shares the same interest of improving the convergence of observability and security in this space. We believe this schema merge brings huge value to the open source community because:

- ECS has years of proven success in the logs, metrics, traces and security events schema, providing great coverage of the common problem domains.
- ECS provides schema for security domain fields, which is an important aspect of telemetry

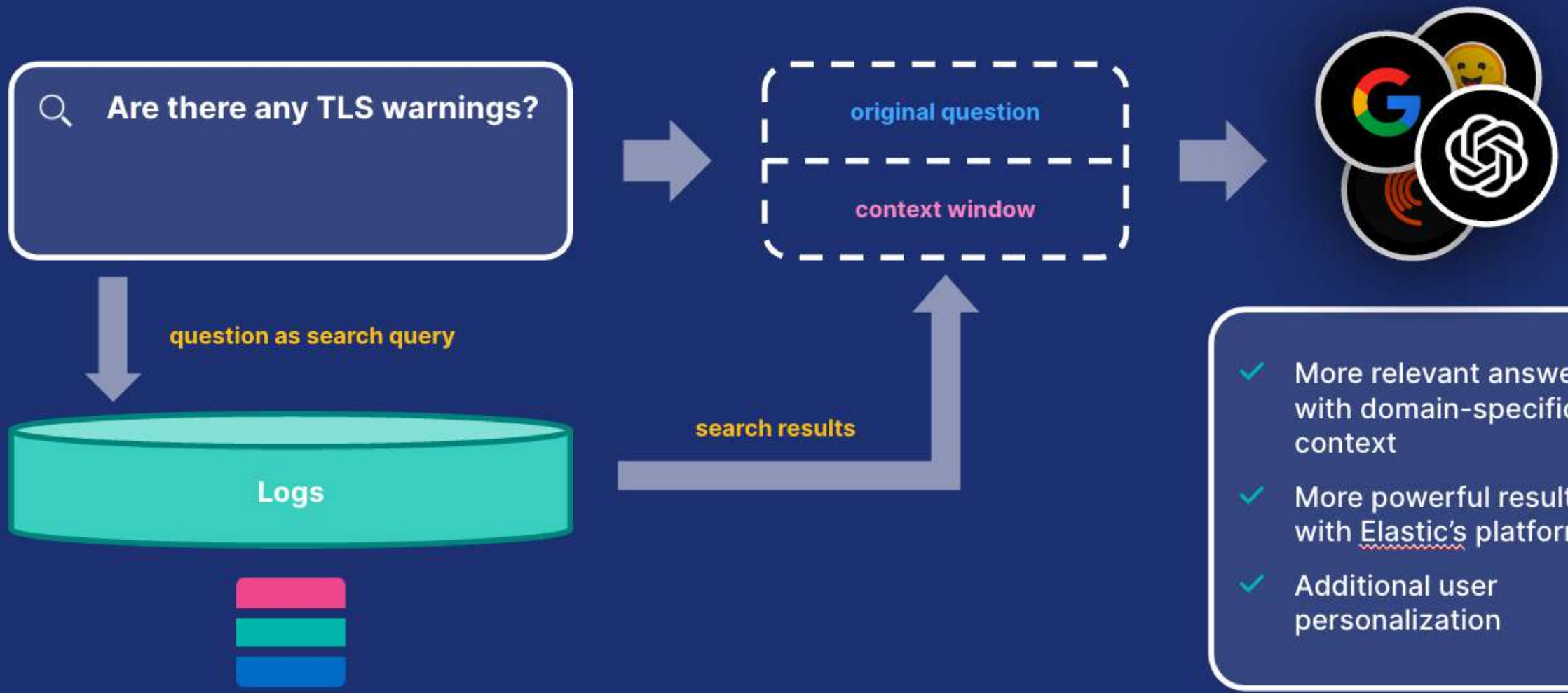
AI / ML

Elastic Learned Sparse Encoder (ELSER)

Term Expansion


D
Dev Tools
Console
Console
Search Profiler
Grok Debugger
Painless Lab
BETA
History
Settings
Variables
Help
200 - OK
452 ms
POST /_ingest/pipeline/elser-v1-demo/_simulate
{
"docs": [
{
"_index": "my_index",
"_id": "id",
"_source": {
"text_field": "Error getting config status, workload certificates may not be configured: HTTP 404"
}
}
]
}
{
"docs": [
{
"doc": {
"_index": "my_index",
"_id": "id",
"_version": "-3",
"_source": {
"text_field": "Error getting config status, workload certificates may not be configured: HTTP 404",
"ml": {
"tokens": {
"403": 0.0793029,
"404": 1.4901267,
"con": 0.8871529,
"##g": 0.009701249,
"required": 0.042209733,
"crash": 0.08333882,
"protocol": 0.00033349197,

The full picture



- More relevant answers with domain-specific context
- More powerful results with Elastic's platform
- Additional user personalization

elastic

PS: "De-rich" Personal Data

Named Entity Recognition (NER) & Redact Ingest Pipeline

Adresses with BANO

La Base Adresses Nationale Ouverte: https://bano.openstreetmap.fr

```
{
  "name": "Joe Smith",
  "address": {
    "number": "23",
    "street_name": "r verdiere",
    "city": "rochelle",
    "country": "France"
  }
}
```



```
{
  "name": "Joe Smith",
  "location": {
    "lat": 46.15735,
    "lon": -1.1551
  }
}
```

1. Load BANO data into Elasticsearch with Logstash

2. Enrich incoming data in Logstash by querying Elasticsearch

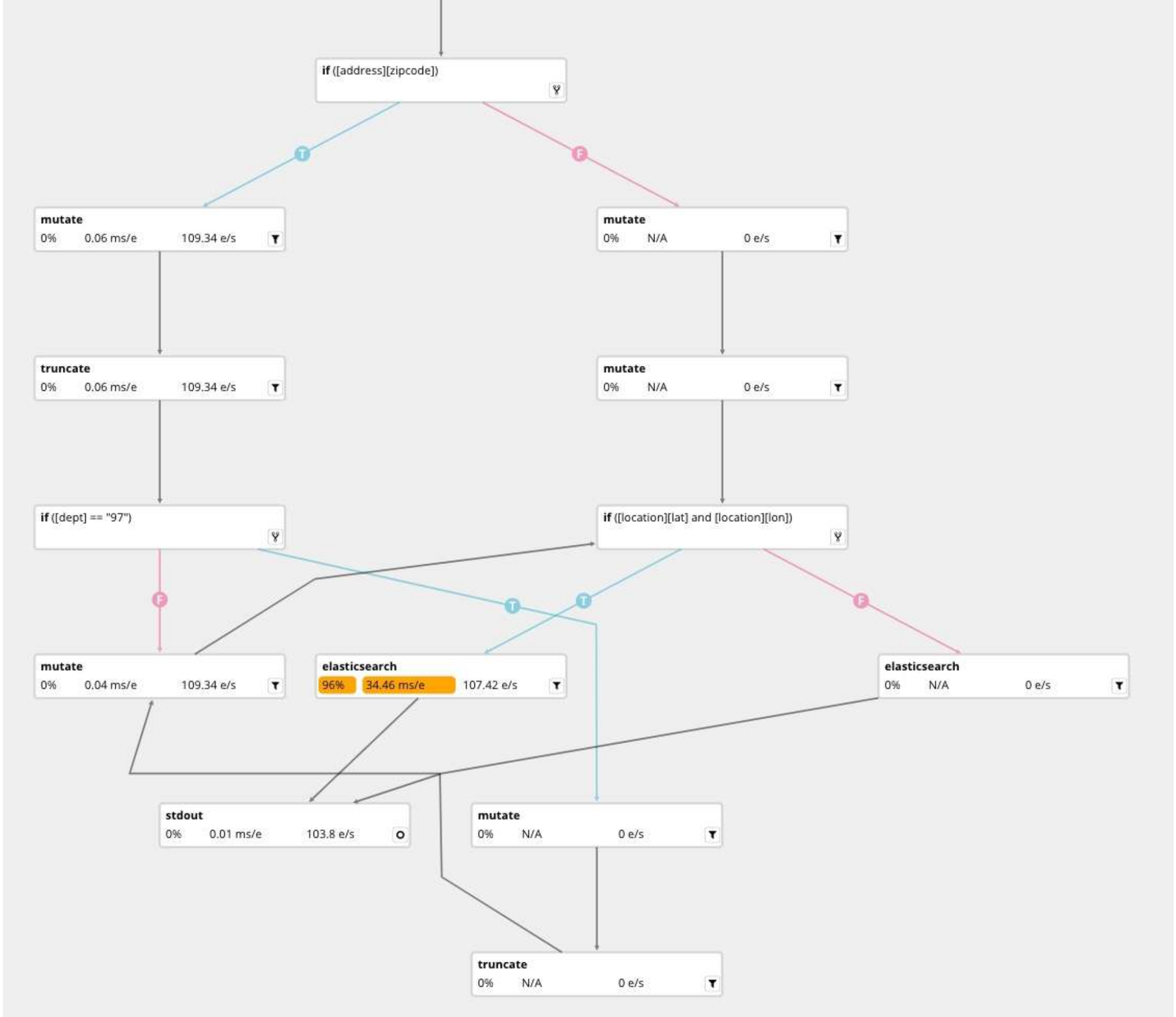
```
filter {
  if [address][zipcode] {
    mutate { add_field => { "dept" => "%{[address][zipcode]}" } }
    truncate {
      fields => ["dept"]
      length_bytes => 2
    }
    if [dept] == "97" {
      mutate { replace => { "dept" => "%{[address][zipcode]}" } }
      truncate {
        fields => ["dept"]
        length_bytes => 3
      }
    }
    mutate { add_field => { "index_suffix" => "-%{dept}" } }
```

```
} else {
  mutate { add_field => { "dept" => "" } }
  mutate { add_field => { "index_suffix" => "" } }
}
if [location][lat] and [location][lon] {
  elasticsearch {
    query_template => "search-by-geo.json"
    index => ".bano"
    fields => {
      "location" => "[location]"
      "address" => "[address]"
    }
    remove_field => ["headers", "host", "@version",
                     "@timestamp", "index_suffix", "dept"]
  }
```

```
  } else {
    elasticsearch {
      query_template => "search-by-name.json"
      index => ".bano"
      fields => {
        "location" => "[location]"
        "address" => "[address]"
      }
      remove_field => ["headers", "host", "@version",
                       "@timestamp", "index_suffix", "dept"]
    }
  }
}
```

Alternative

```
jdbc_static {
  local_db_objects => [ {
    name => "vip"
    index_columns => ["ip"]
    columns => [
      ["name", "VARCHAR(255)"],
      ["vip", "BOOLEAN"],
      ["ip", "VARCHAR(64)"]
    ]
  } ]
}
```


if ([address][zipcode])
T
F
mutate
0% 0.06 ms/e 109.34 e/s
mutate
0% N/A 0 e/s
truncate
0% 0.06 ms/e 109.34 e/s
mutate
0% N/A 0 e/s
if ([dept] == "97")
if ([location][lat] and [location][lon])
F
T
T
F
mutate
0% 0.04 ms/e 109.34 e/s
elasticsearch
96% 34.46 ms/e 107.42 e/s
elasticsearch
0% N/A 0 e/s
stdout
0% 0.01 ms/e 103.8 e/s
mutate
0% N/A 0 e/s
truncate
0% N/A 0 e/s

https://www.elastic.co/blog/enriching-your-postal-addresses-with-the-elastic-stack-part-1

Conclusion

Richer Data with Tradeoffs

When Where How

Enriching Data

Elastic Stack

Philipp Krenn **@xeraa**