

# Introduction à la Suite Elastic (Elasticsearch & Kibana)

Maîtrisez les bases d'Elasticsearch et Kibana via elastic cloud

# INDEX

1

## Présentation

- Présentation de la suite ELK
- Cas d'usage et applications
- Points forts/faibles

2

## Installation/Exploration

- Installation et configuration (Login/access elastic cloud)
- Exploration d'Elasticsearch
- Introduction à Kibana

3

## Ateliers pratiques

- Exercices
- Dataviz

# 01 Présentation de la Suite Elastic

1

« Qu'est-ce que la suite ELK ? »

# 1 Qu'est-ce que la Suite Elastic?

La Suite Elastic est un ensemble **d'outils** et de **solutions logiciels** développés par Elastic NV pour :

- la recherche,
- l'observation,
- la sécurité et
- l'analyse des données.

**Définition [moteur de recherche]** [https://fr.wikipedia.org/wiki/Moteur\\_de\\_recherche](https://fr.wikipedia.org/wiki/Moteur_de_recherche) = Un moteur de recherche est une application permettant à un utilisateur d'effectuer une **recherche locale** ou en **ligne**.

**Définition [Lucene]** <https://fr.wikipedia.org/wiki/Lucene> = Lucene est une **bibliothèque** open source écrite en Java qui permet **d'indexer** et de chercher du texte. Il est utilisé dans certains moteurs de recherche.

2

« A quoi sert la suite ELK ? »

## 2 A quoi sert la suite ELK ?

Elle permet :

1. le **stockage**,
2. l'**analyse**, et
3. la **visualisation**

de grandes quantités de données en temps réel.



# Elastic Stack

100% Open Source

3

« De quoi est composé la suite ELK ? »



### 3 De quoi est composé la suite ELK ?

Voici les principaux composants :



Quels sont les rôles et les fonctionnalités  
de ces principaux composants ?

### 3 De quoi est composé la suite ELK ?

Voici les principaux composants :



#### Collect & Ship Logs

**Beats** collecte et transmet.

C'est léger et spécialisé pour un type de données (fichiers logs, métriques, etc.).

#### Parse Logs

**Logstash** collecte, transforme, et transmet.

C'est un outil complet pour des pipelines de traitement de données plus complexes.

#### Store & Search Logs

**Elasticsearch** est le cœur de la suite.

Il permet de stocker et d'interroger des données structurées et non structurées.

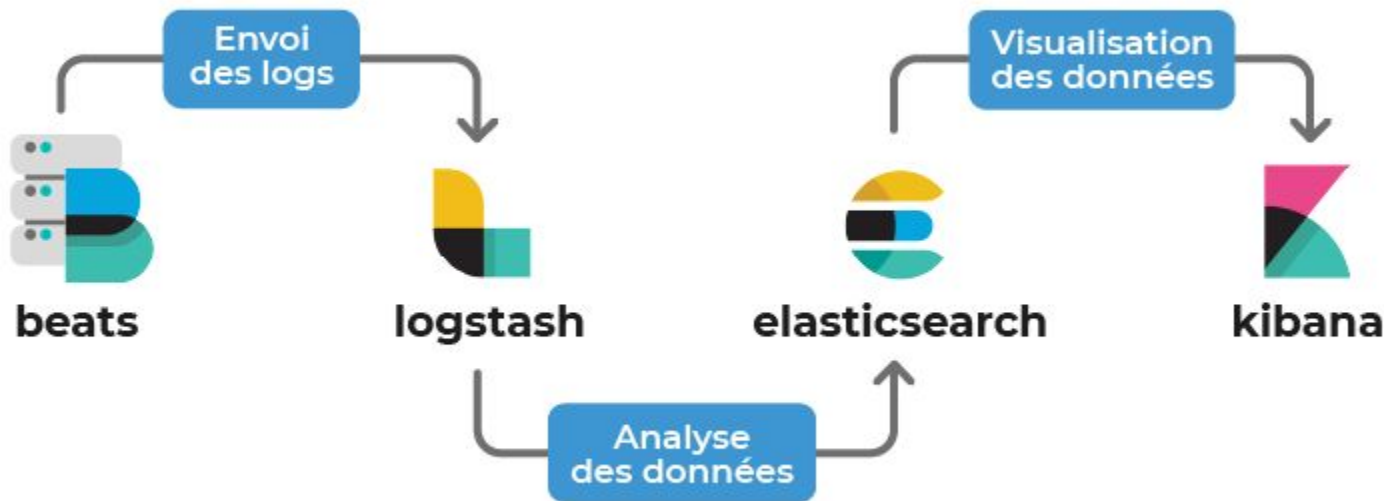
#### Visualize Logs

**Kibana** est une interface utilisateur.

C'est un outil pour visualiser et naviguer dans les données stockées dans Elasticsearch.

### 3 De quoi est composé la suite ELK ?

En résumé :



*\*la majorité du workshop se concentrera sur **Elasticsearch et Kibana** via **Elastic Cloud***

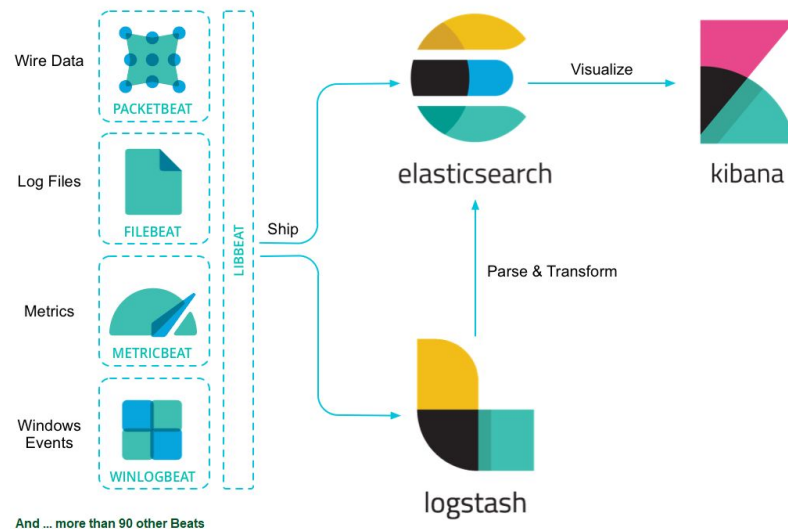
4

« Quelques détails sur  
le premier composant “Beats” »

## 4 Quelques détails sur le premier composant “Beats”

Beats est une plate-forme regroupant des **solutions légères de transfert** qui envoient des données provenant de toutes les machines vers Logstash ou Elasticsearch.

Il existe plus de 93 agents Beats, parmi les plus connus :



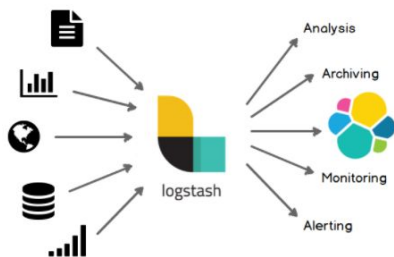
5

« Quelques détails sur  
le second composant “Logstash” »

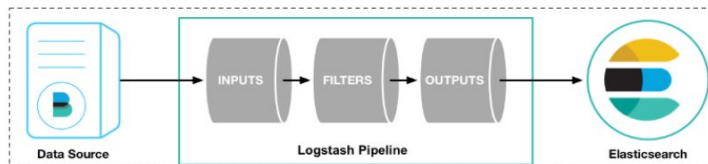
# 5

## Quelques détails sur le second composant “Logstash”

**Logstash** (collecter et transformer)



- Parser
- Filtrer
- Envoyer vers Elasticsearch

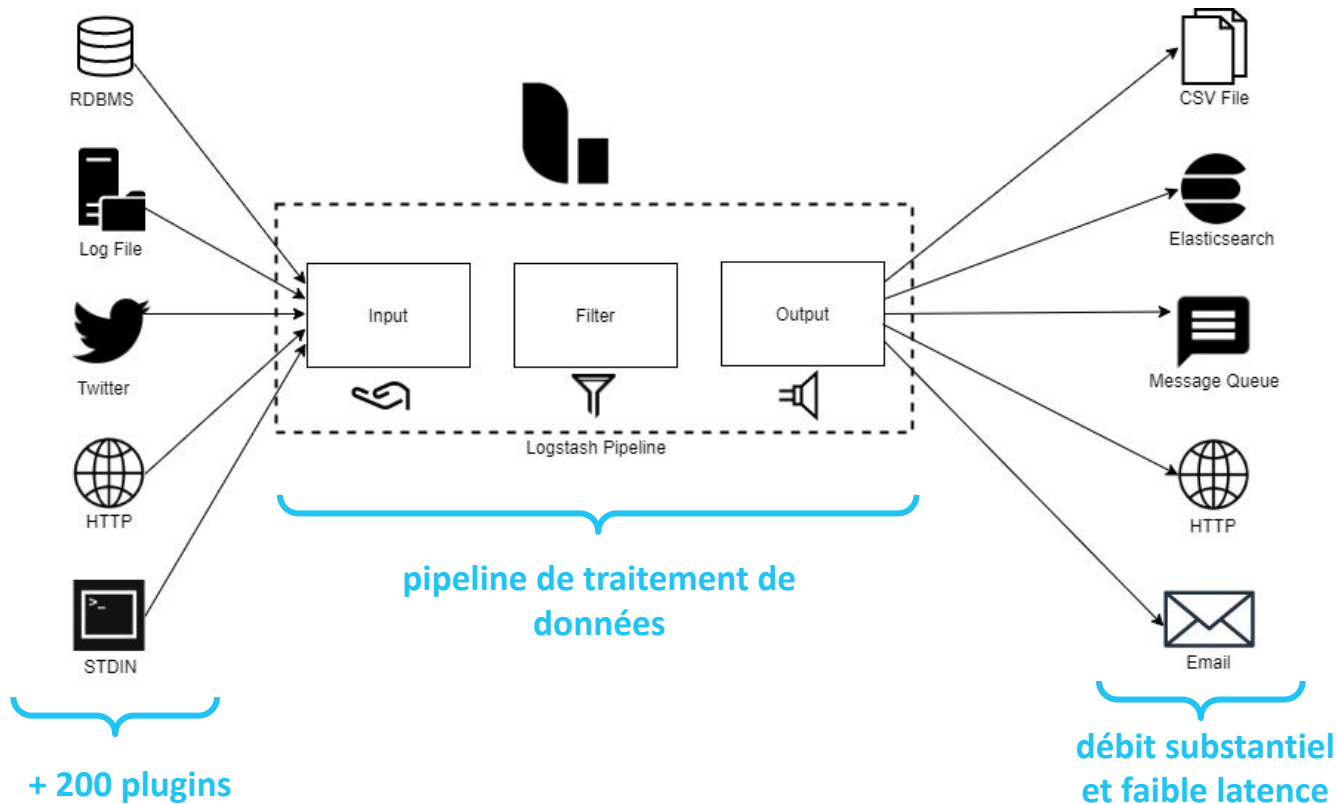


**Son fichier de configuration est composé de 3 parties :**

- Entrées : données de toutes formes, tailles et sources
- Filtres : via une bibliothèque de filtres très riche (analyse et transformation des données)
- Sorties : choisir l'entrepôt (Elasticsearch) et envoyer les données

# 5

## Quelques détails sur le second composant “Logstash”





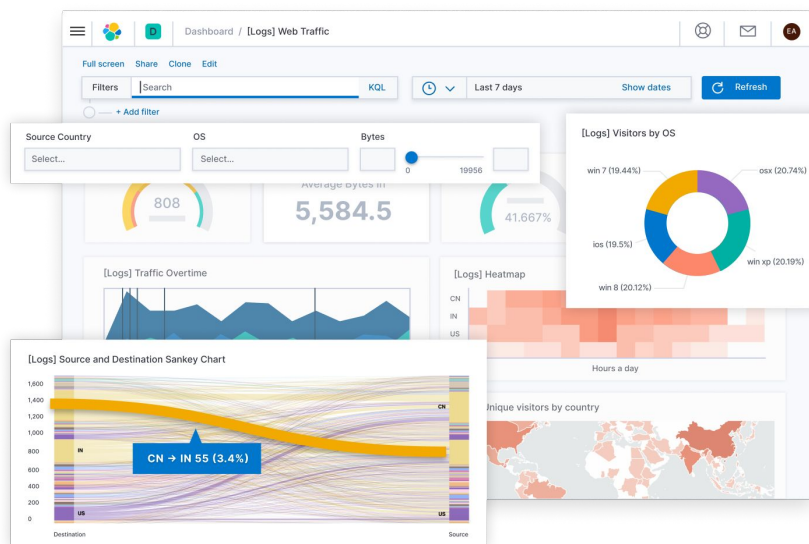
6

« Quelques détails sur  
le quatrième composant “Kibana” »

## 6 Quelques détails sur le quatrième composant “Kibana”

Kibana est une interface de visualisation et de gestion permettant de **consulter** et **d'explorer** via des **dashboards interactifs, paramétrables** et entièrement personnalisables, les données stockées dans Elasticsearch, via :

- des graphiques,
- histogramme,
- carte géo.. etc.

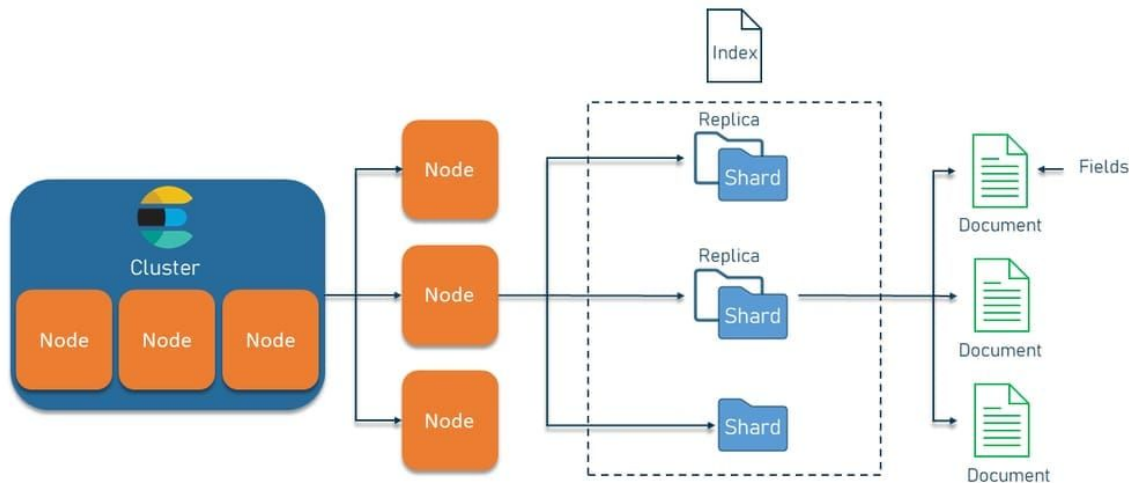


7

« Quelques détails sur  
le troisième composant “Elasticsearch” »

## 7 Quelques détails sur le troisième composant “Elasticsearch”

Architecture de la base de données et ces principaux composants :



Quels sont les rôles de chaque composant  
(cluster, node, shard, replica, index, document) ?

## 7 Architecture de la base de données et ses principaux composants

Composant	Description	Utilité
<b>Cluster</b>	Regroupe plusieurs noeuds (exemple : un groupe de 3 serveurs)	Gère les données et répartit le travail entre les noeuds.
<b>Noeuds</b>	Instances Elasticsearch dans le cluster (ie. “un programme Elasticsearch qui tourne”).	Partagent les tâches de recherche et de stockage.
<b>Shards</b>	Fragments d’un index.	Divisent les données pour une gestion plus rapide et distribuée. Chaque shard peut être traité par un nœud différent.
<b>Replica</b>	Copies des shards.	Assurent la haute disponibilité et tolèrent les pannes.
<b>Index</b>	Regroupe les shards principaux et répliques (exemple : une table en SQL).	Organise les données pour faciliter la recherche. Les données sont indexées à l’aide de structures comme les <b>arbres inversés</b> pour accélérer les recherches.
<b>Document</b>	Unités élémentaires de données structurées (exemple : une ligne en SQL).	Contiennent les informations que vous analysez ou consultez.

## 7 La structure de base des index : index inversé

Elasticsearch utilise une structure appelée **arbre inversé** pour **organiser** et **rechercher** rapidement les données. Explication pas à pas :

### 1. Concept de l'arbre inversé

Un arbre inversé est une structure qui associe les **termes** d'un document à leur **emplacement**.

Contraire d'un index classique :

- Dans un index classique (comme une table SQL), on a :  
Document → Mots clés qu'il contient.
- Dans un arbre inversé, c'est l'inverse :  
Mot clé → Liste des documents où il apparaît.

# 7 La structure de base des index : index inversé

## 2. Étapes de construction dans Elasticsearch

Prenons un exemple avec trois documents contenant des mots :

Document 1 : "Chat noir dort"

Document 2 : "Le chien noir aboie"

Document 3 : "Chat et chien jouent"

### a. Analyse des textes : Les phrases sont découpées en termes (ou tokens).

Doc 1 : ["chat", "noir", "dort"]

Doc 2 : ["chien", "noir", "aboie"]

Doc 3 : ["chat", "chien", "jouent"]

## 7 La structure de base des index : index inversé

### 3. Création de l'arbre inversé :

L'arbre inversé associe chaque mot clé à une liste de documents.

Mot clé	Documents
chat	Doc 1, Doc 3
noir	Doc 1, Doc 2
dort	Doc 1
chien	Doc 2, Doc 3
aboie	Doc 2
jouent	Doc 3



## 7 La structure de base des index : index inversé

### 4. Comment ça accélère les recherches ?

- Si vous cherchez le mot "chat", Elasticsearch consulte directement l'arbre inversé et retourne les documents 1 et 3, sans avoir à parcourir chaque document.
- En combinant plusieurs mots clés (par ex., "chat" + "noir"), Elasticsearch peut croiser les listes pour ne retourner que les documents pertinents (Doc 1 ici).



### Avantages :

⇒ **Recherche rapide** : l'arbre inversé évite de lire tous les documents pour trouver un mot.

⇒ **Scalabilité** : permet de gérer des milliards de documents en répartissant les arbres sur plusieurs shards.



*Any questions?*

Est-ce que vous avez des questions sur l'architecture et ses composants ?



8

« Aller plus loin avec le moteur de recherche “Elasticsearch” »

## 8 Introduction au scoring TF-IDF

Elasticsearch ne se contente pas de stocker et d'indexer les données, il offre également des **fonctionnalités avancées** pour rechercher et analyser les informations de **manière intelligente**.

L'un de ses principaux atouts est la capacité de calculer *la pertinence d'un document par rapport à une requête donnée*. Ce calcul repose sur des modèles statistiques, dont le célèbre scoring **TF-IDF** utilisé pour mesurer la similarité entre les documents et les termes de recherche.

- **TF (Term Frequency)** : Fréquence d'un terme dans un document.  
⇒ Plus un terme apparaît souvent, plus il est important dans ce document.
- **IDF (Inverse Document Frequency)** : Mesure l'importance d'un terme à travers l'ensemble du corpus.  
⇒ Les termes rares sont considérés comme plus significatifs que les termes fréquents.

- La formule : 
$$\text{TF-IDF} = \text{TF} \times \text{IDF}$$

\* possible de l'implémenter avec des bibliothèques **Python** 

## 8 Introduction au scoring TF-IDF : un exemple

**Exemple :** je cherche le document dont le contenu est le plus similaire à la requête.

Document 1 : "Le chat noir dort"

Document 2 : "Le chien noir aboie"

La requête : "chat noir"

### Schéma illustrant l'interaction entre TF et IDF :

Document	Terms	Metric	Value	Metric	Value
Document 1	chat, noir, dort	TF for 'chat'	TF = 0.33	IDF for 'chat'	IDF = $\log(3/2) = 0.176$
Document 2	chien, noir, aboie	TF for 'chat'	TF = 0.0	IDF for 'chat'	IDF = $\log(3/2) = 0.176$
Document 3	chat, chien, jouent	TF for 'chat'	TF = 0.33	IDF for 'chat'	IDF = $\log(3/2) = 0.176$

### Calcul du score pour "chat" dans Document 1 :

$$\text{TF-IDF} = 0.33 \times 0.176 = 0.058$$

⇒ Ce processus est fait pour chaque terme dans chaque document.

# 8

## Introduction au scoring TF-IDF : elasticsearch logs

Décomposition du **score** d'un document à l'aide de l'option `"explain": true`

```
GET https://localhost:9200/training_set/_search?pretty

{
  "multi_match": {
    "query": "Global economic implications of the Russia-Ukraine war",
    "fields": ["DocTitle", "DocContent"]
  }
}
```

```
{
  "value": 53.718456,
  "description": "sum of:",
  "details": [
    {
      "value": 6.169303,
      "description": "weight(DocTitle:global in 0) [PerFieldSimilarity], result of:",
      "details": [
        {
          "value": 6.169303,
          "description": "score(freq=1.0), computed as boost * idf * tf from:",
          "details": [
            {
              "value": 2.2,
              "description": "boost",
              "details": []
            },
            {
              "value": 5.054971,
              "description": "idf, computed as log(1 + (N - n + 0.5) / (n + 0.5)) from:",
              "details": [
                {
                  "value": 0.5547467,
                  "description": "tf, computed as freq / (freq + k1 * (1 - b + b * dl / avgdl)) from:",
                  "details": [
                    {
                      "value": 14.324808,
                      "description": "avgdl, average length of field."
                    }
                  ]
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

Calcul du score pour le terme "global" :

- idf = 5.054971
- tf = 0.5547467
- boost = 2.2

$$\Rightarrow \text{score} = \text{boost} * \text{idf} * \text{tf} = 6.169303$$

Zoom de IDF

```
{
  "value": 5.054971,
  "description": "idf, computed as log(1 + (N - n + 0.5) / (n + 0.5)) from:",
  "details": [
    {
      "value": 2,
      "description": "n, number of documents containing term",
      "details": []
    },
    {
      "value": 391,
      "description": "N, total number of documents with field",
      "details": []
    }
  ]
}
```

Zoom de TF

```
{
  "value": 0.5547467,
  "description": "tf, computed as freq / (freq + k1 + (1 - b + b * dl / avgdl)) from:",
  "details": [
    {
      "value": 1.0,
      "description": "freq, occurrences of term within document",
      "details": []
    },
    {
      "value": 1.2,
      "description": "k1, term saturation parameter",
      "details": []
    },
    {
      "value": 0.75,
      "description": "b, length normalization parameter",
      "details": []
    },
    {
      "value": 0.0,
      "description": "dl, length of field",
      "details": []
    },
    {
      "value": 14.324808,
      "description": "avgdl, average length of field."
    }
  ]
}
```

## 8 Autres atouts d'Elasticsearch : d'autres fonctionnalités

- **Analyseur de texte** (**analyzer**) : utilisation des analyseurs, spécifique par **langue**, pour découper le texte en termes (tokens), les transformer (minuscule, suppression des stop words, etc.) et les indexer.
- **Requêtes de type booléen** (**must**, **must not**, **should**, **filter**) : permet de combiner plusieurs conditions de recherche en utilisant les opérateurs booléens.
- **Scalabilité et performance** (**number\_of\_shards** et **number\_of\_replicas**) : possibilité de créer un index avec une configuration optimisé en ajustant le nombre de shards ou de réplicas souhaité.
- **Recherche multi-critères** (**match**, **multi\_match**, **geo\_distance**, **range**) : permet de rechercher simultanément des données textuelles, numériques, géospatiales et d'utiliser plusieurs types de requêtes imbriquées (comme des filtres avec range).
- **Requête approximative** (**fuzzy**) : possibilité d'effectuer des recherches tolérantes aux fautes de frappe, basée sur une distance d'édition (levenshtein distance).



# 02

## Cas d'usage et applications

La Suite Elastic est utilisée dans une variété de domaines grâce à sa flexibilité et sa puissance.

## Cas d'utilisation typiques :

1. Recherche et Analyse de Données :



**Amélioration des expériences de recherche** sur les applications ou les sites web.

1. Analyse Commerciale :



**Analyse des données commerciales** pour prendre des décisions éclairées.

1. Analyse des Journaux et des Événements :



**Agrégation et analyse des journaux** pour obtenir des insights précieux.

1. Analyse de Sécurité :



**Détection des menaces et sécurisation** des environnements informatiques.

1. Surveillance des Performances :

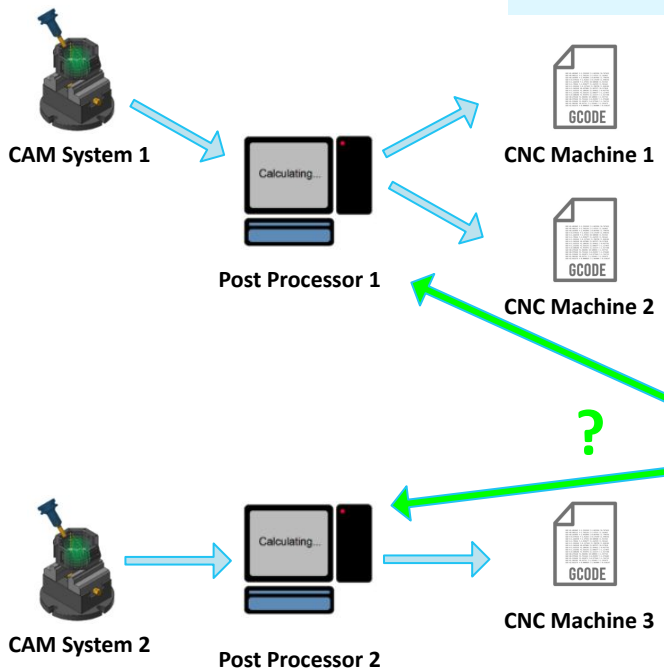


**Surveillance en temps réel** des systèmes et applications.

# Cas d'usage HUPI

Optimisation du processus de mise en service des machines-outils

*Scoring similarity*



What is the most similar CNC file ?

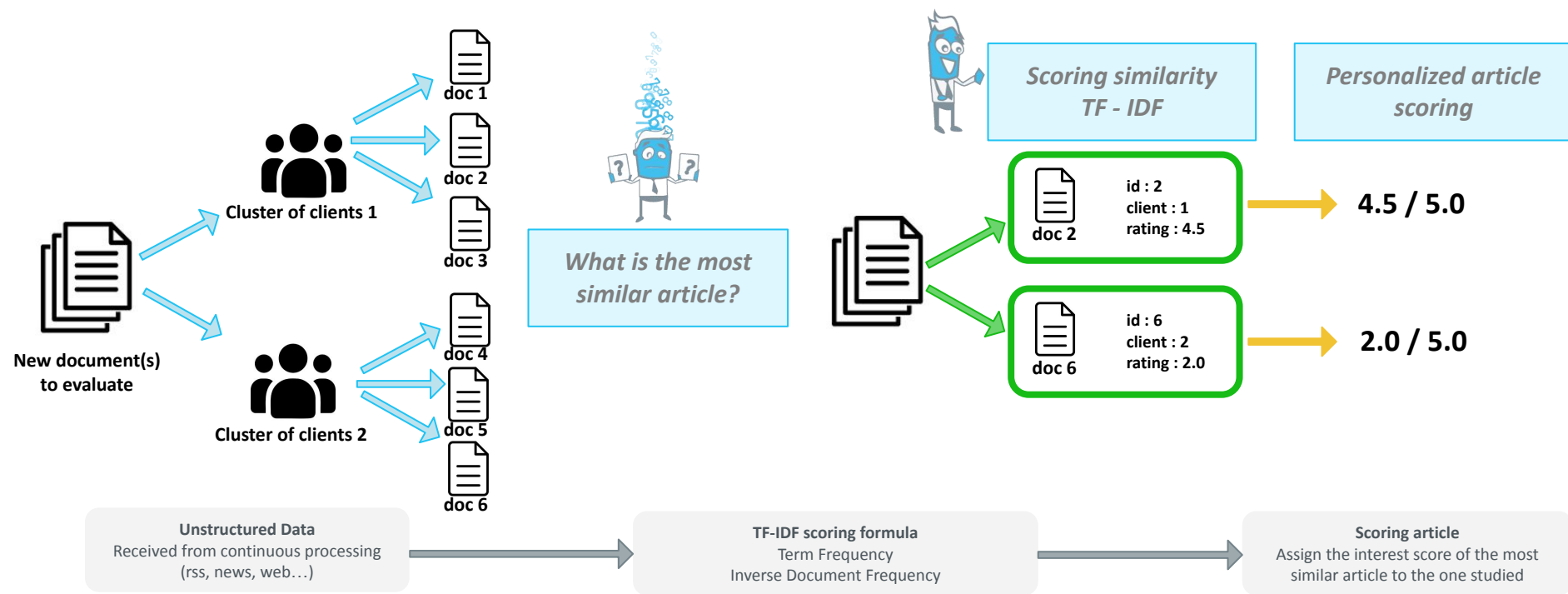


*Scoring TF IDF*  
 "Term Frequency Iverse Document Frequency"

	PUT : 30 Test : 5 File : 3	<b>38</b>
CNC Machine 1		
	Test : 10 Try : 7 glue : 5	<b>22</b>
CNC Machine 2		
	OK : 16 no : 11	<b>27</b>
CNC Machine 3		



Prédiction du “niveau d’intérêt”  
de documents personnalisé par clients



# 03

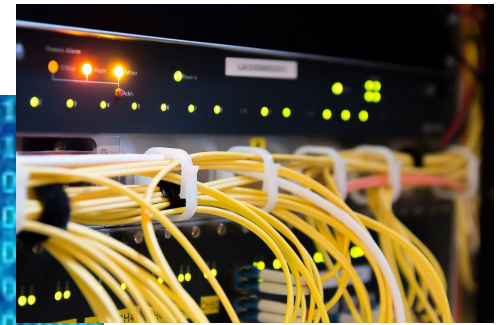
## Points forts et points faibles

“Elasticsearch est un moteur de recherche et d'analyse **rapide, évolutif et flexible**, capable de traiter de grandes quantités de données en temps réel, avec des fonctionnalités avancées propices à l'intégration de **solutions IA et Machine Learning.**”

s'adapte à  
divers types  
de données



offre des  
visualisations  
interactives



"Elasticsearch, bien qu'efficace et puissant, peut être **complexe à maîtriser** et nécessite des **ressources système importantes** pour garantir des performances optimales."

# Comparaison entre Elasticsearch et Python (TF-IDF)

Elasticsearch et Python offrent des approches **distinctes** pour le traitement de texte :

Critères	Elasticsearch	Python (TF-IDF)
Vitesse de recherche	<b>Architecture distribuée</b> pour des recherches rapides, même sur de grands volumes.	Plus <b>lent</b> , en particulier sur des datasets volumineux.
Scalabilité	<b>Scalabilité</b> horizontale pour gérer la croissance des données.	<b>Scalabilité limitée</b> ; nécessite des ressources supplémentaires.
Facilité d'intégration	S'intègre parfaitement avec la stack ELK pour une analyse complète.	Demande des efforts supplémentaires pour une solution intégrée.
Fonctionnalités AI	Intègre des outils avancés pour l'analyse et l'intelligence artificielle, mais nécessite une <b>maîtrise approfondie</b> d'Elasticsearch.	Permet des <b>analyses complexes</b> , potentiellement avec des bibliothèques tierces et expertise.

**Conclusions :** quand choisir

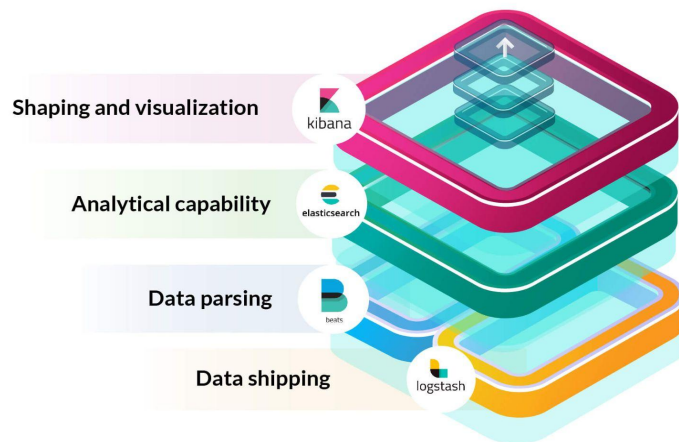
⇒ **Elasticsearch** : pour des recherches **rapides et scalables** sur de grandes quantités de données textuelles.

⇒ **Python (TF-IDF)** : pour des analyses **personnalisées/complexes** qui nécessitent une flexibilité et un contrôle approfondi.



# 04 Installation et configuration

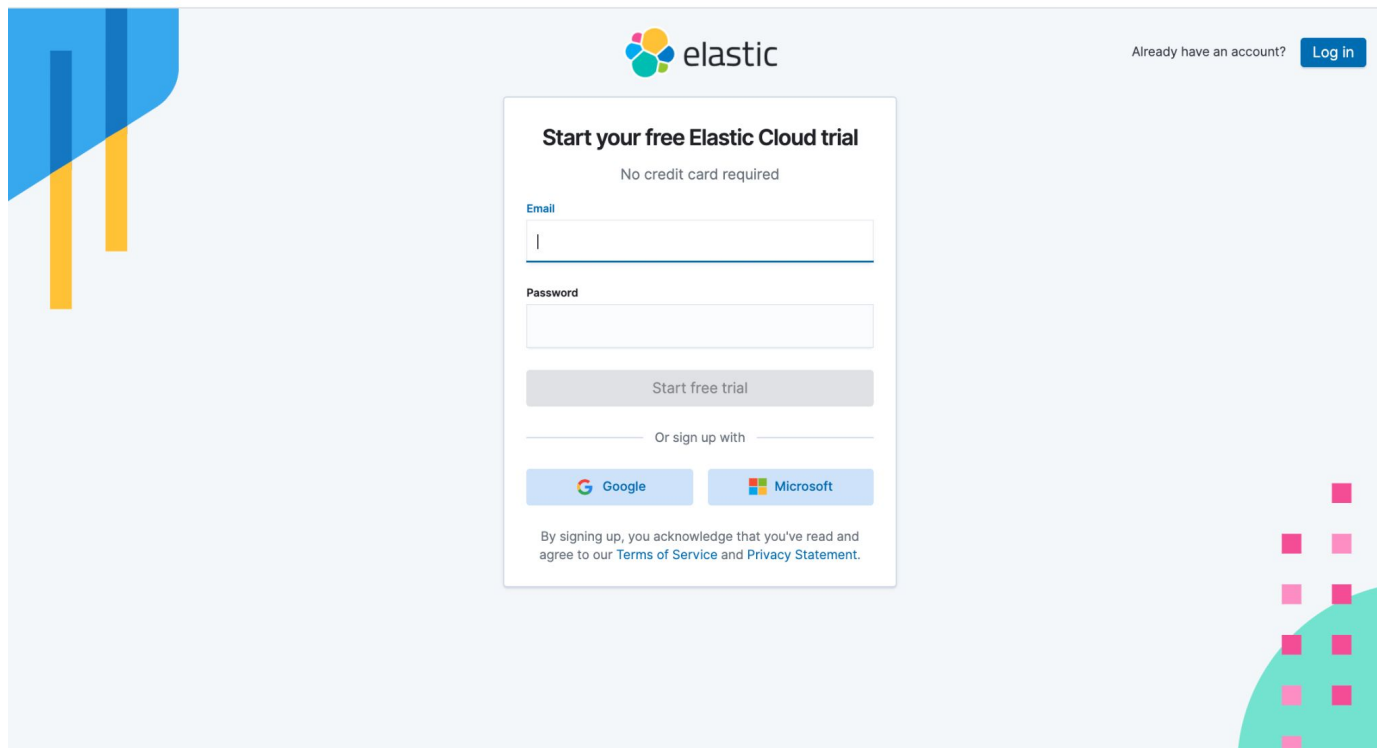
**Elastic Cloud** est un ensemble d'offres SaaS qui fournissent des services Elasticsearch et Kibana hébergés.



- ➔ **Avantages** : facilité de configuration, mises à jour automatiques, de sauvegardes et de la liberté de créer un déploiement.
- ➔ **Sécurité** : conçu avec des mesures de sécurité pour protéger les données.
- ➔ **Intégrations**: intégration facile de services/plateformes (AWS, Google Cloud et Microsoft Azure).

# Création d'un compte sur Elastic Cloud

La période d'essai sur Elastic Cloud dure **14 jours** pour un compte rattaché à une adresse mail.



The image shows the Elastic Cloud trial sign-up page. It features the Elastic logo at the top center. To the right of the logo is a link 'Already have an account?' followed by a 'Log in' button. The main content is a form titled 'Start your free Elastic Cloud trial' with the subtext 'No credit card required'. The form contains an 'Email' field, a 'Password' field, and a 'Start free trial' button. Below the button is a section 'Or sign up with' with 'Google' and 'Microsoft' options. At the bottom, there is a disclaimer: 'By signing up, you acknowledge that you've read and agree to our Terms of Service and Privacy Statement.'

elastic

Already have an account? [Log in](#)

### Start your free Elastic Cloud trial



No credit card required

Email

Password

Start free trial

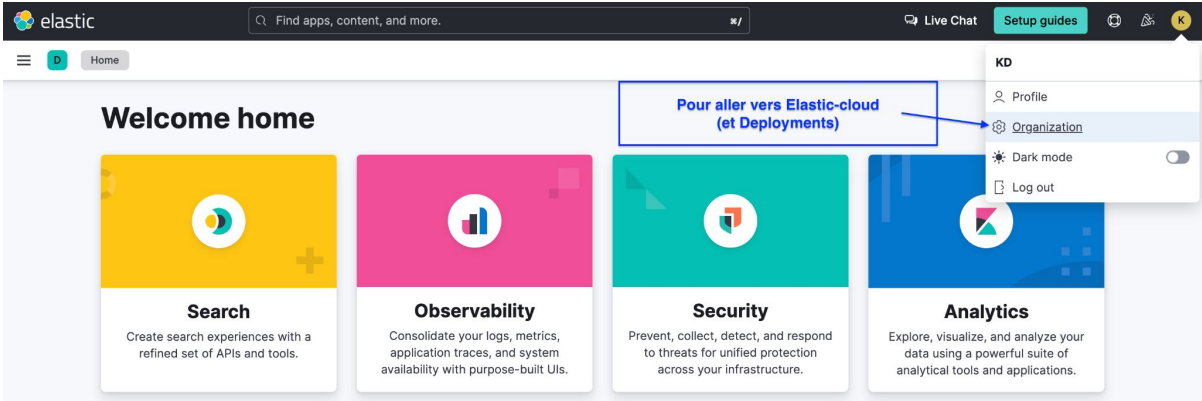
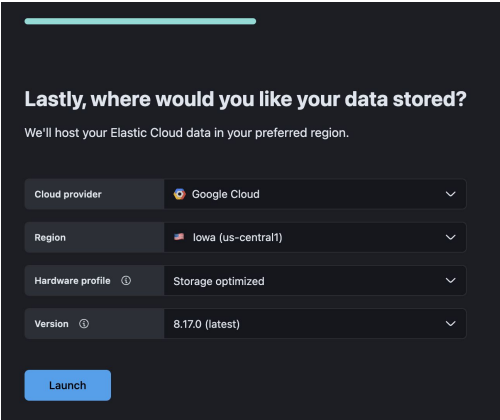
Or sign up with

 Google  Microsoft

By signing up, you acknowledge that you've read and agree to our [Terms of Service](#) and [Privacy Statement](#).

# Configuration de la base de données sur Elastic Cloud

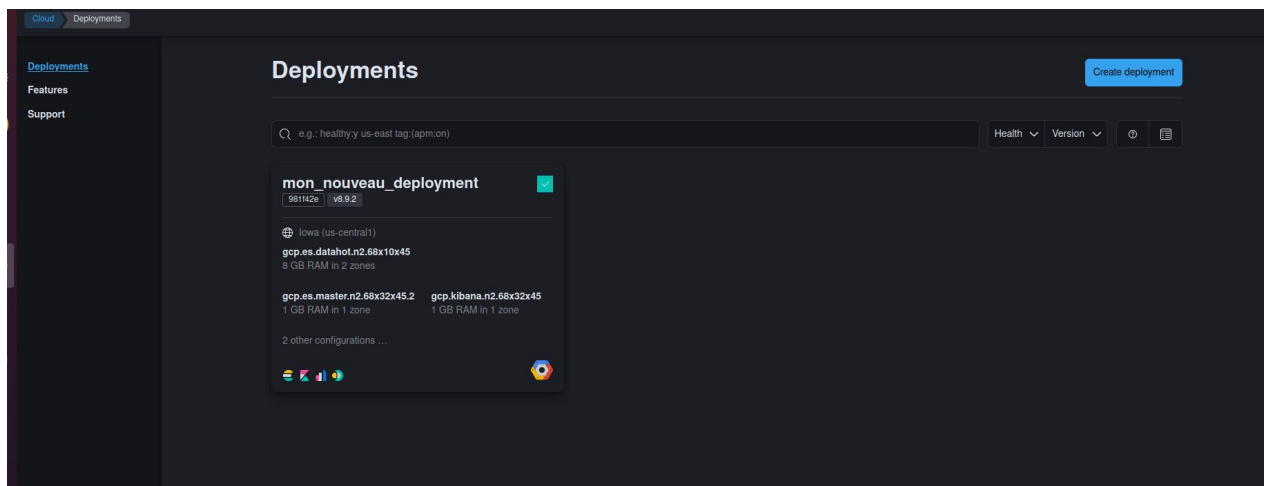
1. Créer un environnement avec un serveur dédié
  - a. Cliquer sur **“Create a deployment”**
  - b. Choisir les configurations
1. Lancer le déploiement
1. Sauvegarder vos identifiants
1. Pour accéder à la page **Deployment**



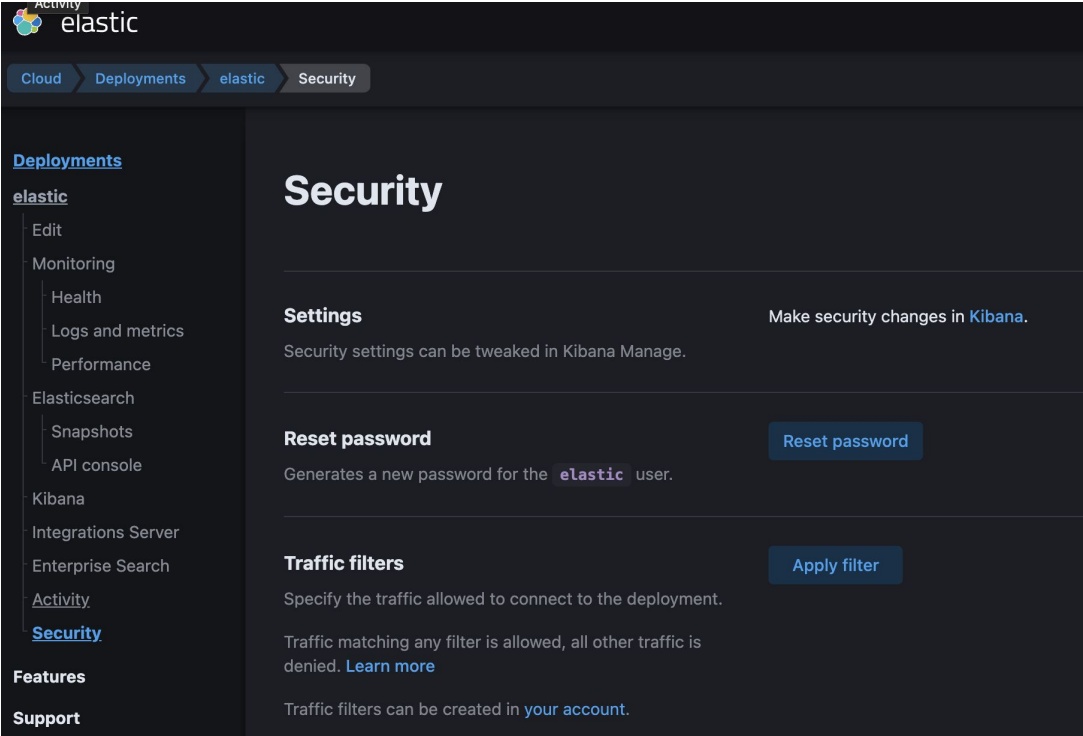
**Déploiement :** Il s'agit de l'ensemble des **ressources provisionnées** pour héberger vos **services Elastic**, y compris :

- au moins un cluster Elasticsearch,
- ainsi que des instances de Kibana,
- APM,
- et d'autres services selon vos besoins.

La page **Deployment** :

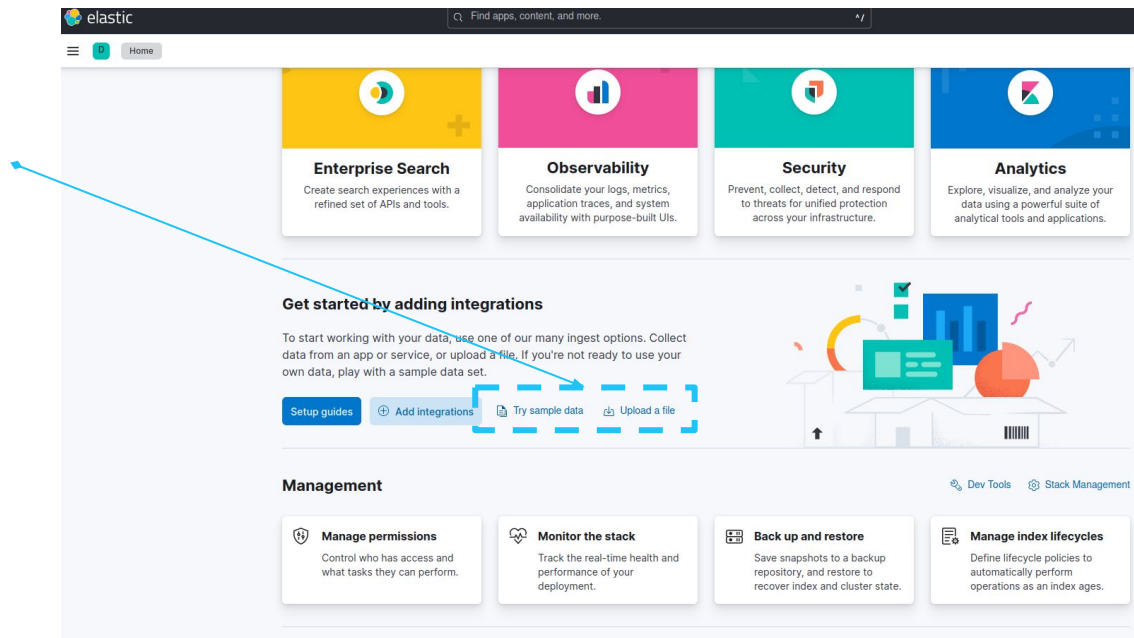


Pour réinitialiser le mot de passe : **Reset password**



# Importation de données dans cluster Elasticsearch sur Elastic Cloud

## Importer des données en local :



## Exemples de jeux de données avec des données textuelles :

- <https://www.kaggle.com/datasets/abdallahwagih/books-dataset>
- <https://www.kaggle.com/datasets/thedevastator/books-sales-and-ratings>

En résumé, nous explorerons comment accéder et configurer Kibana une fois que votre déploiement sur **Elastic Cloud** est en place.

**Kibana** est votre fenêtre sur votre cluster Elasticsearch, vous permettant de gérer vos données et de créer des visualisations et des tableaux de bord.

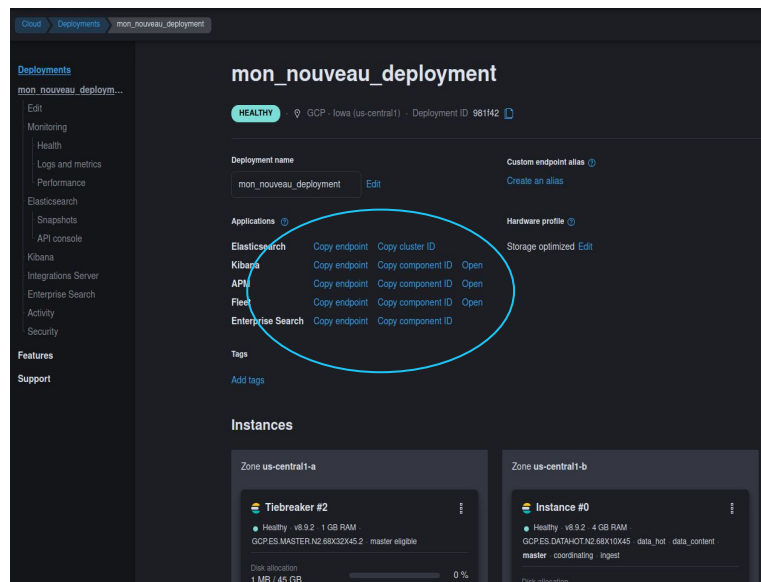
## Étapes initiales :

1. Accédez à votre déploiement sur Elastic Cloud.
2. Trouvez le lien "Kibana" et copiez l'Endpoint.
3. Ouvrez cet Endpoint dans un navigateur pour accéder à l'interface Kibana.
4. Une fois sur Kibana, prenez un moment pour explorer les différentes fonctionnalités disponibles dans le menu latéral gauche.
5. Chargez les données d'échantillon fournies pour commencer à explorer les fonctionnalités de visualisation de Kibana.



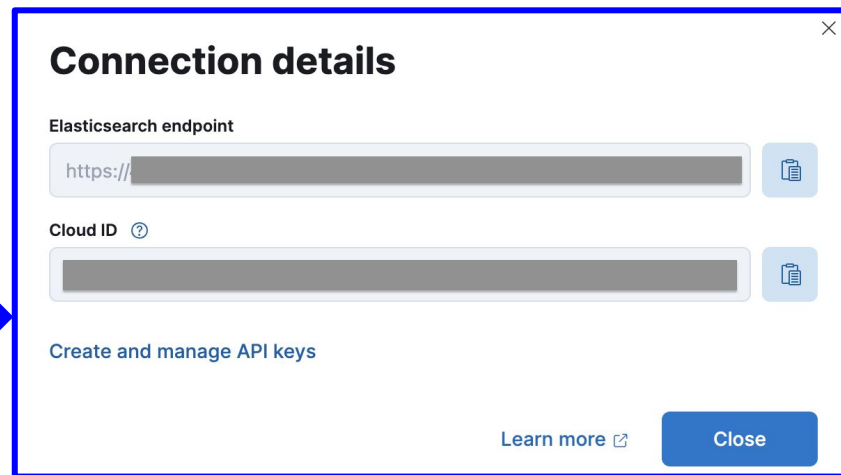
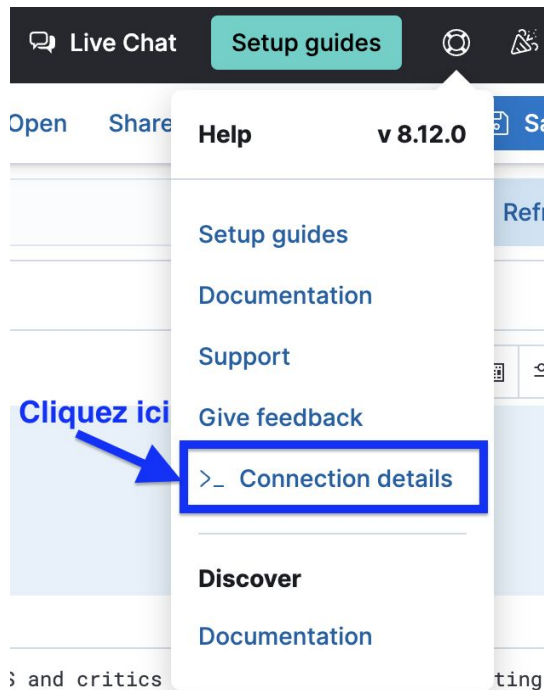
Dans la vue **Deployments** d'Elasticsearch sur Elastic Cloud :

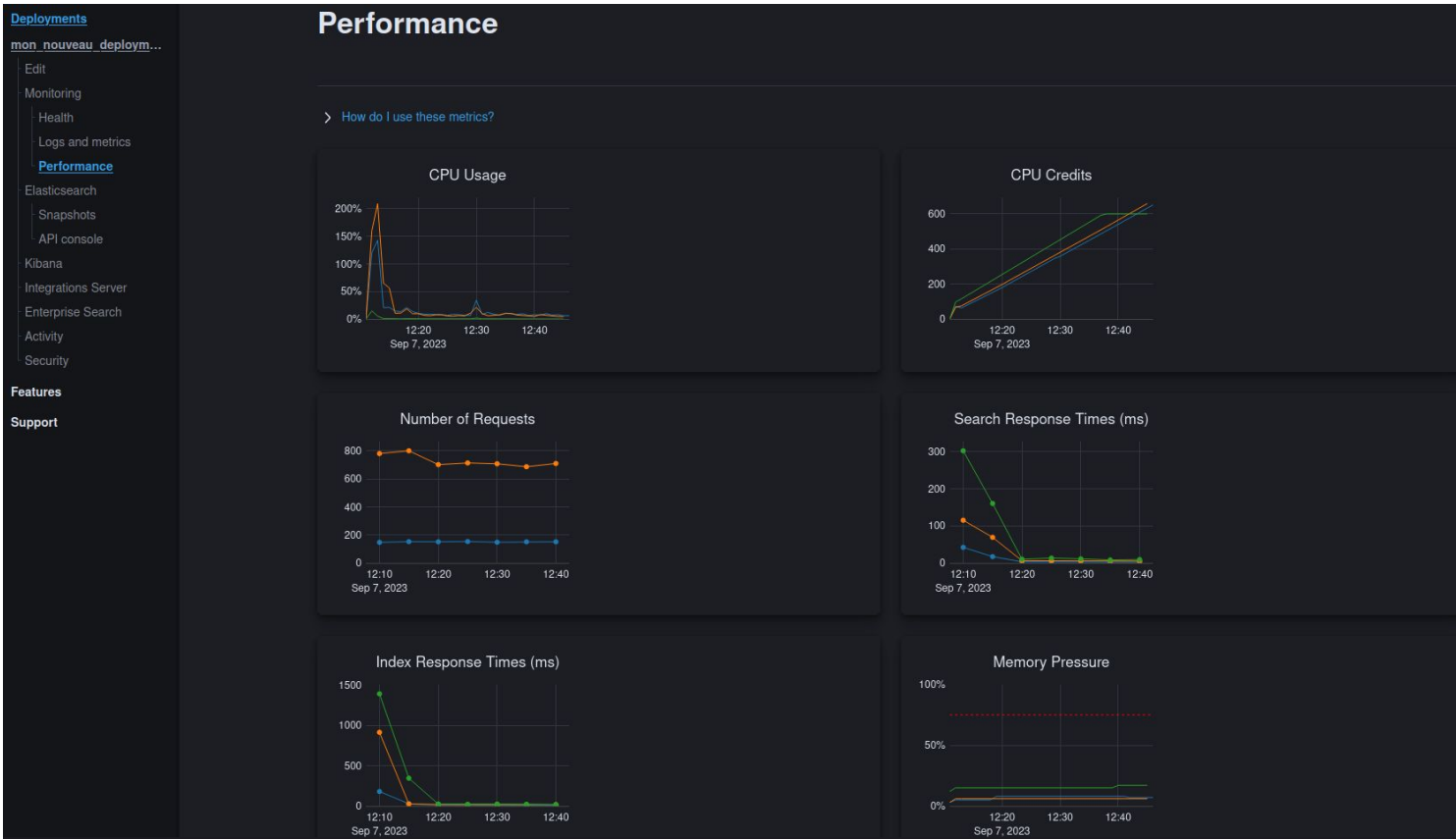
- **Copy endpoint** : vous permet de copier l'URL de point de terminaison (endpoint) de votre déploiement.
- **L'URL de point de terminaison** : est l'adresse à laquelle votre cluster Elasticsearch est accessible. Elle est généralement utilisée pour interagir avec votre cluster via des clients Elasticsearch, Kibana, ou d'autres outils.

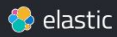


# Exemples vue elastic cloud ...



Il faut tout d'abord créer la connexion à la base Elasticsearch. Vous trouverez ci-dessous le code afin de créer la connexion avec vos propres identifiants de connexion "username" et "password". Pour récupérer le lien URL de connexion, vous suivrez les étapes suivantes depuis votre navigateur et votre compte sur Elastic Cloud.










Find apps, content, and more.




 Home

## Welcome home




### Observability

Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.



### Security

Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.







### Analytics

Explore, visualize, and analyze your data using a powerful suite of analytical tools and applications.

### Get started by adding integrations

To start working with your data, use one of our many ingest options. Collect data from an app or service, or upload a file. If you're not ready to use your own data, play with a sample data set.



### Try managed Elastic

Deploy, scale, and upgrade your stack faster with Elastic Cloud. We'll help you quickly move your data.

Move to Elastic Cloud

# 05 Ateliers pratiques

1

# « TD 1 : Prise en main Elasticsearch »

# 1 TD n°1 : Prise en main Elasticsearch

Feuille de TD n°1 : [cliquer ici pour télécharger le fichier](#)

Notebook de TD n°1 : [cliquer ici pour télécharger le fichier](#)

Good luck and work!

## 2

« TD 2 – Machine Learning :  
*Prédiction de la note moyenne des livres* »



## 2 TD n°2 – Machine Learning : Prédiction de la note moyenne des livres

Feuille de TD n°2 : [cliquer ici pour télécharger le fichier](#)

Notebook de TD n°2 : [cliquer ici pour télécharger le fichier](#)

Good luck and work!

MILESKER  
—  
MERCİ



***“Garapen ekonomikoa xedea baino gehiago, baliabide bat da”***

*“Le Développement Économique est un Moyen et pas une Finalité”*

### **HUPI S.A.S.**

Technopole Izarbel  
45 allée Théodore Monod  
64210 Bidart, FRANCE

### **HUPI IBERICA S.L.U.**

Gipuzkoako Parke Teknologikoa  
Paseo Miramon N°170  
20009 San Sebastian, SPAIN

[contact@hupi.fr](mailto:contact@hupi.fr)