



An overview of the Elastic Stack geospatial capabilities

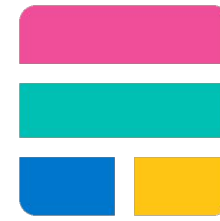
Jorge Sanz and Thomas Neiryneck

jorge.sanz@elastic.co - thomas@elastic.co

<https://ela.st/foss4g-2021>

2021-10-01 - FOSS4G 2021 Buenos Aires

Agenda



What is the Elastic Stack

Quick intro to the different components of the Elastic portfolio of products



Ingesting geospatial data

Different approaches to upload geodata into your cluster



Search and aggregate

Core processes to analyze geospatial data



Visualize

Render geospatial data at scale



The Elastic Stack

We build search solutions on a single stack

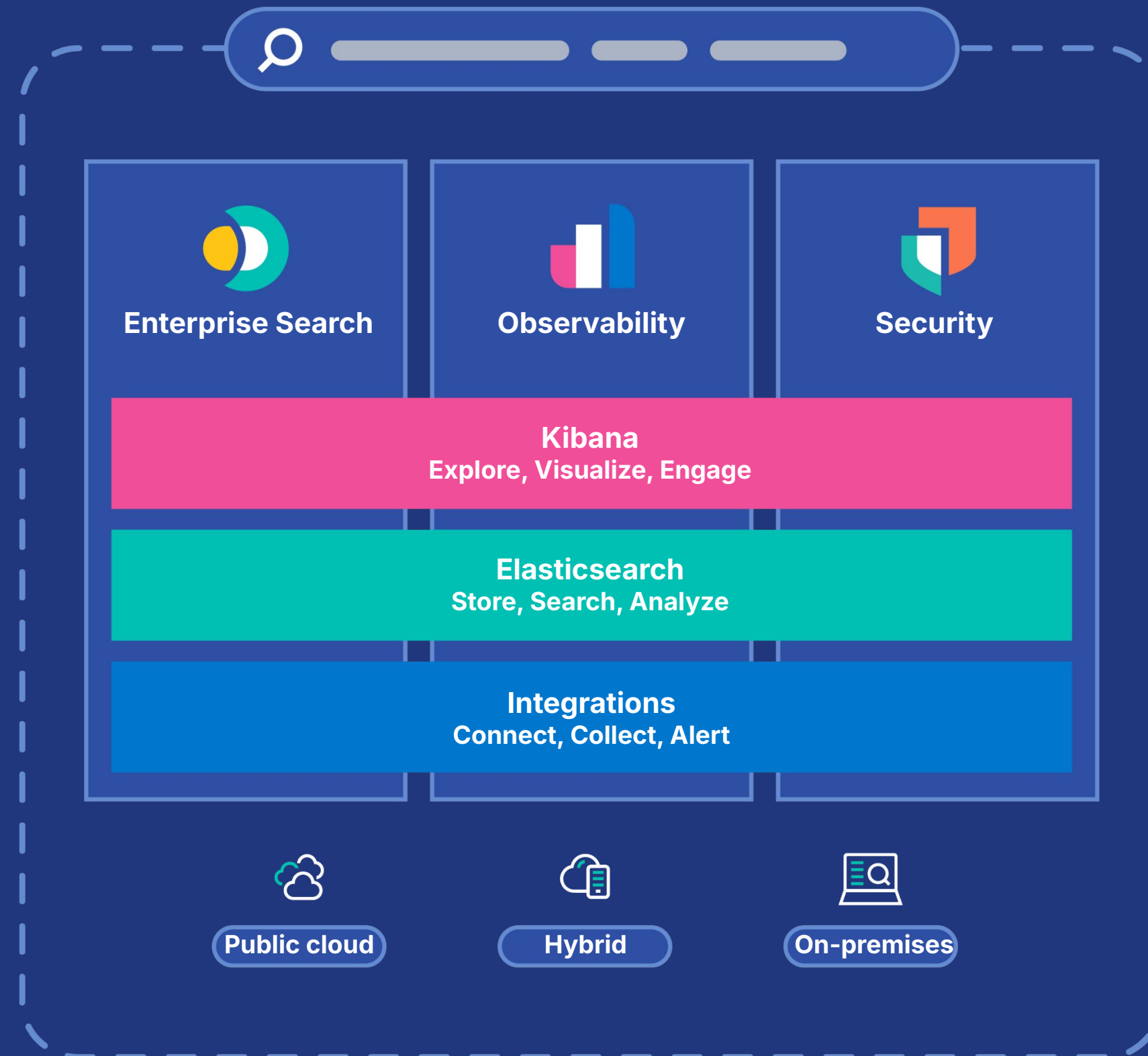
Enterprise Search

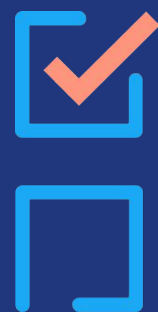
Observability

Security



The Elastic Search Platform





Speed

.....

Find matches in milliseconds
within structured and unstructured
datasets

Scale

.....

Scale massively and
horizontally across
hundreds of systems

Relevance

.....

Generate highly relevant
results and actionable
insights from data



Community

<https://github.com/elastic>

<https://ela.st/slack>

<https://discuss.elastic.co>

The screenshot shows the Elastic community forum interface. At the top is the Elastic logo and navigation icons. Below the header, there are filters for 'all categories' and 'all tags', followed by tabs for 'Categories', 'Latest', 'New (154)', 'Unread (21)', and 'Top'. A '+ New Topic' button is on the right. The main content area is divided into two columns. The left column lists categories: 'Announcements' (1 topic, 1 unread), 'Elastic Stack' (322 topics, 19 unread), 'Elastic Enterprise Search' (5 topics, 2 new), and 'Elastic Observability' (23 topics, 14 new). The right column shows a 'Latest' feed of topics, including 'Notes on Using These Forums', 'Logstash pipeline graceful shutdown: потеря in-memory данных?', 'Collapse within top hit aggregation results', 'Drilldown is not working with Visualization', 'Do not show results on page load', 'Custom transactions in checkout process', 'How to view sql queries in APM', and 'Installation seems to hang'.

Category	Topics	Latest
Announcements Release and security announcements and other bits about all of our Elastic products that we think will be useful to everyone. ■ Security Announcements ■ Community Ecosystem 1 unread	1 / week 1 unread 1 new	Notes on Using These Forums 2 ■ Meta Elastic Apr 2017
Elastic Stack Elasticsearch, Kibana, Beats, and Logstash - also known as the ELK Stack. Reliably and securely take data from any source, in any format, then search, analyze, and visualize it in real time. Please post your your topic under the relevant product category - Elasticsearch, Kibana, Beats, Logstash. ■ Elasticsearch 4 unread 59 new ■ Kibana 14 unread 32 new ■ Beats 20 new ■ Logstash 1 unread 18 new	322 / week 19 unread 129 new	Logstash pipeline graceful shutdown: потеря in-memory данных? 0 ■ Вопросы на русском языке 4m
Elastic Enterprise Search Easily implement powerful, modern search experiences for your busy team. Quickly add pre-tuned search to your website, app, or workplace. Search it all, simply. ■ App Search 2 new ■ Site Search ■ Workplace Search	5 / week 2 new	Collapse within top hit aggregation results 0 ■ Elasticsearch 5m
Elastic Observability Bring your logs, infrastructure and availability metrics, and APM traces together at scale in a	23 / week 14 new	Drilldown is not working with Visualization 2 ■ Kibana 9m
		Do not show results on page load 0 ■ App Search 20m
		Custom transactions in checkout process 0 ■ Elasticsearch 21m
		How to view sql queries in APM 6 ■ APM dotnet 23m
		Installation seems to hang 3

Ingesting data into Elasticsearch

Using Elastic stack and third party products to
upload geospatial datasets



Beats

Lightweight data shippers



FileBeat
CSVs



MetricBeat
System metrics



PacketBeat
Network Data



WinLogBeat
Window Events



HeartBeat
Uptime Monitoring



AuditBeat
Audit Data



FunctionBeat
Serverless Shipper

GeoIP

Plus, more than 70 community Beats and growing...



build passing

Earthquakebeat

Welcome to Earthquakebeat.

Earthquakebeat is a beat which periodically pulls data from [USGS earthquake API](#). There are 2 api calls done each `Period` which request new and updated earthquakes.

`New` earthquakes call will request data in GeoJSON format and use attribute `starttime` set to `Now-Period`. That means beat will pull data from past X Period of time you define. Example

```
https://earthquake.usgs.gov/fdsnws/event/1/query?format=geojson&starttime=2019-08-13T09%3A18%3A18
```

`Updated` earthquakes does the same, except it uses attribute `updatedafter` to pull last updated data. Example:

```
https://earthquake.usgs.gov/fdsnws/event/1/query?format=geojson&starttime=2019-08-13T09%3A18%3A18
```

All other attributes are default and earthquakes from all over the world are being pulled.

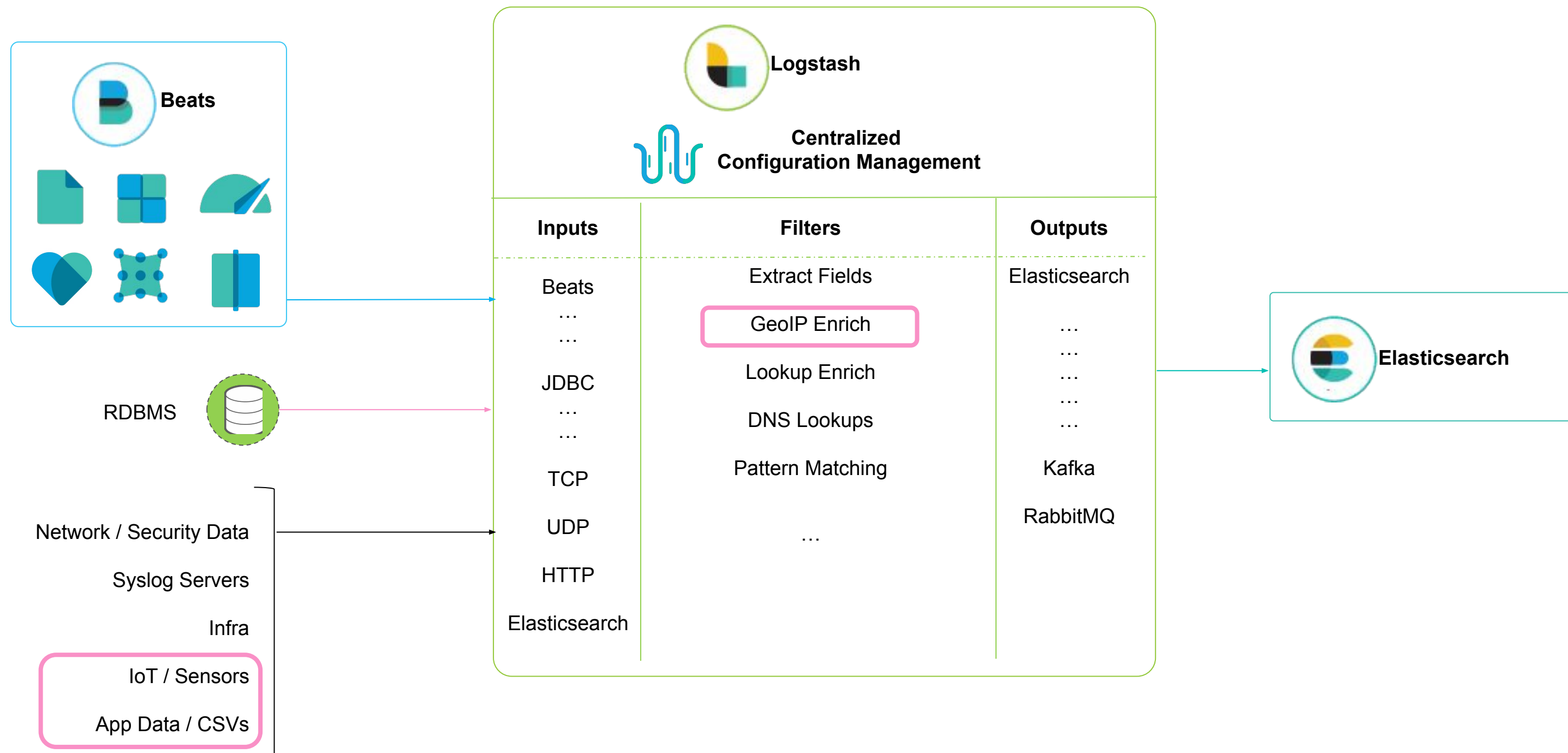
Note: Beat preserve earthquake original ID to not to duplicate data in index.

<https://github.com/radoondas/earthquakebeat>





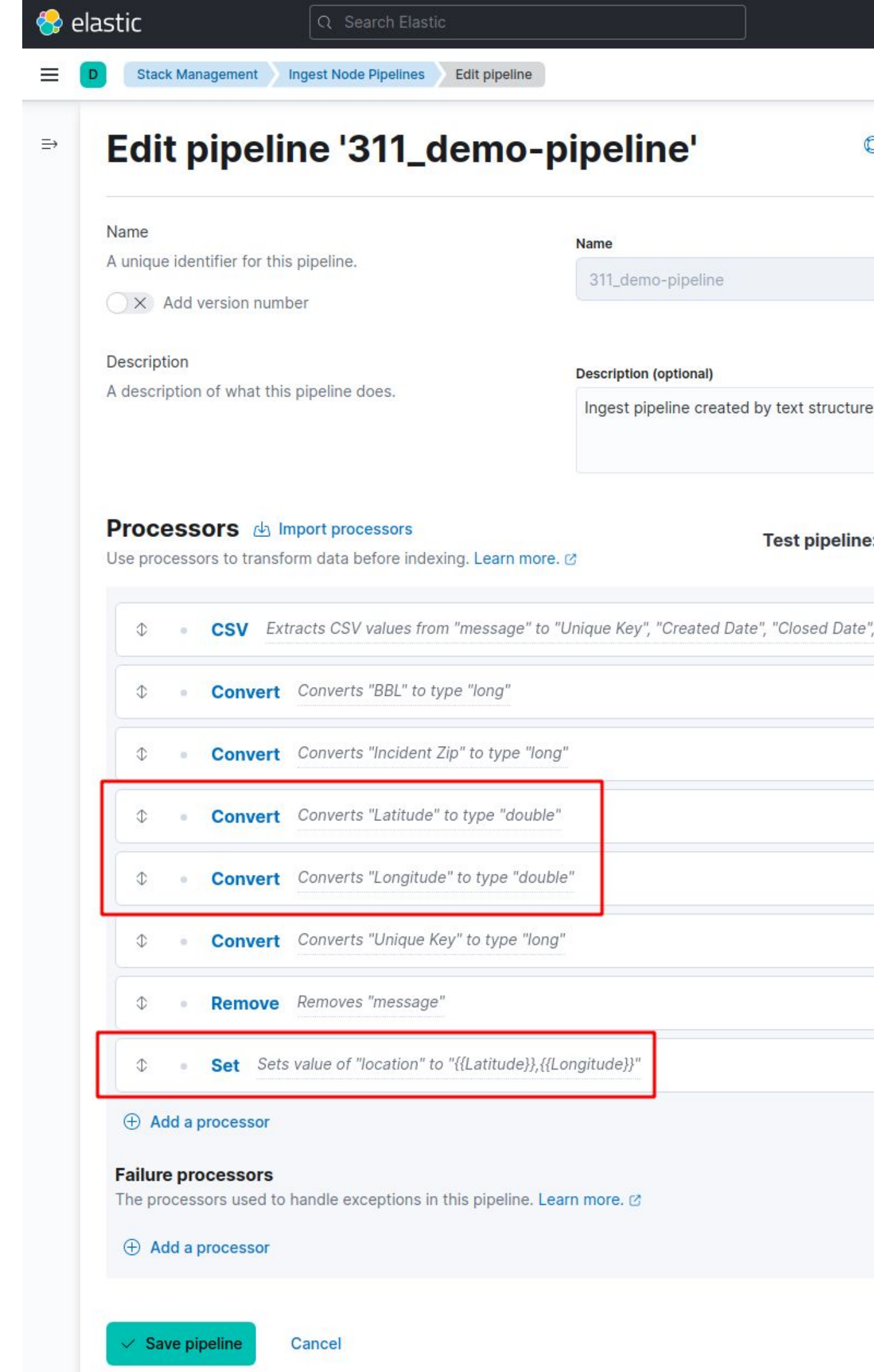
Logstash

Normalize and Enrich Data before Indexing



Ingest pipelines

- Transform data at ingest time
- They can run in dedicated nodes 
- Processors 
 - convert, set/remove fields, ...
 - **geoip, enrich**
- Blog: [How to map custom boundaries in Kibana with reverse geocoding](#)
- Convenient UI in Kibana Stack Management



The screenshot shows the 'Edit pipeline' interface in Kibana Stack Management. The pipeline is named '311_demo-pipeline'. It has a description: 'Ingest pipeline created by text structure'. The 'Processors' section lists several processors: 'CSV' (Extracts CSV values from 'message' to 'Unique Key', 'Created Date', 'Closed Date'), 'Convert' (Converts 'BBL' to type 'long'), 'Convert' (Converts 'Incident Zip' to type 'long'), 'Convert' (Converts 'Latitude' to type 'double'), 'Convert' (Converts 'Longitude' to type 'double'), 'Convert' (Converts 'Unique Key' to type 'long'), and 'Remove' (Removes 'message'). The 'Set' processor is highlighted with a red box, showing it sets the value of 'location' to '{{Latitude}},{{Longitude}}'. The 'Failure processors' section is empty. At the bottom, there are 'Save pipeline' and 'Cancel' buttons.

Edit pipeline '311_demo-pipeline'

Name: 311_demo-pipeline

Description: Ingest pipeline created by text structure

Processors [Import processors](#) [Test pipeline](#)

Use processors to transform data before indexing. [Learn more.](#)

- CSV** Extracts CSV values from "message" to "Unique Key", "Created Date", "Closed Date"
- Convert** Converts "BBL" to type "long"
- Convert** Converts "Incident Zip" to type "long"
- Convert** Converts "Latitude" to type "double"
- Convert** Converts "Longitude" to type "double"
- Convert** Converts "Unique Key" to type "long"
- Remove** Removes "message"
- Set** Sets value of "location" to "{{Latitude}},{{Longitude}}"

[Add a processor](#)

Failure processors

The processors used to handle exceptions in this pipeline. [Learn more.](#)

[Add a processor](#)

[Save pipeline](#) [Cancel](#)

Ingest with ogr2ogr

<https://gdal.org/drivers/vector/elasticsearch.html>

- ogr2ogr can read and write into Elasticsearch
- Support for custom mapping definitions
- Blog posts:
 - [How to ingest geospatial data into Elasticsearch with GDAL](#)
 - [Import OSM data into Elasticsearch with ogr2ogr and Docker](#)



Have you used [Elastic Maps](#) in Kibana yet? I am very excited about multi-layer support. Heat maps, vector layers from the Elastic Maps Service, individual documents all in the same interface! What a fantastic way to and visualize your data.

But what about geospatial data that's not in Elasticsearch? Maybe you overlay a shapefile of regional sales territories with sales aggregations. you have a CSV file of distribution center locations, and you want to get data into Elasticsearch, but configuring Filebeat or Logstash is not ideal for ingesting static datasets. Well, we have the perfect solution for you: GDAL.

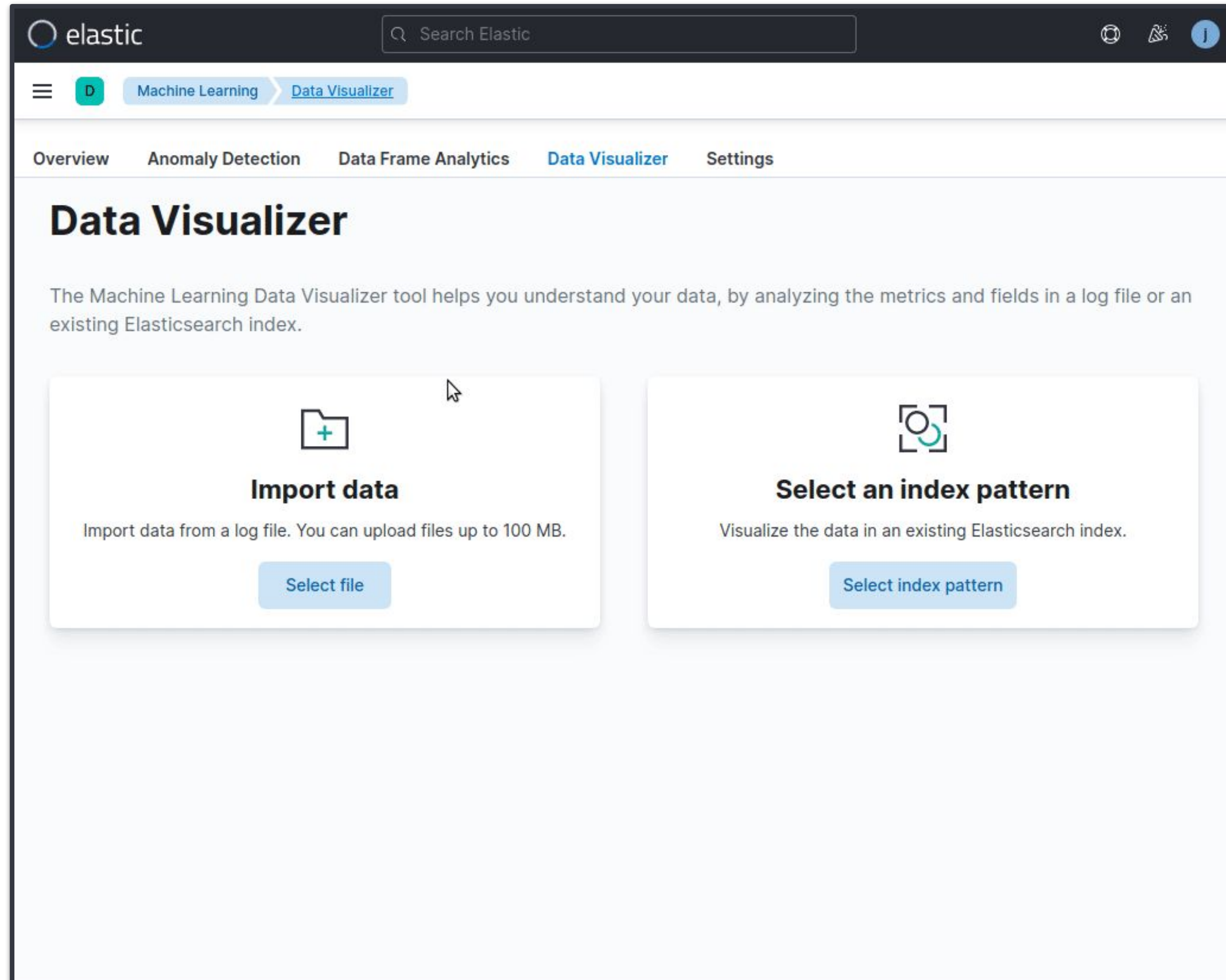
[GDAL](#) (Geospatial Data Abstraction Library) contains command line tools that can convert geospatial data between over 75 different geospatial file formats including [Elasticsearch](#). GDAL can be [compiled from source](#) or [installed via package managers](#). GDAL can also be installed via [Homebrew OSGeo4Mac](#) (ex. `brew tap osgeo/osgeo4mac && brew install osgeo-gdal`). Note, you need to have GDAL v3.1 or later to ingest data into Elasticsearch 7.x.

Connecting to Elasticsearch

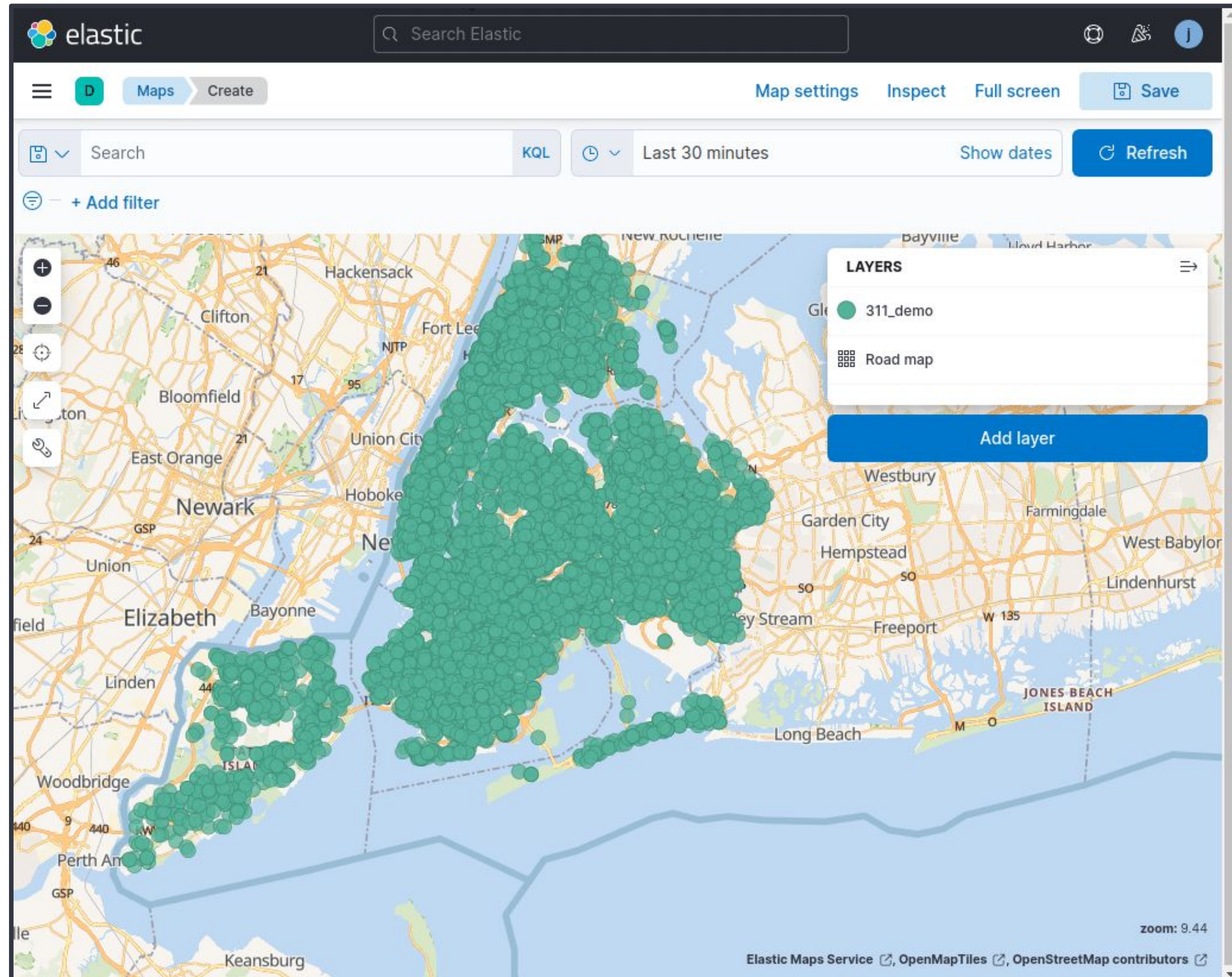
Once you've installed GDAL, open your command line or terminal window and try connecting to your Elasticsearch cluster using the `ogrinfo` tool. We use the URL with "ES:" to tell GDAL to use the Elasticsearch driver.



```
ogrinfo ES:http://localhost:9200/
```


Ingest with Kibana: CSV file upload



Ingest with Kibana: GeoJSON upload






- Docs 
- Tutorial 

Store, search, aggregate

Data types, Elastic query DSL,
geospatial aggregations

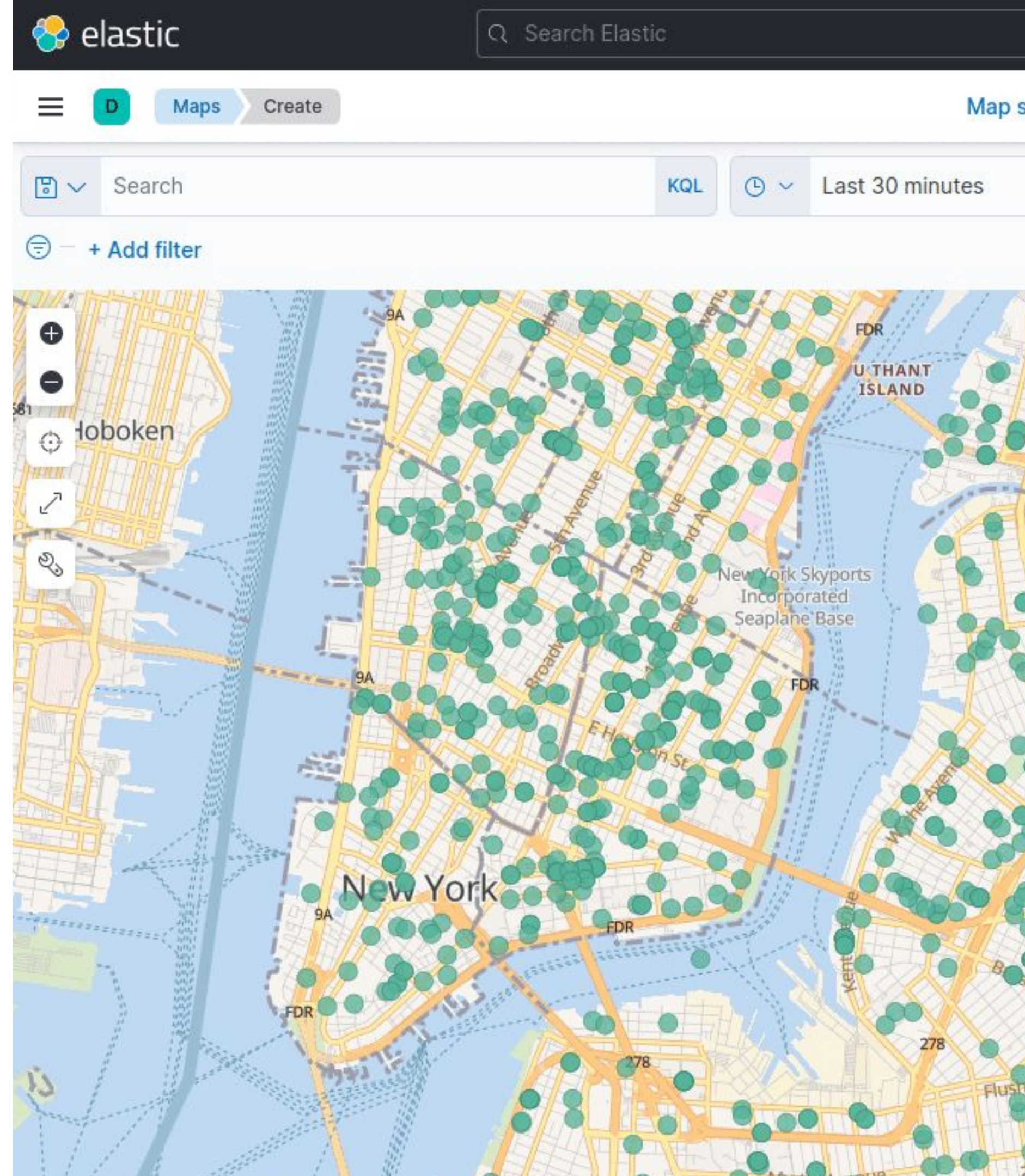
Elasticsearch geospatial data types

- `geo_point` 
 - A single pair of latitude and longitude **coordinates**
 - Can be inserted as an object, WKT, array, geohash
- `geo_shape` 
 - Supports any **lat/lon** geometry type, incl. envelope and circle
 - Inserted with GeoJSON or WKT notation
- `shape` 
 - Supports any **cartesian** geometry type
 - Inserted with GeoJSON or WKT notation

Elasticsearch queries

Filter documents with geospatial relationships

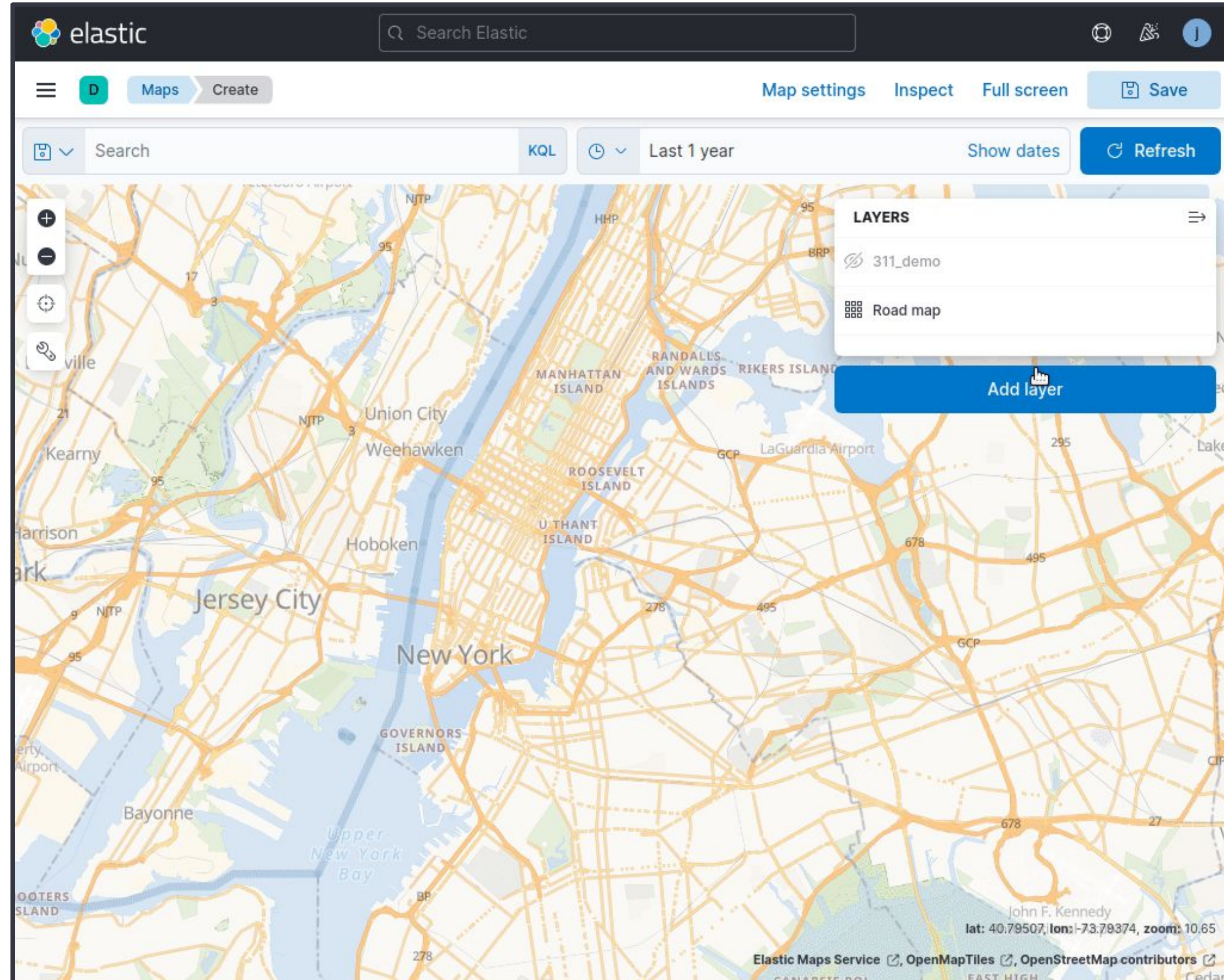
- Bounding box
- Point and radius
- Polygon
- An indexed geo_shape



Elasticsearch bucket aggregations

Bin documents based on their location into categories

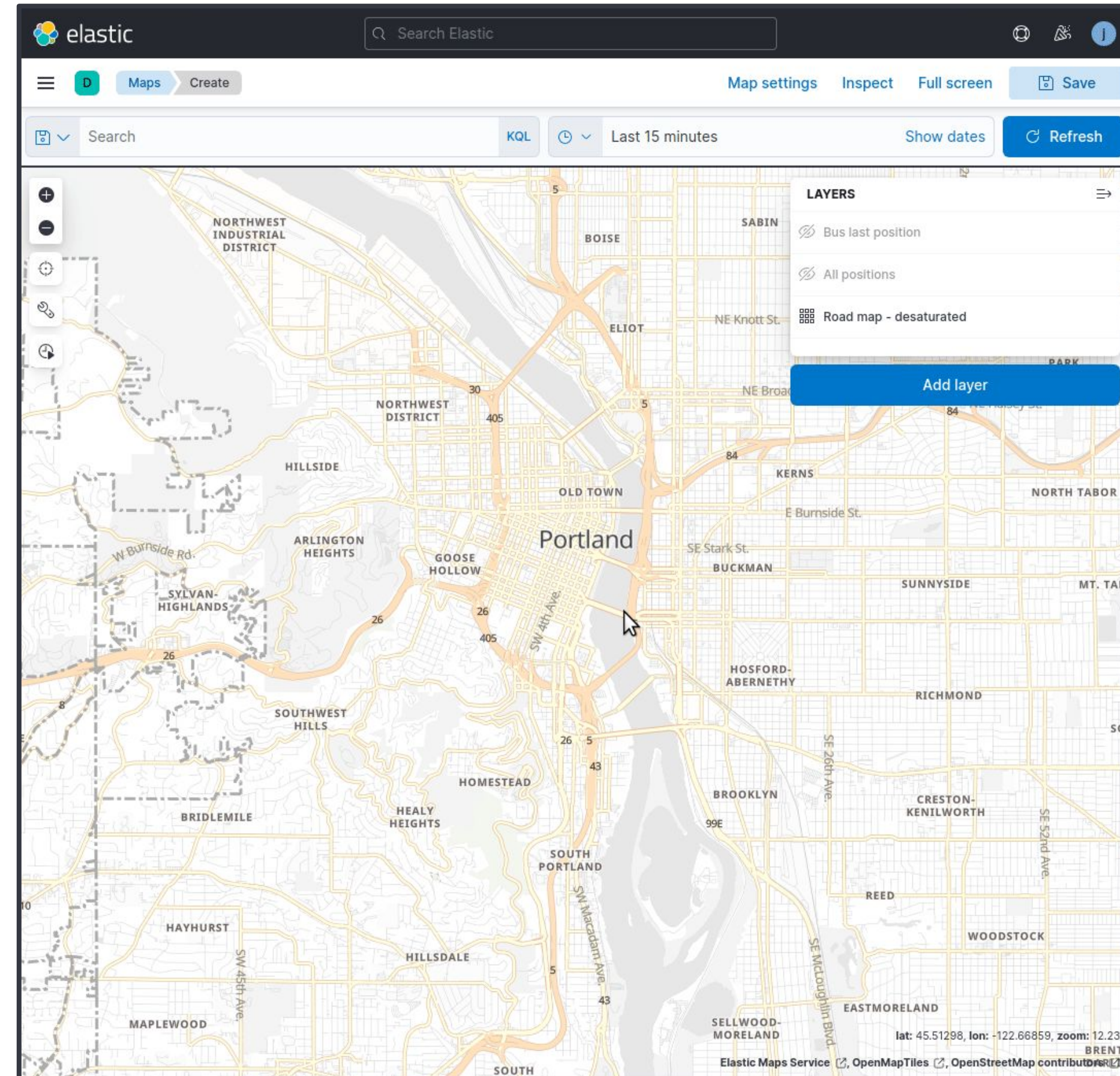
- Distance (rings) 📖
- Hash 📖
- Geotile 📖



Elasticsearch metric aggregations

Compute geospatial metrics derived from aggregating documents

- Centroid 📖
- Bounds 📖
- Geoline 📖



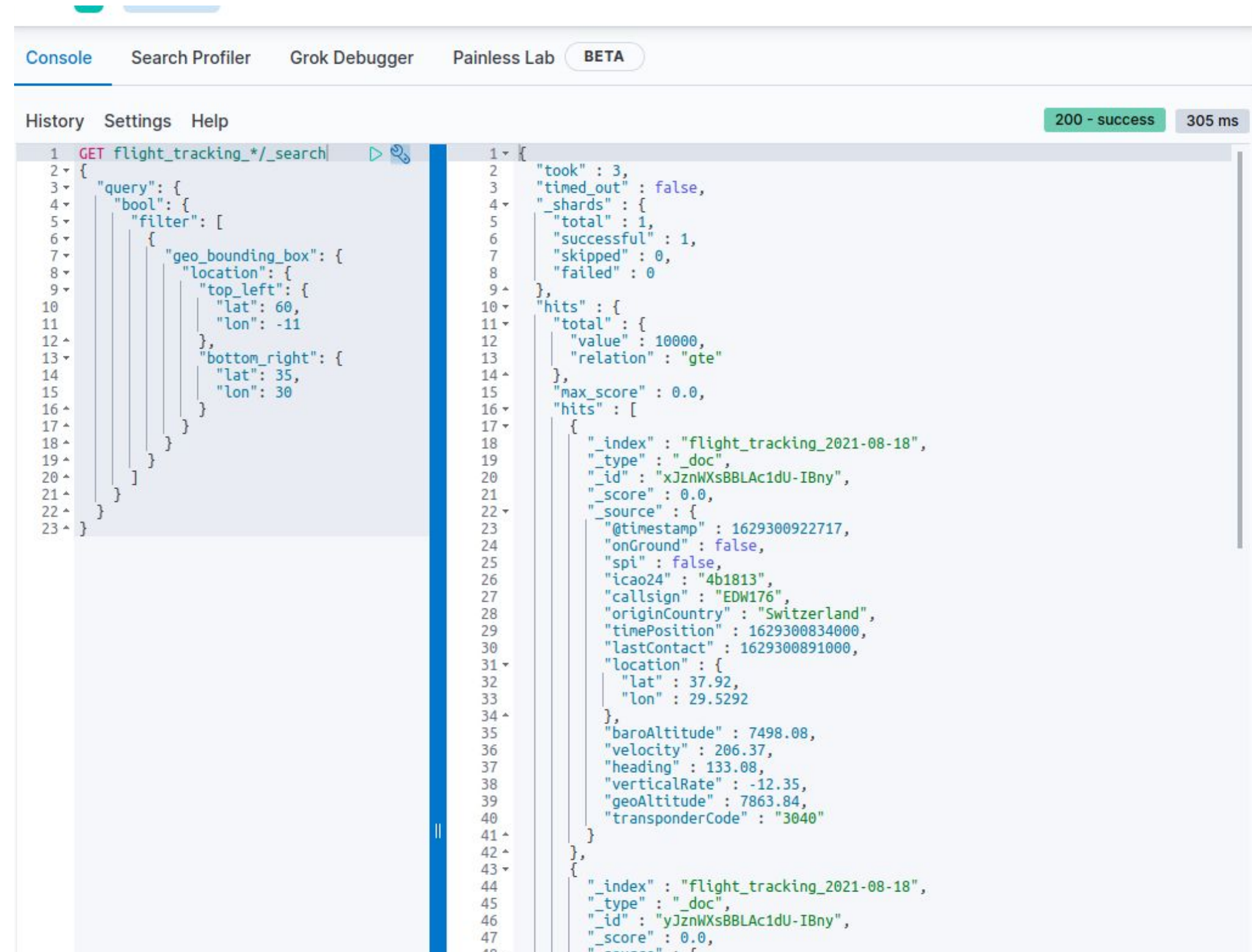
Visualization strategies

Render geospatial data with Kibana
or your own application

Elasticsearch JSON output

Getting data out to visualize

- **JSON** is the default output format
- `_search` limits to 10K docs
- Pagination needs a "point in time" to freeze the search context



The screenshot shows the Elasticsearch DevTools interface. The top navigation bar includes 'Console', 'Search Profiler', 'Grok Debugger', 'Painless Lab', and a 'BETA' badge. The 'Console' tab is active, displaying a search query and its results. The query is a GET request to `flight_tracking*/_search` with a filter based on a geographic bounding box. The response is a JSON object containing search metadata and a list of hits. The first hit is a document from the `flight_tracking_2021-08-18` index, representing a flight with various attributes like timestamp, location, and altitude.

```
1 GET flight_tracking*/_search
2 {
3   "query": {
4     "bool": {
5       "filter": [
6         {
7           "geo_bounding_box": {
8             "location": {
9               "top_left": {
10                "lat": 60,
11                "lon": -11
12              },
13              "bottom_right": {
14                "lat": 35,
15                "lon": 30
16              }
17            }
18          }
19        ]
20      }
21    }
22  }
23 }
```


```
1 {
2   "took" : 3,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 10000,
13      "relation" : "gte"
14    },
15    "max_score" : 0.0,
16    "hits" : [
17      {
18        "_index" : "flight_tracking_2021-08-18",
19        "_type" : "_doc",
20        "_id" : "xJznWXsBBLAc1dU-IBny",
21        "_score" : 0.0,
22        "_source" : {
23          "@timestamp" : 1629300922717,
24          "onGround" : false,
25          "spi" : false,
26          "icao24" : "4b1813",
27          "callsign" : "EDW176",
28          "originCountry" : "Switzerland",
29          "timePosition" : 1629300834000,
30          "lastContact" : 1629300891000,
31          "location" : {
32            "lat" : 37.92,
33            "lon" : 29.5292
34          },
35          "baroAltitude" : 7498.08,
36          "velocity" : 206.37,
37          "heading" : 133.08,
38          "verticalRate" : -12.35,
39          "geoAltitude" : 7863.84,
40          "transponderCode" : "3040"
41        }
42      },
43      {
44        "_index" : "flight_tracking_2021-08-18",
45        "_type" : "_doc",
46        "_id" : "yJznWXsBBLAc1dU-IBny",
47        "_score" : 0.0,
48        "_source" : {
49          "@timestamp" : 1629300922717,
50          "onGround" : false,
51          "spi" : false,
52          "icao24" : "4b1813",
53          "callsign" : "EDW176",
54          "originCountry" : "Switzerland",
55          "timePosition" : 1629300834000,
56          "lastContact" : 1629300891000,
57          "location" : {
58            "lat" : 37.92,
59            "lon" : 29.5292
60          },
61          "baroAltitude" : 7498.08,
62          "velocity" : 206.37,
63          "heading" : 133.08,
64          "verticalRate" : -12.35,
65          "geoAltitude" : 7863.84,
66          "transponderCode" : "3040"
67        }
68      }
69    ]
70  }
71 }
```

200 - success 305 ms

OGC servers and Elasticsearch

Expose Elasticsearch indices as OGC services

GeoServer



GeoServer

[About](#) | [Blog](#) | [Download](#) |

GeoServer 2.20.x User Manual » Community modules » Elasticsearch data store

[previous](#) | [next](#) | [modules](#)

Elasticsearch data store

Elasticsearch is a popular distributed search and analytics engine that enables complex search features in near real-time. Default field type mappings support string, numeric, boolean and date types and allow complex, hierarchical documents. Custom field type mappings can be defined for geospatial document fields. The `geo_point` type supports point geometries that can be specified through a coordinate string, geohash or coordinate array. The `geo_shape` type supports Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon and GeometryCollection GeoJSON types as well as envelope and circle types. Custom options allow configuration of the type and precision of the spatial index.

This data store allows features from an Elasticsearch index to be published through GeoServer. Both `geo_point` and `geo_shape` type mappings are supported. OGC filters are converted to Elasticsearch queries and can be combined with native Elasticsearch queries in WMS and WFS requests.

Contents:

- [Elasticsearch data store](#)
 - [Configuration](#)
 - [Configuring data store](#)
 - [Configuring authentication](#)
 - [Configuring HTTPS/SSL](#)

Table Of Contents

- Elasticsearch data store
 - » Configuration
 - » Configuring data store
 - » Configuring authentication
 - » Configuring HTTPS/SSL
 - » Configuring layer
 - » Configuring logging
 - » Filtering
 - » Native queries
 - » Examples
 - » Aggregations
 - » Geohash grid aggregations
 - » Grid Strategy
 - » Basic
 - » Metric
 - » Nested
 - » Implementing a custom Grid Strategy
 - » FAQ

Continue Reading

- » Previous: Optimize rendering of complex polygons
- » Next: GeoMesa data store

pygeoapi

OpenAPI

Data publishing

Providers overview

Publishing vector data to OGC API - Features

Providers

Connection examples

Data access examples

Publishing raster data to OGC API - Coverages

Publishing map tiles to OGC API - Tiles

Publishing processes via OGC API - Processes

Publishing metadata to OGC API - Records

Publishing data to OGC API - Environmental Data Retrieval

Publishing files to a SpatioTemporal Asset Catalog

Customizing pygeoapi: plugins

Elasticsearch

Note

Elasticsearch 7 or greater is supported.


To publish an Elasticsearch index, the following are required in your index:

indexes must be documents of valid GeoJSON Features

index mappings must define the GeoJSON `geometry` as a `geo_shape`

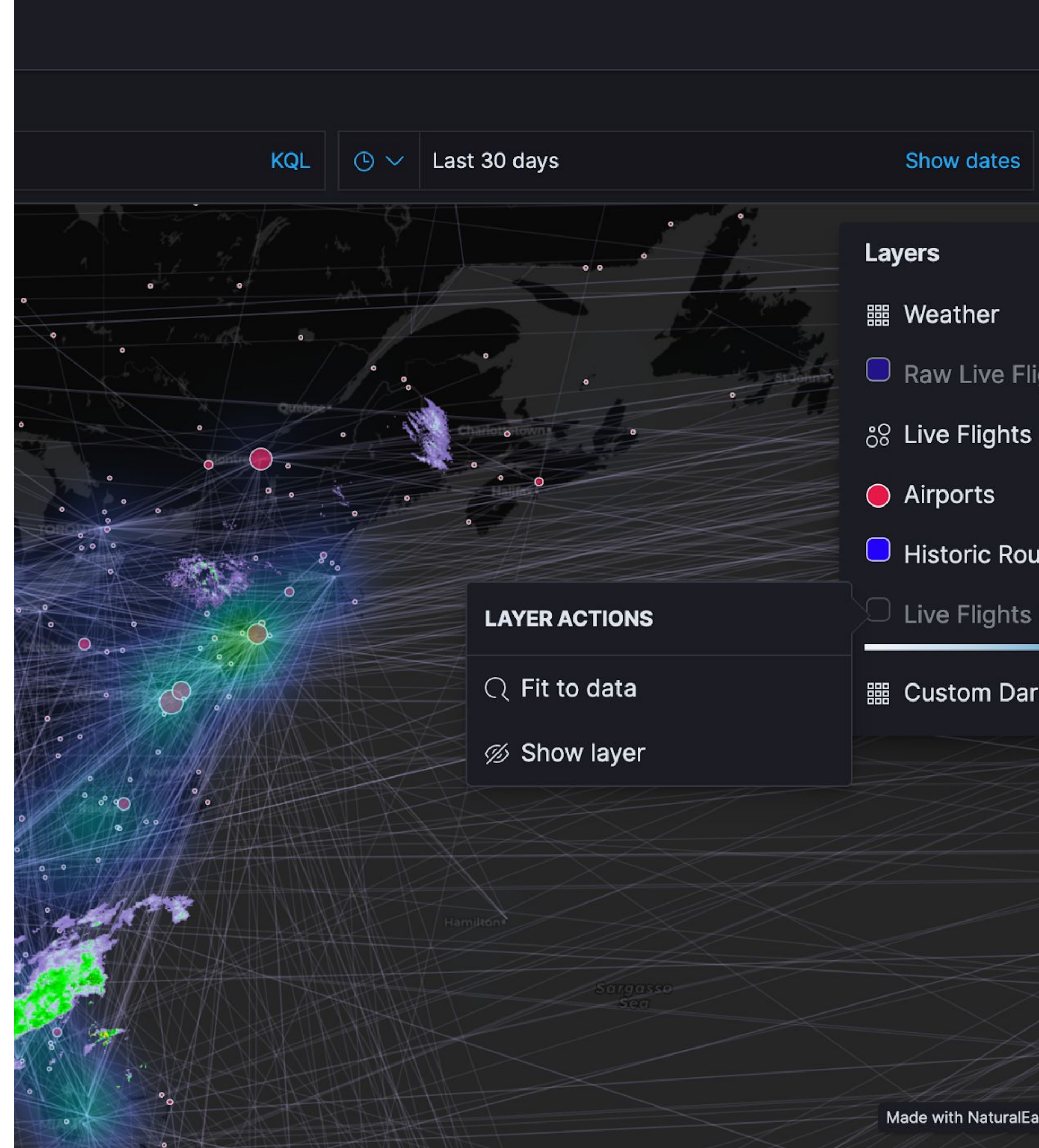
```
providers:  
- type: feature  
  name: Elasticsearch  
  data: http://localhost:9200/ne_110m_populated_places_simple  
  id_field: geonameid  
  time_field: datetimefield
```

This provider has the support for the CQL queries as indicated in the table above.



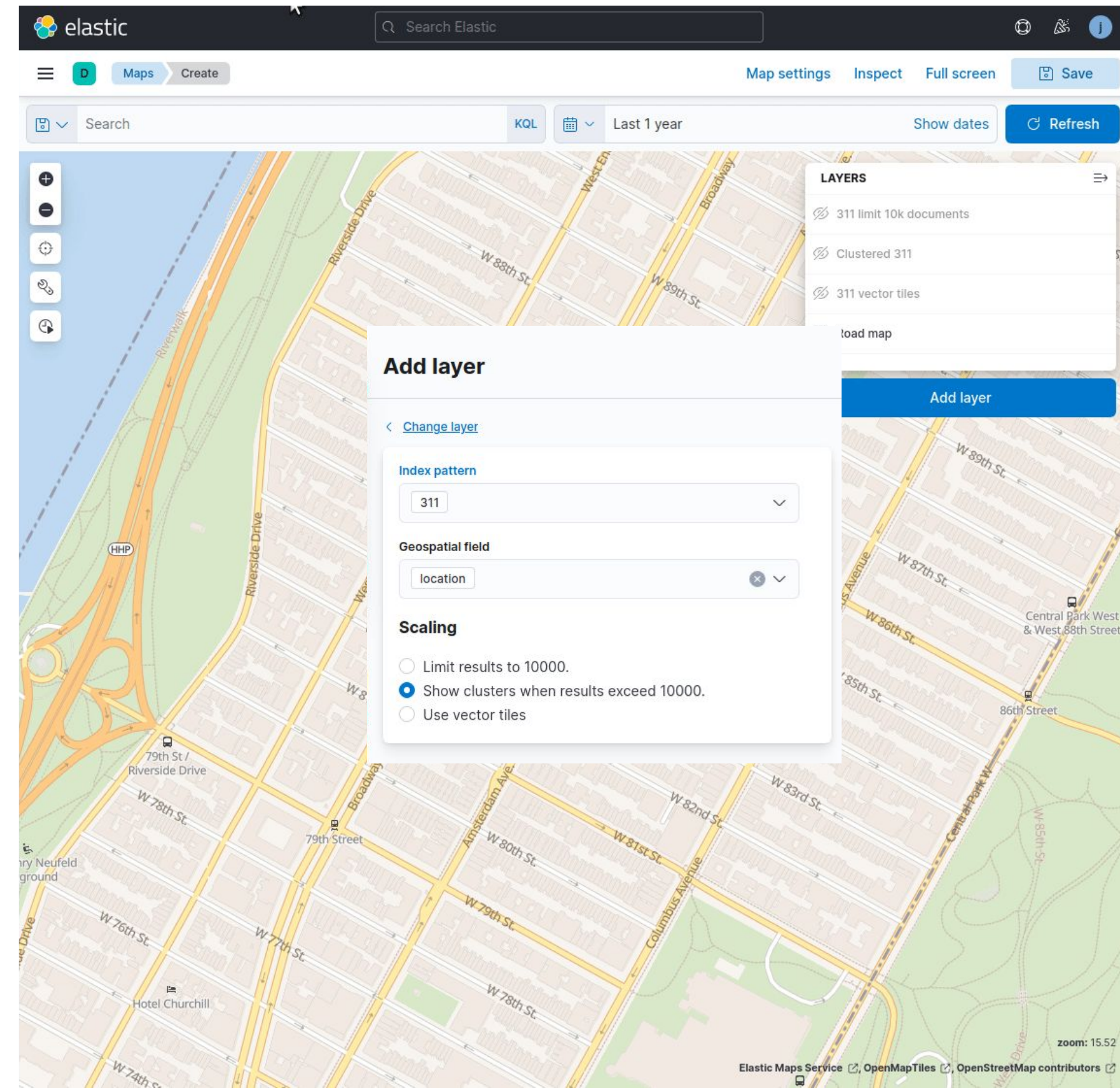
Elastic Maps

- Friendly user experience
- MapboxGL/MapLibre
- `geo_point` and `geo_shape`
- Aggregations: heat map, clustering, grids, geoline
- Data driven styling
- Tools for drawing, filtering, measuring
- Used alone or in dashboards or Canvas workdpads
- The map component for other Kibana applications



Elastic Maps rendering strategies

- Single request with up to 10K documents by map extent
- Automatic clustering when >10K documents
- Vector tiles with up to 10K documents per tile (better caching)

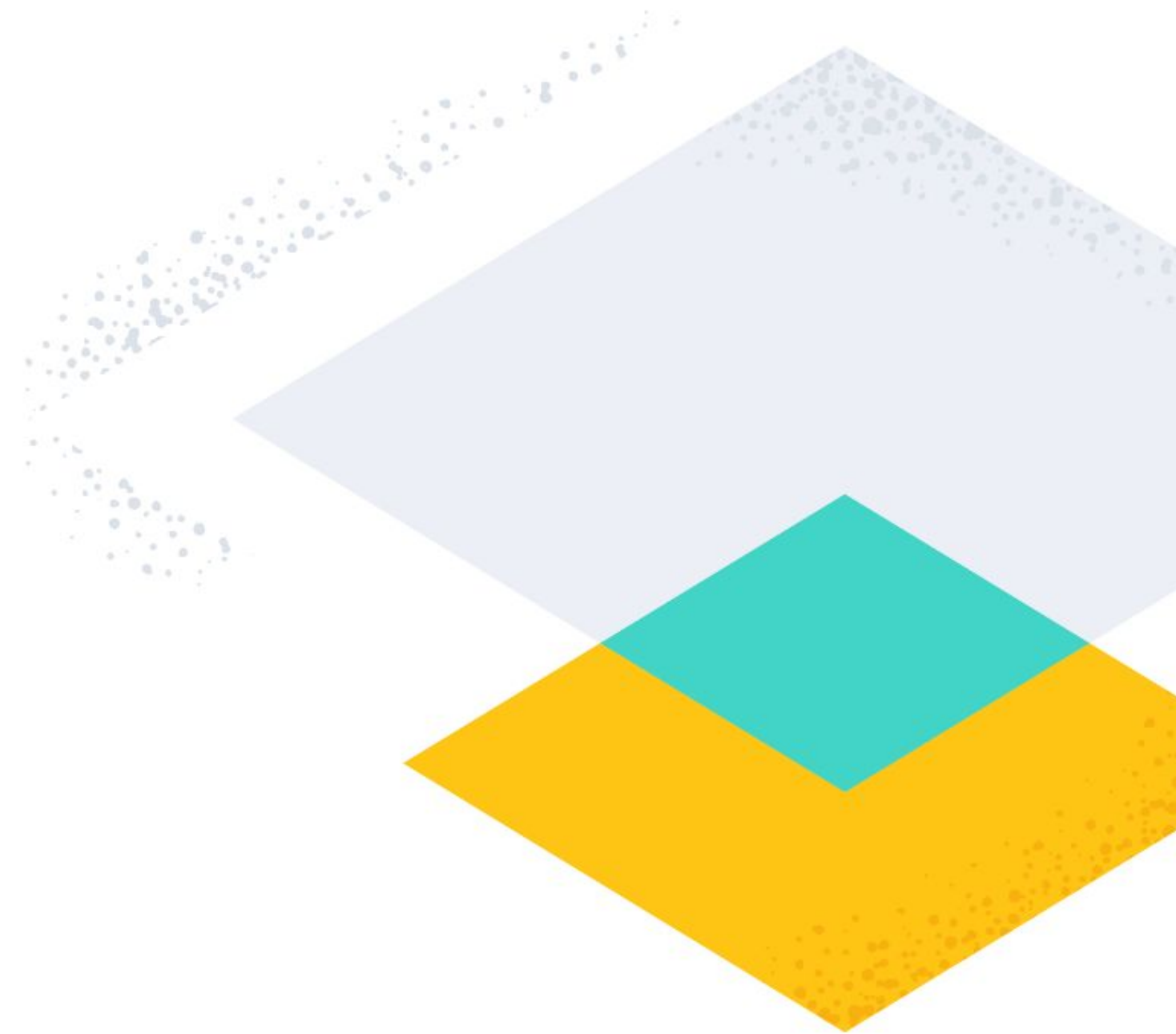


New features

Vector Tiles


A new output format for Elasticsearch

- New endpoint: `_mvt` (7.15)
- Output in protobuf format
- To be consumed by a middleware to configure filters, aggregations, etc.
- Will replace current vector tiles implementation in Kibana Elastic Maps



Runtime fields and geo

Elasticsearch is not anymore only a “schema on write” database

- Ability to define new fields at query time
- New in **7.14**: get centroid, height, and width of `geo_shape` fields with Painless scripting 
- More geospatial functionality will come

Wrap up

- **Download** the stack or start a **Elastic Cloud** 14 days free trial from <https://www.elastic.co/downloads>
- Use any of the **ingest** tools to upload your own data
- **Explore** and **visualize** with Elastic Maps and Kibana
- Share your feedback and questions at discuss.elastic.co



Thanks!

Jorge Sanz and Thomas Neirynck

jorge.sanz@elastic.co - thomas@elastic.co

<https://ela.st/foss4g-2021>

<https://ela.st/elasticon21-geo-kibana>

2021-10-01 - FOSS4G 2021 Buenos Aires