Estudiante: *Jose Pablo Arias Navarro - 2021024635*

Resumen 1

What is Elasticsearch?

Elasticsearch es el motor de análisis y búsqueda de Elastic Stack. Este hace uso de ***Logstash*** y ***Beats***, los cuales facilitan la recopilación, agregación y almacenamiento de los datos en Elasticsearch. Además, también hace uso de ***Kibana*** el cual permite explorar y visualizar los diferentes datos indexados en Elasticsearch.

Elasticsearch proporciona búsqueda y análisis casi en tiempo real para todo tipo de datos. Este puede almacenarlos e indexarlos de manera eficiente lo cual permite que se realicen búsquedas rápidas. Asimismo, con este también podemos recuperar datos y agregar información con el fin de descubrir tendencias y patrones en los datos almacenados. Podriamos decir que Elasticsearch ofrece velocidad y flexibilidad para manejar datos en una amplia variedad de casos.

Data in: document and indices

Elasticsearch almacena estructuras de datos complejas que se han serializado como documentos JSON. Es importante mencionar que cuando se tienen varios nodos de Elasticsearch en un clúster, los documentos almacenados se distribuyen por todo el cluster y se puede acceder a ellos de inmediato desde cualquier nodo.

Elasticsearch hace uso de una estructura de datos llamada ***índice invertido***, la cual básicamente enumera cada palabra única que aparece en cualquier documento e identifica todos los documentos en los que aparece cada palabra, permitiendo búsquedas rápidas de textos completos. (Para tener una idea de como funciona este índice, lo podemos ver como una colección de documentos y cada documento es una colección de campos, los cuales son una pareja llave-valor que contiene sus datos)

En Elasticsearch también es posible indexar documentos sin especificar explícitamente como manejar cada uno de los diferentes campos que aparecen en un documento, ya que cuando el mapeo dinámico se encuentra activado, este detecta y agrega automáticamente nuevos campos al índice. Esto hará que podamos comenzar a indexar documentos y Elasticsearch automáticamente detectará y asignará los tipos de datos apropiados al campo correspondiente.

Information out: search and analyze

En parte, el verdadero poder de Elasticsearch proviene del fácil acceso que tiene al conjunto completo de capacidades de búsqueda integradas en la librería del motor de búsqueda Apache Lucene. Elasticsearch suministra una API REST para administrar nuestro cluster, indexar y buscar los datos.

Las API REST de Elasticsearch permiten los siguientes tipos de consultas:

- **Consultas estructuradas:** Estas son similares a las consultas que podemos utilizar en SQL.
- **Consultas de texto completo:** Estas encuentran todos los documentos que contienen el string consultado y los retorna ordenados por relevancia.

- **Consultas complejas:** Estas combinan los dos tipos de consultas anteriormente mencionadas.

Además de buscar términos individuales, puede realizar búsquedas de frases, similitud y prefijo, e incluso ofrece autocompletado. Podemos acceder a todas estas capacidades de búsqueda utilizando el lenguaje de consulta de estilo JSON de Elasticsearch. No menos importante, para buscar y agregar datos de forma nativa dentro de Elasticsearch también podemos utilizar consultas de tipo SQL.

Las agregaciones de Elasticsearch le permiten crear resúmenes complejos y obtener información sobre métricas, patrones y tendencias. Estas agregaciones son muy rápidas, esto le permite analizar y visualizar sus datos en tiempo real. Otro aspecto importante es que se puede utilizar machine learning principalmente para conocer el comportamiento normal de los datos y para identificar patrones con alguna anomalía.

Scalability and resilience: clusters, nodes y shards

Elasticsearch esta construido para tener una alta disponibilidad y escalabilidad. Es capaz de agregar servidores (nodes) a un clúster para aumentar la capacidad y posteriormente distribuir automáticamente la carga de datos y consultas en todos los nodos disponibles.

Para entender un poco mejor, un índice en Elasticsearch es en realidad una agrupación lógica de uno o más shards físicos, donde cada shard es un índice autónomo, lo cual al distribuir los documentos en un índice a través de múltiples shards y distribuir esos shards en varios nodos provoca que Elasticsearch pueda garantizar redundancia, ocasionando como se mencionó anteriormente, una gran disponibilidad en caso de fallas de hardware y aumenta la capacidad de consultas entre más nodos sean agregados a un cluster.

Hay dos tipos de shards, de los cuales podemos decir lo siguiente:

1. **Primarios:** Cada documento de un índice pertenece a una shard primario y la cantidad de shards primarios en un índice se fija en el momento en que se crea el índice.
2. **Réplicas:** Es una copia de un shard primario, estas proporcionan copias redundates de sus datos para proteger contra fallas de hardware y aumentar la capacidad de atender solicitudes de lectura, búsqueda o recuperación de un documento.

Entre más cantidad de shards, más gastos generales habrá simplemente por el mantenimiento de esos índices y entre mayor sea el tamaño del shard, más tiempo tomará mover los fragmentos cuando se necesite reequilibrar o balancear un cluster.

Para prevenir un caso de desastre en el que un servidor se cae y necesitamos que otro servidor lo suplante rápidamente es importante conocer el concepto de ***Cross-cluster replication (CCR)***. Básicamente, el CCR sincroniza automáticamente los índices del clúster principal con un clúster remoto el cual también puede servir como copia de seguridad activa. Otro aspecto importante es que se puede usar el CCR para crear clusters secundarios los cuales en muchos casos son utilizados para atender solicitudes de lectura. (Cluster primario maneja solicitudes de escritura y cluster secundario solo de lectura)

Con respecto al cuidado y la alimentación, para poder asegurar, monitorear y administrar los clusters de Elasticsearch se puede utilizar ***Kibana*** como un centro de control.

Información recuperada de What is Elasticsearch?