

Buscadores

Sesion III

Agenda

- Kibana:
 - Capacidades de visualización
 - Captura con Metricbeats
 - Series Temporales
 - Carga de documentos
- Otros componentes del ecosistema buscadores

- Visualización y exploración de datos alojados en Elasticsearch
- Aplicación desarrollada en java
- Lista de funcionalidades más relevantes:
 - Visualización
 - Exploración
 - Tableros de mando preconfigurados
 - Monitorización

<https://www.elastic.co/es/products/kibana/features>

Preparamos el entorno

- Descargamos y arrancamos elasticsearch si no lo has hecho ya antes

<https://www.elastic.co/es/downloads/elasticsearch>

- Hacemos lo mismo con Kibana

<https://www.elastic.co/downloads/kibana>

- Conectate a la consola principal de kibana:

<https://localhost:5601>

- Pulsando en el icono superior izquierdo ve al home y descarga el juego de datos de ejemplo:



Desde Home en Kibana

The screenshot shows the Kibana interface with the 'elastic' logo and a search bar at the top. Below the navigation bar, the 'Integrations' tab is selected, and the 'Sample data' sub-tab is active. The main heading is 'More ways to add data', followed by the text: 'In addition to adding [integrations](#), you can try our sample data or upload your own data.'

Below this text are two tabs: 'Sample data' (selected) and 'Upload file'. The 'Sample data' tab displays three cards, each representing a different sample dataset:

- Sample eCommerce orders:** This card shows a preview of the eCommerce dashboard with metrics like 'Sum of revenue' (\$77,377.84), 'Median spending' (\$66.89), and 'Avg. items sold' (2.2). It includes a description: 'Sample data, visualizations, and dashboards for tracking eCommerce orders.' and an 'Add data' button.
- Sample flight data:** This card shows a preview of the flight dashboard with metrics like 'Total flights' (2,180), 'On-time' (24.5%), and 'Cancelled' (12.8%). It includes a description: 'Sample data, visualizations, and dashboards for monitoring flight routes.' and an 'Add data' button.
- Sample web logs:** This card shows a preview of the web logs dashboard with metrics like 'Visits' (1,609), 'Bounces' (4.2%), and 'Bounces' (3.1%). It includes a description: 'Sample data, visualizations, and dashboards for monitoring web logs.' and an 'Add data' button.

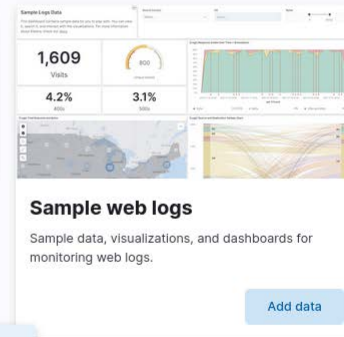
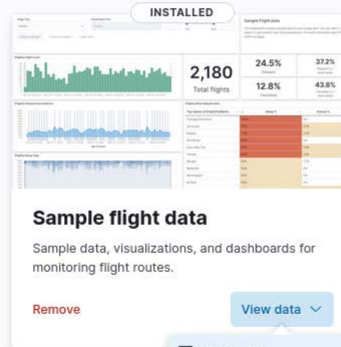
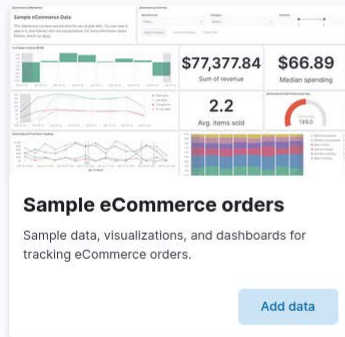
Ejemplos de Sample Data

- Seleccionamos un juego de datos, lo cargamos y vemos el ejemplo

More ways to add data

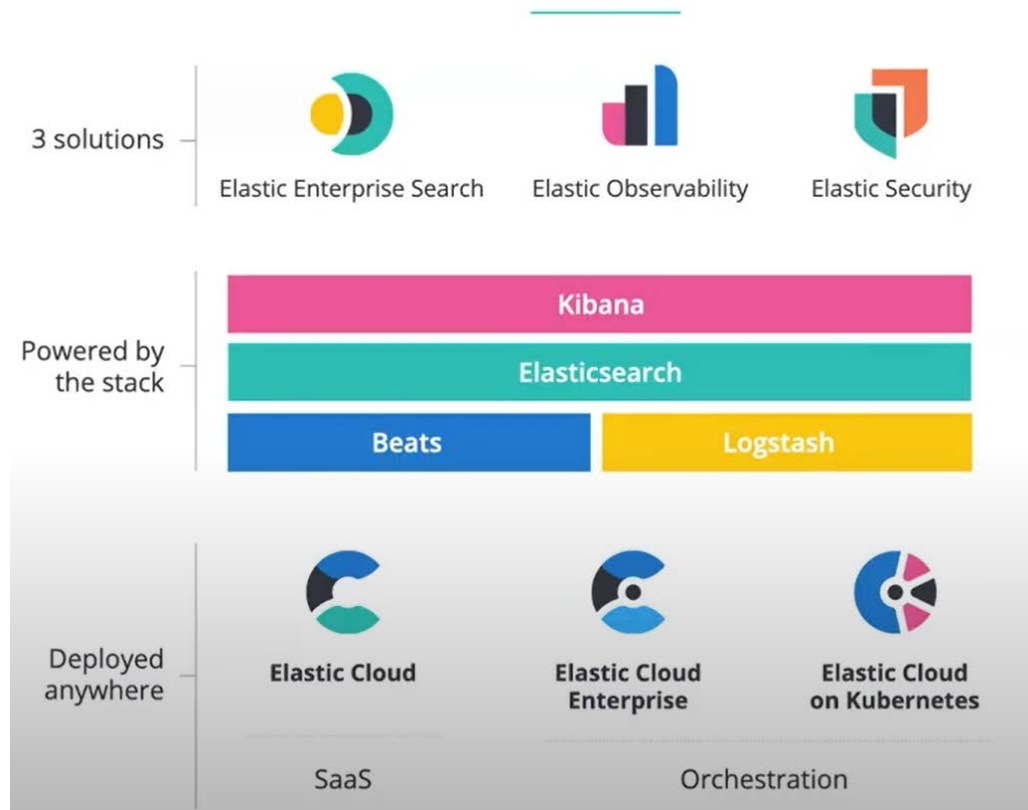
In addition to adding [integrations](#), you can try our sample data or upload your own data.

[Sample data](#) Upload file



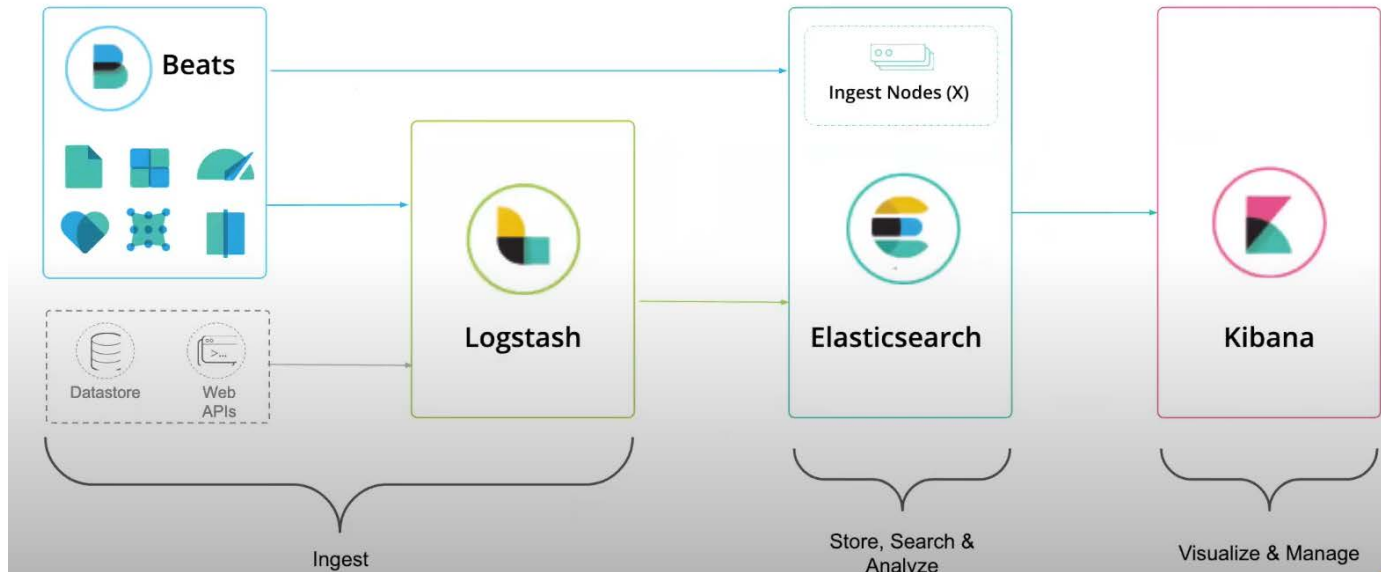
- Dashboard
- Canvas
- Map

Elastic Ecosystem



Cómo trabajan juntos

Se ingestan datos con Beats y/o Logstash y se gestionan y visualizan con Kibana

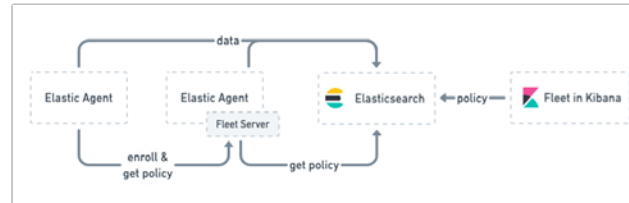


Entrada de Datos: Beats

- Beats son los agentes de transporte que se instalan en los servidores para enviar datos operacionales a Elasticsearch
- Están clasificados en familias:
 - Auditoría
 - Ficheros de logs y diarios de movimientos
 - Cloud
 - Disponibilidad
 - Métricas
 - Tráfico de red
 - Eventos de Windows
- Beats pueden enviar los datos directamente a Elasticsearch o a través de Logstash

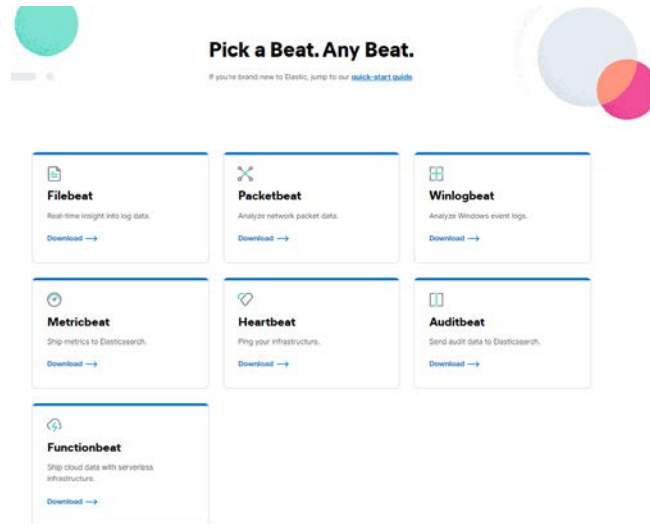
Entrada de Datos: Elastic-Agents

- Elastic Agent es un agente único para registros, métricas, datos de seguridad y prevención de amenazas.
- Se puede desplegar en dos modos diferentes:
 - Gestionado por Fleet. Un único punto para gestionar los agentes y su política/configuración
 - Modo autónomo. Una vez instalado, toda la configuración se aplica al Elastic Agent manualmente.



Plataforma del Master

- En la plataforma está ya instalado metricbeat al nivel 7.10.0 (compatible con la versión 7.16.3 de elasticsearch que estamos utilizando)



Para descargar otras versiones: <https://www.elastic.co/downloads/beats/>

Metricbeat en Linux

- La configuración por defecto se encuentra en `/etc/metricbeat/metricbeat.yml`
- No es necesario modificar el fichero que se encuentra en la instalación
- Para actualizar kibana e incluir los informes adecuados ejecutamos (aparecerán avisos de API deprecado que no nos afectan)

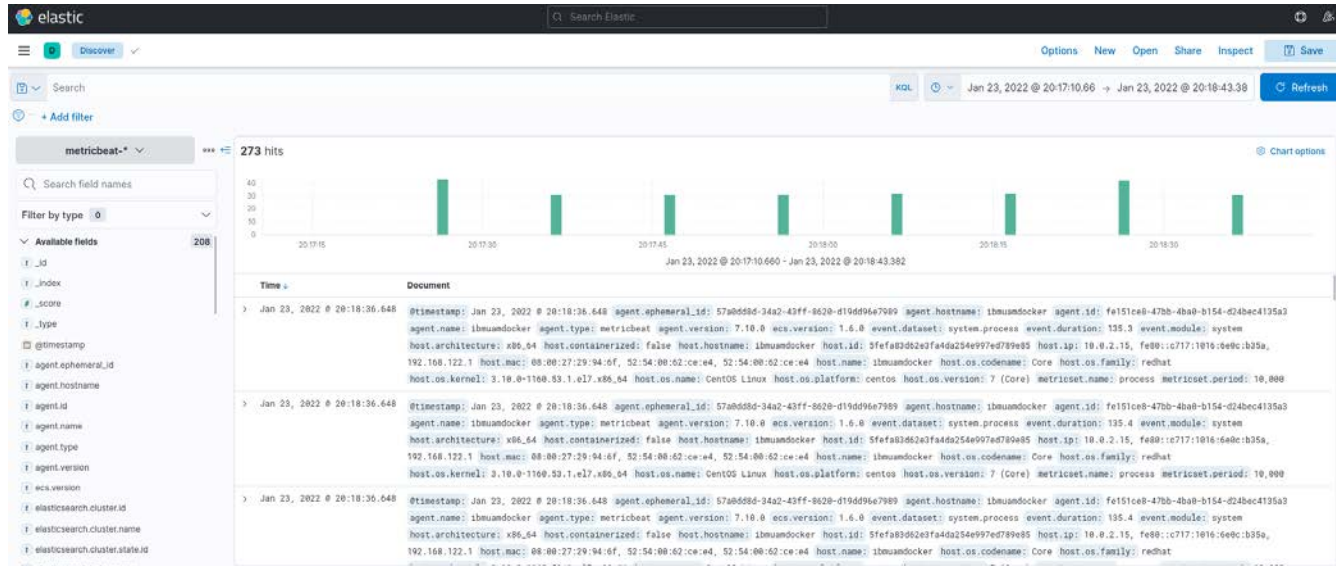
```
sudo metricbeat setup
```

- Y para arrancar el servicio y que envíe información a elasticsearch ejecutamos

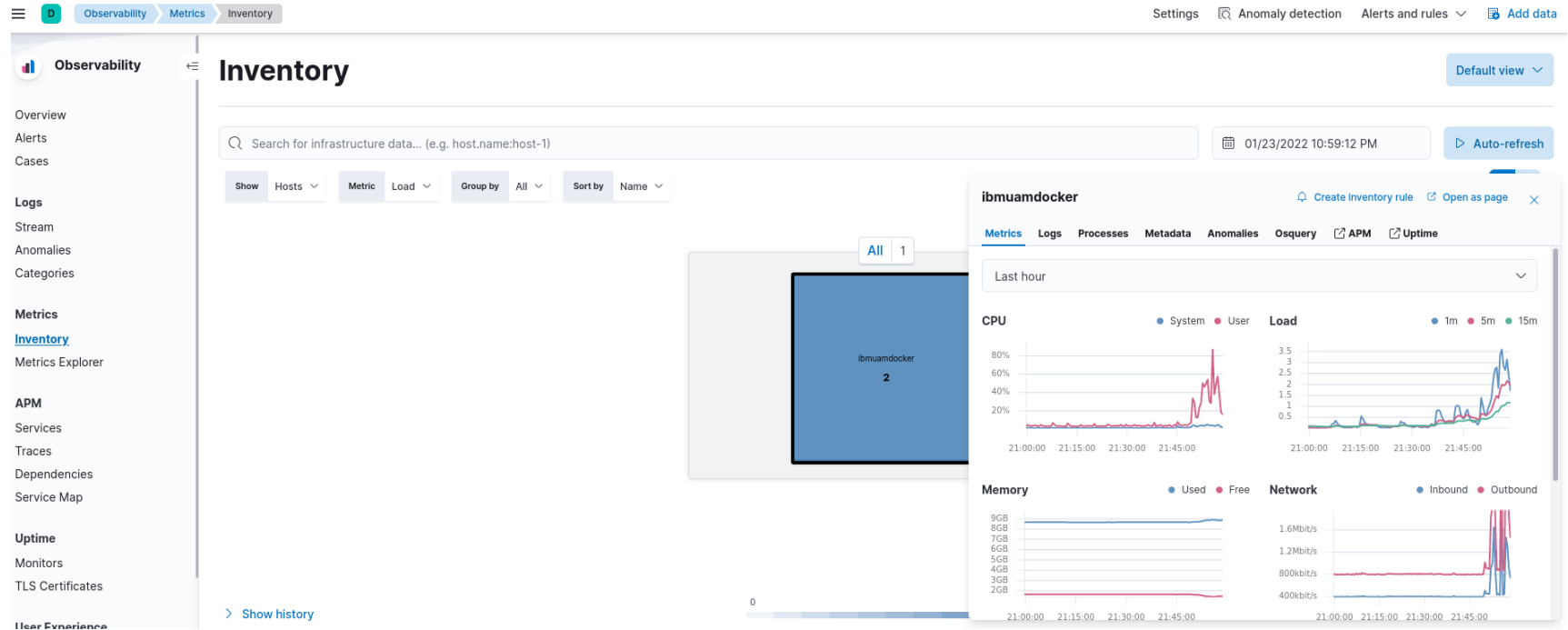
```
systemctl start metricbeat
```

Utiliza la opción de Discovery

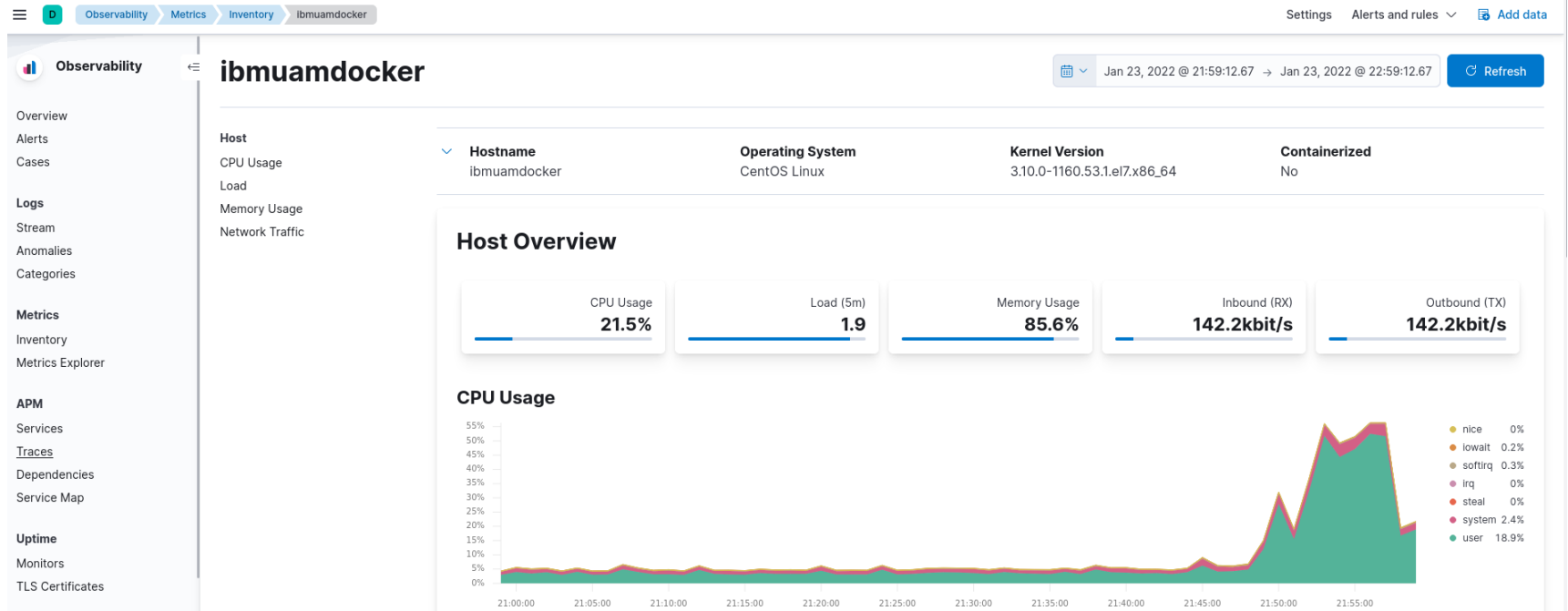
Una vez tienes arrancado el servicio, consulta la captura de datos en las opciones de Discover, Inventory o Explorer en Kibana



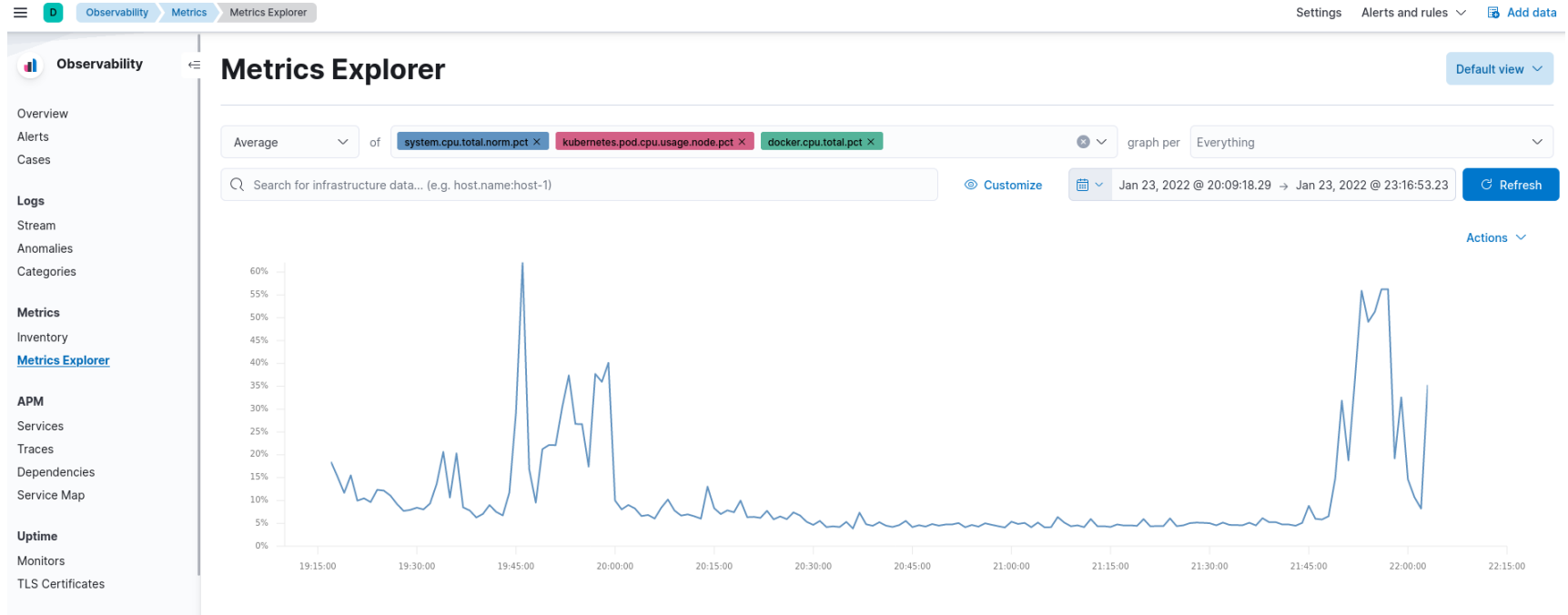
Metrics Inventory



Metrics Inventory

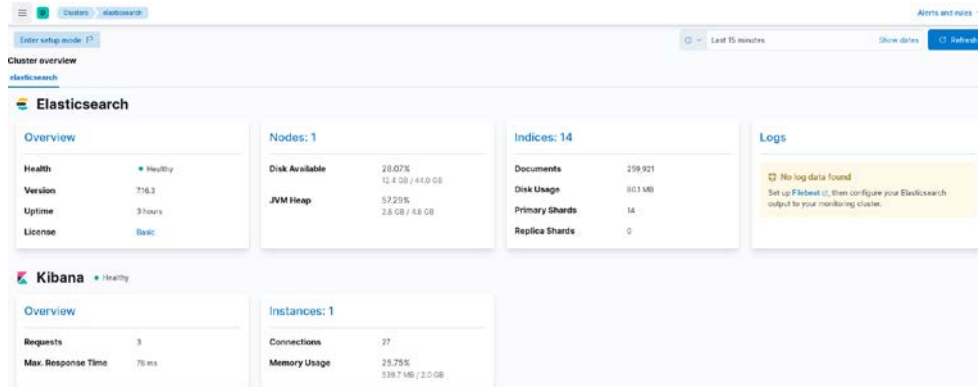


Metrics Explorer



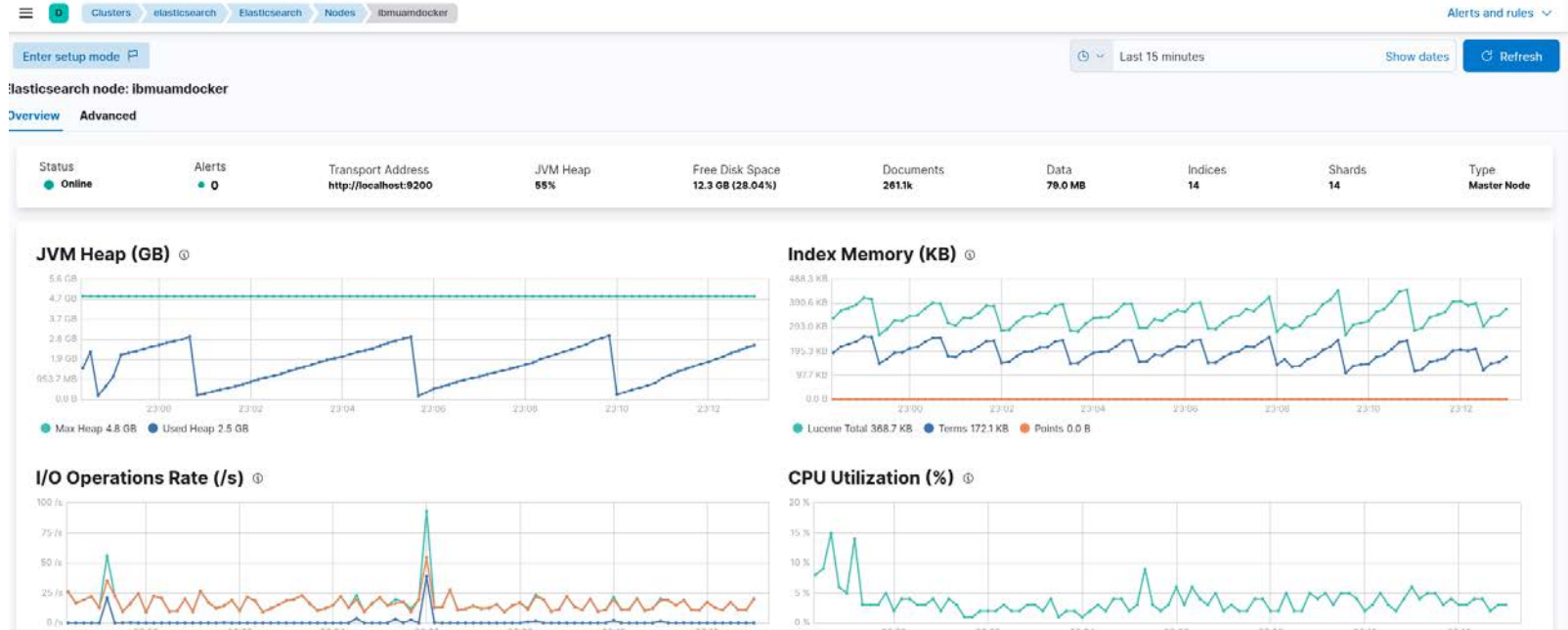
Monitorización

- Selecciona la opción de stack monitoring y activa la monitorización en el sistema a través de los siguientes comandos:
 - metricbeat modules enable elasticsearch-xpack
 - metricbeat modules enable kibana-xpack
- En algunos casos se hace necesario rearrancar los servicios (elasticsearch, kibana y metricbeat)



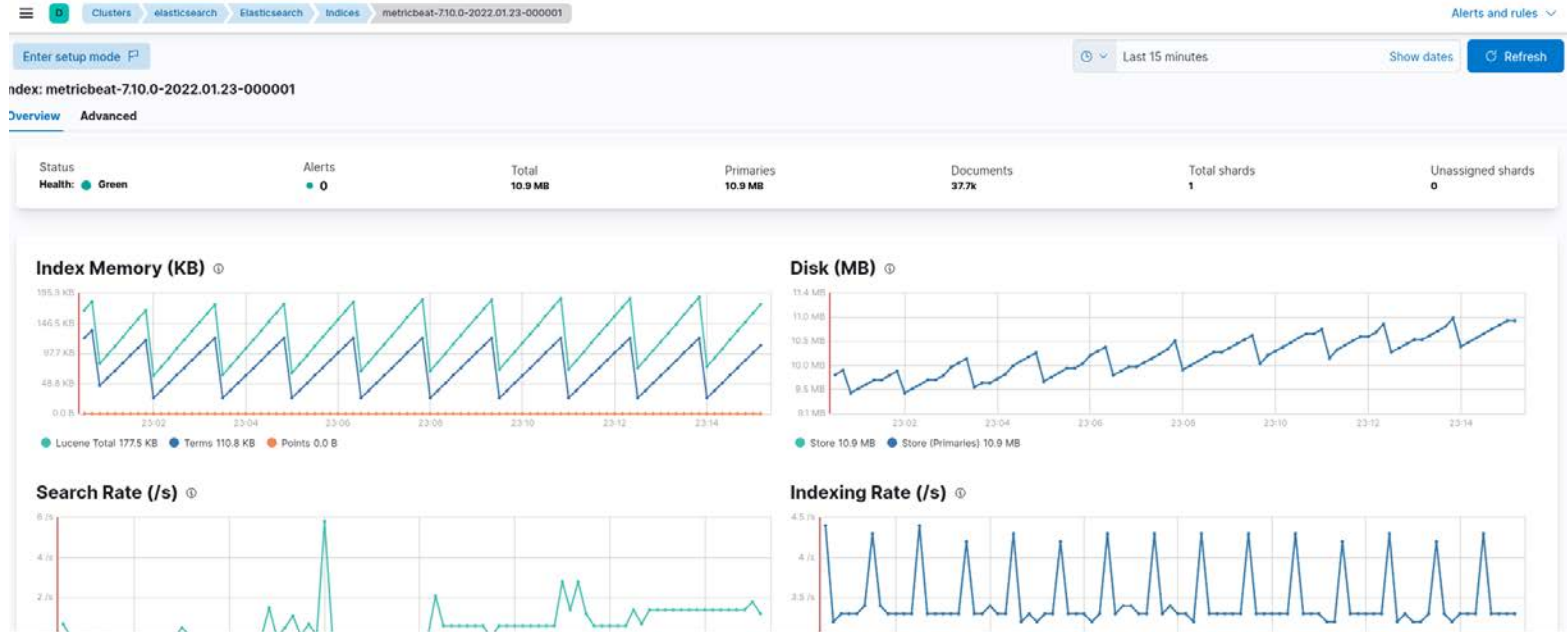
Monitorización. Visualización (I)

Selecciona, por ejemplo, Nodos y ibmuamdocker



Monitorización. Visualización (II)

Selecciona, por ejemplo, índices y metricbeat



Visualizemos otros datos: movierate

- Cojamos el juego de datos de Movierate
- Cargemos algunos datos utilizando la fecha correctamente
- Aunque se puede cambiar, la fecha por defecto de elasticsearch tiene que estar en el formato ISO 8601:

YYYY-MM-DDTHH:MM:SS

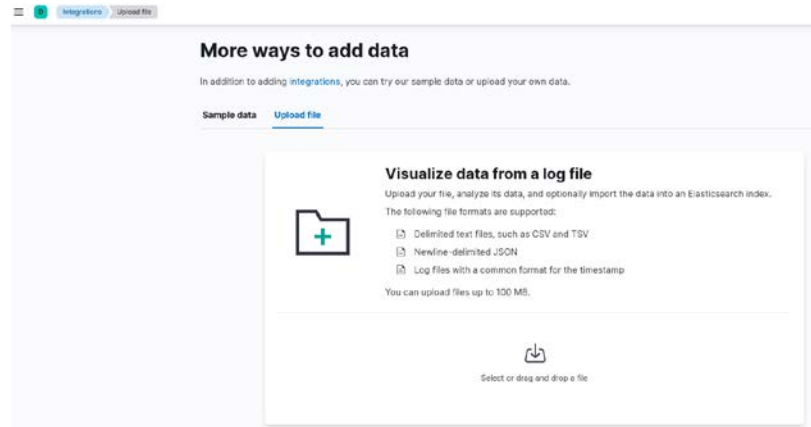
- ElasticSearch asigna un tipo de dato en el proceso de carga
- Si cargamos los datos de movierate: UserId, MovieId, Rate, Timestamp (en el formato ISO 8601) los tipos que se asignan son los siguientes

Series Temporales. Carga de datos

- Utiliza la línea de comandos para cargar los datos de las valoraciones de las películas (fichero *ratings.json*) o la consola principal de Kibana:

```
curl -X PUT "localhost:9200/movierate?pretty"
```

```
curl -X POST -H "Content-Type: application/json" \  
"localhost:9200/movierate/_doc/_bulk?pretty&refresh" \ --data-binary "@./ratings.json"
```



Movierate: tipos de datos

POST movierate/_bulk

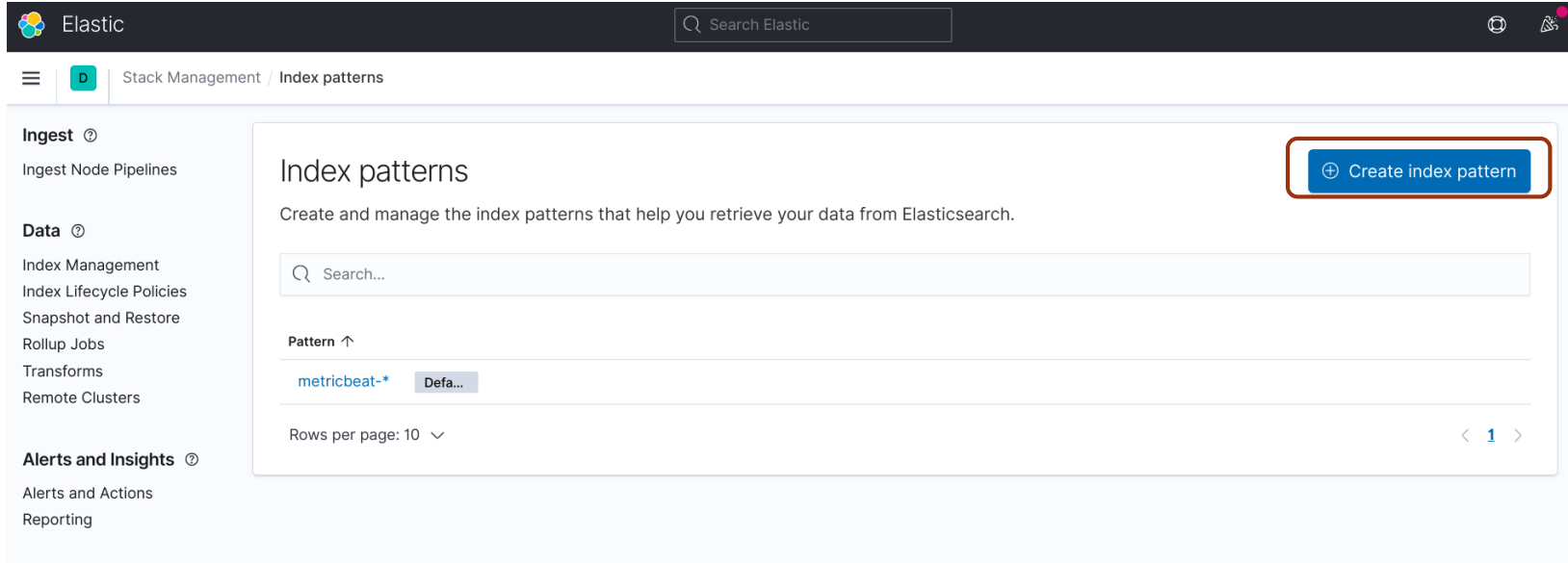
```
{ "index": {} }
{ "userId": 1, "movieId": 147, "rating": 4.5, "fecha": "2015-03-10T00:07:15" }
{ "index": {} }
{ "userId": 1, "movieId": 858, "rating": 5.0, "fecha": "2015-03-09T23:52:03" }
{ "index": {} }
{ "userId": 1, "movieId": 1221, "rating": 5.0, "fecha": "2015-03-09T23:52:26" }
{ "index": {} }
{ "userId": 1, "movieId": 1246, "rating": 5.0, "fecha": "2015-03-09T23:52:36" }
{ "index": {} }
{ "userId": 1, "movieId": 1968, "rating": 4.0, "fecha": "2015-03-10T00:02:28" }
{ "index": {} }
{ "userId": 1, "movieId": 2762, "rating": 4.5, "fecha": "2015-03-09T23:48:20" }
{ "index": {} }
{ "userId": 1, "movieId": 2918, "rating": 5.0, "fecha": "2015-03-09T23:53:13" }
{ "index": {} }
{ "userId": 1, "movieId": 2959, "rating": 4.0, "fecha": "2015-03-09T23:53:21" }
```



```
{
  "movierate" : {
    "aliases" : { },
    "mappings" : {
      "properties" : {
        "fecha" : {
          "type" : "date"
        },
        "movieId" : {
          "type" : "long"
        },
        "rating" : {
          "type" : "float"
        },
        "userId" : {
          "type" : "long"
        }
      }
    },
    "settings" : {
      "index" : {
        "creation_date" : "1574373880247",
        "number_of_shards" : "1",
        "number_of_replicas" : "1",
        "uuid" : "m6d5YiTNRUip8aaU21L1bg",
        "version" : {
          "created" : "7040299"
        },
        "provided_name" : "movierate"
      }
    }
  }
}
```

Movierate: Definiendo un patrón de índice (I)

➤ Seleccionamos Stack Management e Index Patterns



The screenshot shows the Elastic Stack Management interface. The top navigation bar includes the Elastic logo, a search bar, and user profile icons. The left sidebar contains navigation links for Ingest, Data, and Alerts and Insights. The main content area is titled 'Index patterns' and includes a 'Create index pattern' button (highlighted with a red box), a search bar, a 'Pattern' dropdown menu showing 'metricbeat-*' and 'Defa...', and a 'Rows per page' selector set to 10.

➤ Creamos un nuevo patrón en Movierate

Movierate: Definiendo un patrón de índice (II)

- Especificamos movierate y como campo temporal *fecha*
- Creamos el patrón de índice

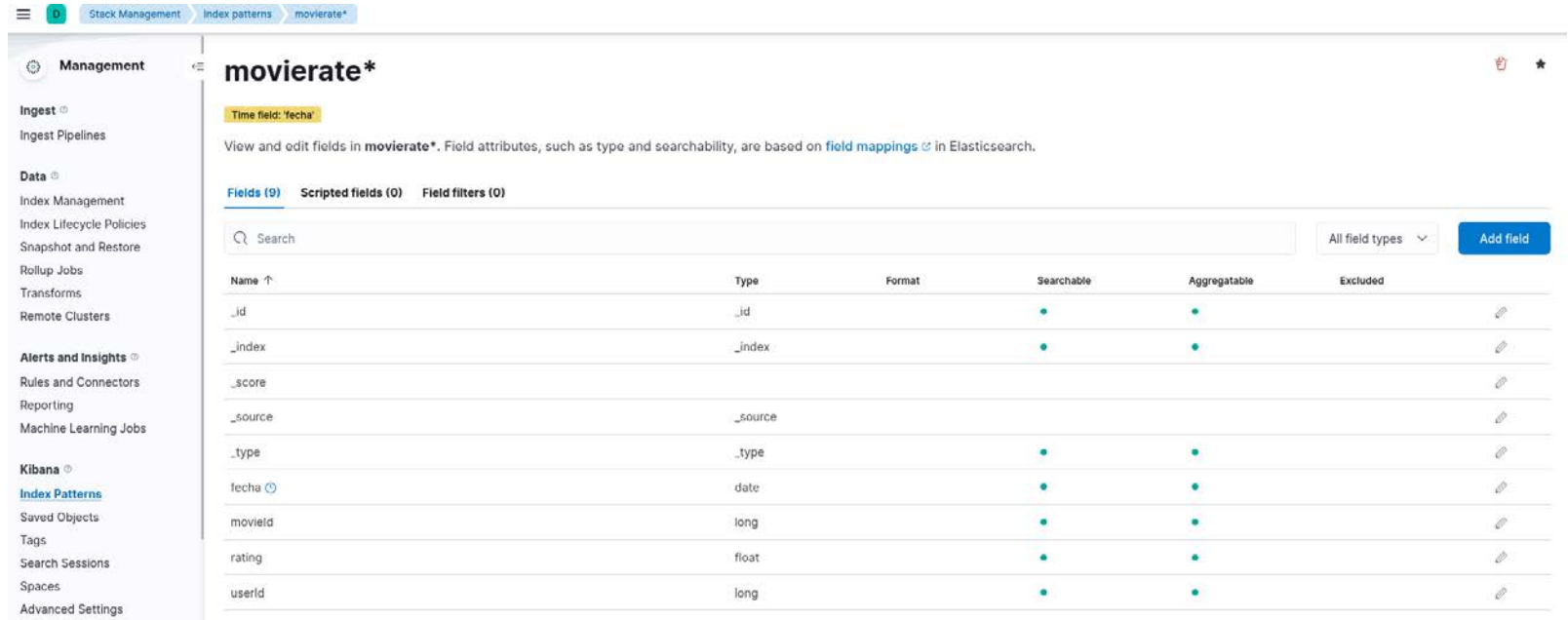
The screenshot displays the Elasticsearch Stack Management interface. The top navigation bar shows 'Stack Management' and 'Index patterns'. The left sidebar contains a 'Management' section with links to 'Ingest' and 'Data', and a 'Data' section with links to 'Index Management', 'Index Lifecycle Policies', 'Snapshot and Restore', 'Rollup Jobs', 'Transforms', and 'Remote Clusters'. The main content area is titled 'Create index pattern' and contains the following fields:

- Name:** A text input field containing 'movierate*'. Below it, a note states: 'Use an asterisk (*) to match multiple characters. Spaces and the characters , / ? " ' < > | are not allowed.'
- Timestamp field:** A dropdown menu with 'fecha' selected. Below it, a note states: 'Select a timestamp field for use with the global time filter.'

Below the form, there is a link 'Show advanced settings'. The right sidebar shows a confirmation message: '✓ Your index pattern matches 1 source.' Below this, the index pattern 'movierate' is displayed with an 'Index' button next to it. At the bottom of the right sidebar, it says 'Rows per page: 10' with a dropdown arrow.

Movierate: Definiendo un patrón de índice (III)

Queda definido ya el patrón de índice



The screenshot shows the Kibana interface for the 'movierate*' index pattern. The left sidebar contains navigation links for Management, Ingest, Data, Alerts and Insights, and Kibana. The main content area displays the 'Fields (9)' tab, showing a table of fields with their attributes.

Time field: 'fecha'

View and edit fields in **movierate***. Field attributes, such as type and searchability, are based on [field mappings](#) in Elasticsearch.

Fields (9) Scripted fields (0) Field filters (0)

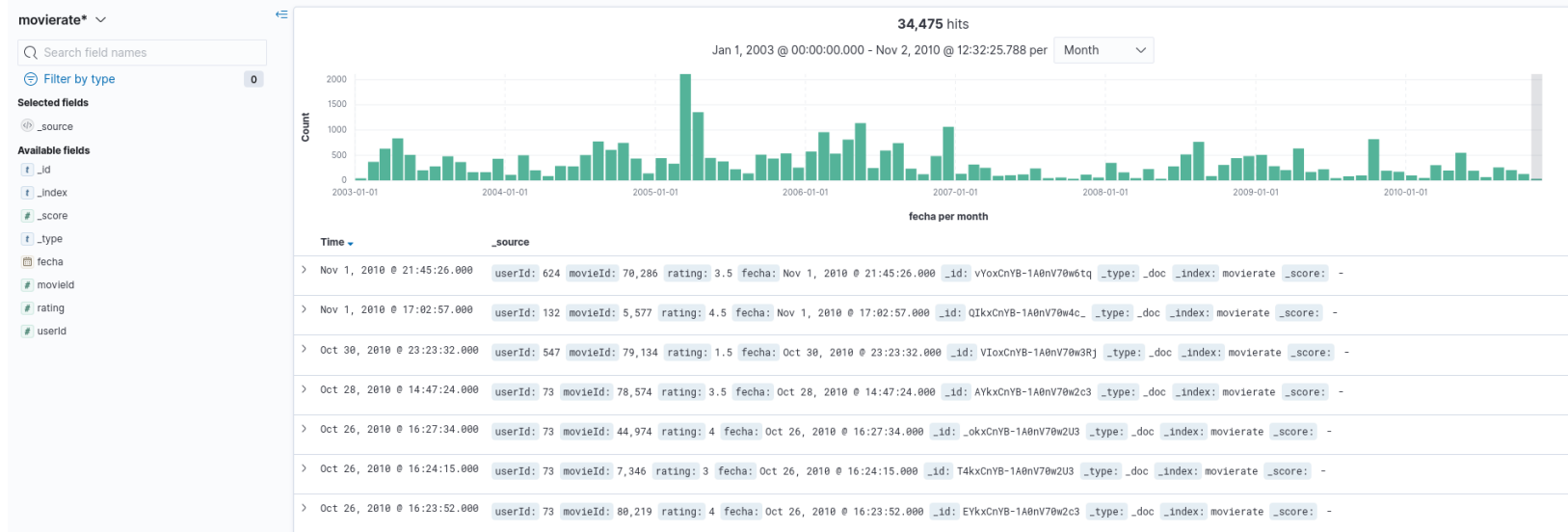
Search

All field types Add field

Name ↑	Type	Format	Searchable	Aggregatable	Excluded
_id	_id		•	•	
_index	_index		•	•	
_score					
_source	_source				
_type	_type		•	•	
fecha	date		•	•	
movielid	long		•	•	
rating	float		•	•	
userid	long		•	•	

Movierate: Definiendo un patrón de índice (V)

- Accede ahora a la opción de Discovery y modificando la intervalo temporal de consulta (año, mes, semanas) visualiza el número de votos ó el promedio de votación de los usuarios...



ElasticSearch. Ingesta de ficheros (I)

- La ingesta de ficheros en ElasticSearch requiere un plugin: Ingest Attachments
- Este plugin se encarga de acceder al contenido del fichero (por ejemplo PPT, XLS, PDF) y para ello utiliza (a través de ese plugin) el framework Apache Tika
- El campo fuente a utilizar dentro del registro a incluir en ElasticSearch tiene que estar en base64

➤ Instalación del plugin en Elasticsearch

Aunque da warnings de seguridad puedes instalarlo sin llegar a utilizar sudo

```
Continue with installation? [y/N]y
```

ElasticSearch. Ingesta de ficheros (II)

- Creamos una pipeline para el Attachment Processor (mecanismo de uso del plugin). Utilizamos la consola de desarrollo de Kibana

```
PUT _ingest/pipeline/attachment
```

```
{  
  "description" : "Ingestar datos de documentos",  
  "processors" : [  
    {  
      "attachment" : {  
        "field" : "data"  
      }  
    }  
  ]  
}
```

ElasticSearch. Ingesta de ficheros (III)

- Creamos nuestro índice para almacenar esos documentos (desde consola de desarrollo de Kibana):

```
PUT uamdocuments
```

- Generamos la cadena en base64 correspondiente al documento a ingestar utilizando el comando base64 (desde terminal en sistema operativo)

```
base64 Documento.txt > Documento_base64.txt
```

- Con ese contenido lo ingestamos en Kibana utilizando la pipeline creada anteriormente

ElasticSearch. Ingesta de ficheros (IV)

- Desde la consola de desarrollo de Kibana:

```
PUT uamdocuments/_doc/1?pipeline=attachment
```

```
{
```

```
  "data": "e1xydGYxXGFuc2kNCkxvcmVtIGlwc3VtIGRvbG9yIHdpdCBhbWV0DQpccGFyIH0="
```

```
}
```

ElasticSearch. Ingesta de ficheros (V)

```
{
  "took" : 558,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "uamdocuments",
        "_type" : "doc",
        "_id" : "2",
        "_score" : 1.0,
        "_source" : {
          "data" : "RXJhbW9zIGNvbW8gZG9zIHJlbW9zIGJvZ2FuZG8gZW4gbGEgbWlzbWEgZGlyZW9uLgo=",
          "attachment" : {
            "content_type" : "text/plain; charset=ISO-8859-1",
            "language" : "es",
            "content" : "Eramos como dos remos bogando en la misma direccion.",
            "content_length" : 54
          }
        }
      }
    ]
  }
}
```


Cerebro. Herramienta para la gestión de ElasticSearch

Imenezes / cerebro Public

<> Code Issues 157 Pull requests 19 Actions Projects Security Insights

main 3 branches 33 tags

Go to file Code

About

README

Cerebro

`docker pull: 13M` `CI build` `ci build`

cerebro is an open source(MIT License) elasticsearch web admin tool built using Scala, Play Framework, AngularJS and Bootstrap.

Requirements

cerebro needs Java 11 or newer to run.

Installation

- Download from <https://github.com/Imenezes/cerebro/releases>
- Extract files
- Run bin/cerebro or bin/cerebro.bat if on Windows
- Access on <http://localhost:9000>

Chocolatey (Windows)

You can install cerebro using Chocolatey:

```
choco install cerebro-es
```

Package creates windows service cerebro. Access on <http://localhost:9000>

Docker

You can find the official docker images in the official [docker hub repo](#).

Visit [cerebro-docker](#) for further information.

Configuration

HTTP server address and port

You can run cerebro listening on a different host and port(defaults to 0.0.0.0:9000):

```
bin/cerebro -Dhttp.port=1234 -Dhttp.address=127.0.0.1
```

Miguel Olivares Merge pull request #518 from mortenhaberg/make-text-selectable ccb6044 on 3 Jul 2021 430 commits

.github/workflows	Create scala.yml	13 months ago
app	fix scala 3 deprecation warnings	13 months ago
conf	replace jscs with eslint	13 months ago
examples	Example of configuration of open ldap with group check	17 months ago
project	update sbt plugins	13 months ago
public	Make text selectable	7 months ago
src	simplify clipboard service	13 months ago
test	fix scala 3 deprecation warnings	13 months ago
tests	flag for running karma tests continuously	13 months ago
.gitignore	replace jscs with eslint	13 months ago
.travis.yml	update travis ci config	13 months ago
CHANGES.md	version bump 0.9.4	10 months ago
CONTRIBUTING.md	update contributing.md	5 years ago
Gruntfile.js	flag for running karma tests continuously	13 months ago
LICENSE	add license	4 years ago
README.md	Update README.md	13 months ago
build.sbt	version bump 0.9.4	10 months ago
package-lock.json	version bump 0.9.4	10 months ago
package.json	version bump 0.9.4	10 months ago

README.md

No description or website provided.

[elasticsearch](#) [admin](#)

Readme

MIT License

4.8k stars

198 watching

663 forks

Releases 33

v0.9.4 Latest on 10 Apr 2021

+ 32 releases

Packages

No packages published

Contributors 28

+ 17 contributors

Languages

JavaScript 80.4%

Scala 15.0%

HTML 3.5%

CSS 0.5%

UNIVERSIDAD AUTÓNOMA DE SAN PABLO DE LOS RÍOS

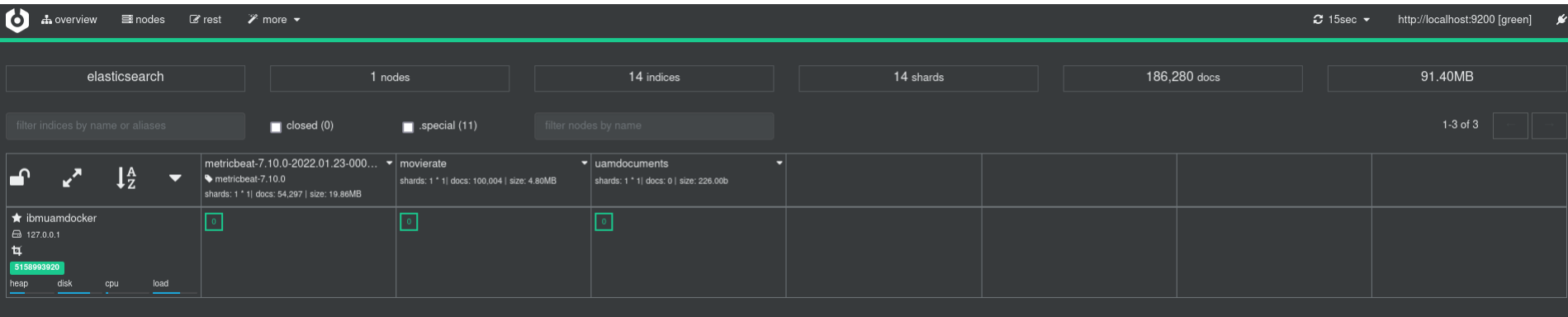
Máster en Big Data y Data Science

Ciclo de Vida Analítico del Dato

32

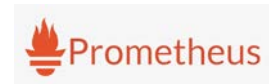
Cerebro

- Permite conocer el estado del cluster, índices, configuración, etc.

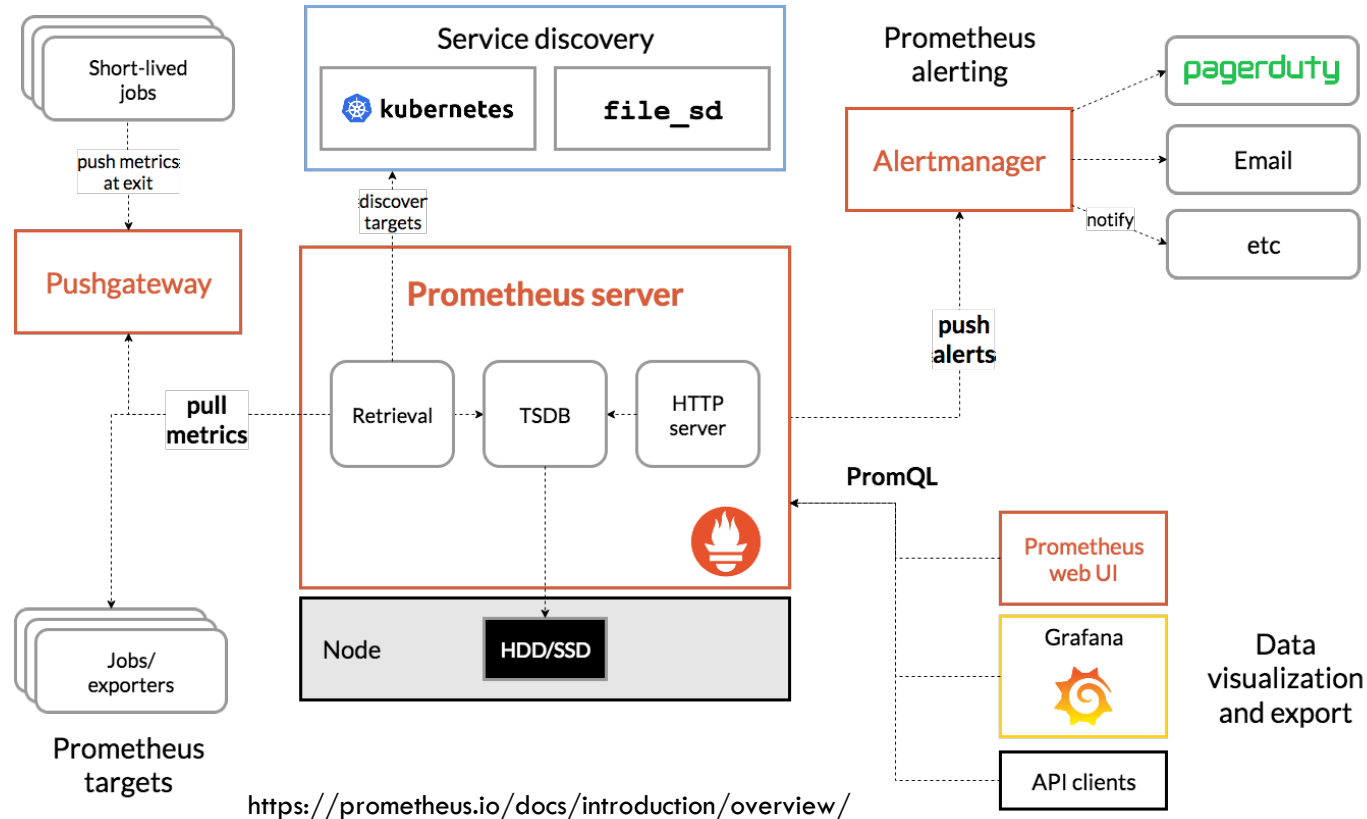


Otros componentes del Ecosistema de Buscadores

- El entorno de monitorización, visualización y métricas está cambiando a una gran velocidad. Por ejemplo, en la actualidad están, entre otros:
 - Grafana: Permite visualizar, consultar, emitir alertas y analizar las métricas independientemente de dónde estén almacenadas
 - Prometheus: Permite monitorizar y emitir alertas. Tiene un modelo de datos propio especialmente preparado para registrar series temporales tanto para entornos independientes (sistemas) como entornos de microservicios (arquitecturas de múltiples sistemas)



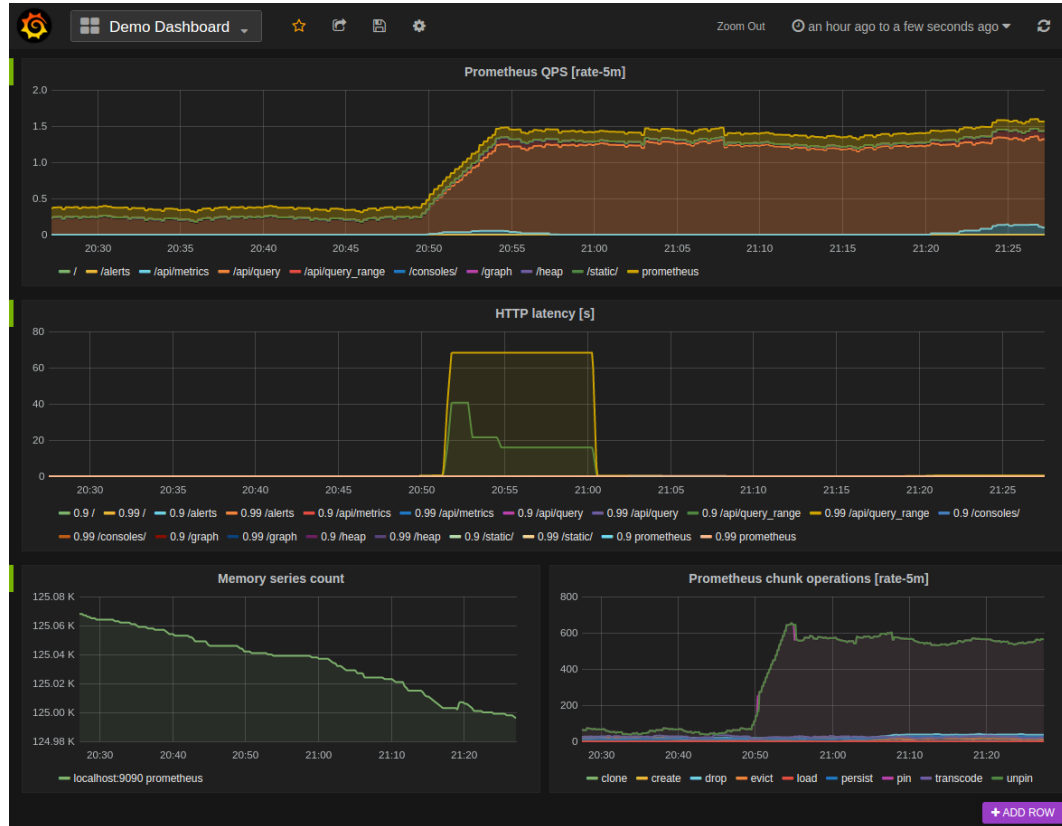
Prometheus y Grafana



Grafana



Grafana consultando Prometheus



Valoraciones. Kibana versus Grafana

Visualización

- Kibana. Su integración con Elasticsearch hace que si trabajamos con logs y deseamos buscar en ellos es una de las mejores decisiones
- Grafana. Tiene muy buena visualización de métricas integradas con la herramienta

Fuentes

- Kibana. La información deberá almacenada en Elasticsearch
- Grafana. Permite acceder a datos en múltiples fuentes para las que tiene conectores, incluida Elasticsearch

Consultas

- Kibana. Las capacidades de consulta son muy potentes al delegar en Lucene
- Grafana. Tiene un editor de consulta que depende de la fuente donde se esté almacenando la información

Alertas

- Kibana. Las alertas dependen de plugins o paquetes adicionales
- Grafana. La capacidad de configurar y lanzar alertas es muy alta