

## ElasticSearch : mise en oeuvre et programmation

CB021

Durée: 2 jours

1220 €

6 au 7 mars  
19 au 20 juin

25 au 26 septembre  
11 au 12 décembre

### Public:

Architectes techniques, ingénieurs système, administrateurs..

### Objectifs:

Comprendre le fonctionnement et les apports d'Elasticsearch dans le traitement de données, et savoir le mettre en oeuvre, analyser les données, programmer des requêtes et créer des rapports et tableaux de bord avec kibana.

### Connaissances préalables nécessaires:

Connaissances générales des systèmes d'information, et des systèmes d'exploitation (Linux ou Windows). Les travaux pratiques sont réalisés sur Linux. Connaissance d'un langage de programmation structuré

### Programme:

- Introduction : Présentation ElasticSearch, fonctionnalités, licence  
Les différentes versions : fonctionnalités et particularités des versions de 2.0 à 5.0.  
Nouveautés de la version 6.0.  
Positionnement d'Elasticsearch et des produits complémentaires : Watcher, Marvel, Kibana, Logstash, Beats, X-Pack  
Principe : base technique Lucene et apports d'ElasticSearch  
Fonctionnement distribué
- Installation et configuration : Prérequis techniques.  
Utilisation de l'interface Marvel.  
Premiers pas dans la console Sense.

## ElasticSearch : mise en oeuvre et programmation

Format et stockage des données	: Format des données. Conversion au format JSON des données à traiter. Structure des données. Stockage, indexation. Terminologie Elasticsearch : notions de document, type, index Métadonnées : <code>_index</code> , <code>_type</code> , <code>_ID</code> Choix de l'identifiant par l'application avec l'API index, ou génération automatique d'un identifiant.. Indexation inversée.
Outils d'interrogation	: Java API avec "Node client" et "Transport client" API RESTful en HTTP Exemples de requêtes simples et plus complexes : recherche de «phrases», extraction de plusieurs documents, etc .. Notion de pertinence du résultat : «score» Requêtes avec Search Lite et avec Query DSL (domain-specific language) Utilisation de 'filtre' pour affiner des requêtes. Autres clients : Perl, Python, Ruby, etc... Agrégation de résultats.
Mises à jour	: Fonctionnement d'Elasticsearch pour les ajouts, modifications, suppression. Notion de version affectée par Elasticsearch. L'API bulk pour les traitements groupés. Réalisation de scripts avec groovy
Gestion des accès concurrents	: Utilisation du numéro de version. Gestion par l'application : différentes méthodes selon les contraintes fonctionnelles. Utilisation d'un numéro de version externe.
Kibana présentation	: Fonctionnalités : recherche, visualisation, création de tableaux de bord et graphiques à partir des données fournies par Elasticsearch

## ElasticSearch : mise en oeuvre et programmation

CB021

Kibana, mise en oeuvre : Installation, configuration du mapping avec Elasticsearch.

Paramétrage dans le fichier kibana.yml

Mapping automatique ou manuel. Configuration des indexes à explorer.

Visualisation et sauvegarde de graphiques, étude des différents types de graphiques disponibles, création de tableaux de bord et rapports à partir des graphiques.