

Instituto Tecnológico de Costa Rica**IC4302 - Bases de Datos II****Resumen 1****Profesor:** Nereo Campos Araya**Estudiante:** Melany Salas Fernández - 2021121147

Data in- Documents and indices

Elasticsearch **guarda estructuras de datos complejas**, serializadas en documentos JSON, además, cuando se tienen múltiples nodos en un clúster, los documentos guardados se distribuyen y pueden ser accedidos desde cualquier nodo.

Cuando se guardan los documentos, son indexados, esto permite una búsqueda en tiempo muy cercano al real. La **indexación** se hace mediante índice invertido (inverted index) que permite búsqueda en textos de forma rápida, listando cada palabra que aparece en el documento e identificándola. Este índice puede verse como una colección de documentos donde estos son una colección de campos llave valor, cada una de estas tiene una estructura optimizada, esto es lo que hace que los resultados se obtengan rápidamente.

Elasticsearch también permite que los documentos se indexen sin que se especifique como manejar los campos, mediante el uso mapping dinámico, que hace una detección automáticamente, facilitando la indexación y la exploración de datos. Sin embargo, se pueden definir reglas para controlar el mapping dinámico y decidir cómo se guardan los campos y cómo se indexan, esto permite personalizar formatos, implementar análisis de texto en lenguajes específicos, entre otros. En varias ocasiones, es útil indexar un campo de formas diferentes para dependiendo del propósito.

Information out: search and analyze

Con Elasticsearch se puede acceder a las capacidades de búsqueda de la librería Apache Lucene. Elasticsearch brinda una Rest Api para el manejo del clúster, la indexación y la búsqueda de datos, también permite subir solicitudes desde la línea de comandos o desde la Developer Console en Kibana.

Búsqueda en los datos

Las Rest Api soportan distintas queries:

- **Queries estructuradas:** Similares a SQL
- **Full text queries:** Encuentran documentos que hacen match con el query string y los devuelven ordenados de acuerdo con que tan buen match hacen.

También hay queries que combinen ambos tipos, llamadas **queries complejas**.

Por otro lado, para búsquedas de términos se puede usar **phrase searches**. Para datos geoespaciales y numéricos, Elasticsearch tiene **non-textual indexes y estructuras optimizadas**.

El acceso a las capacidades de búsqueda de Elasticsearch se permite mediante el **JSON-style query language** (Query DSL). También se puede construir SQL-style query para buscar y agregar datos nativamente.

Análisis de datos

Las **agregaciones** permiten construir resúmenes de datos para tener patrones, estas también implementan las estructuras optimizadas de Elasticsearch, por lo que son bastante rápidas, permitiendo la visualización de datos en tiempo real. Además, las agregaciones trabajan en conjunto con las solicitudes de búsqueda, por lo que se puede filtrar resultados y analizar datos a la vez.

Finalmente, Elasticsearch permite el **uso de machine learning**, este sirve para la detección de anomalías en el comportamiento de datos, identificar comportamientos...

Scalability and resilience

Elasticsearch es **distribuido**, para mantener la disponibilidad, esto también le permite escalar de acuerdo a las necesidades. Gracias a esto, se pueden agregar nodos a un clúster para aumentar la capacidad y Elasticsearch se encargará de distribuir la información en los nodos disponibles.

Los **índices** de Elasticsearch son básicamente la agrupación de fragmentos de base de datos físicos, donde cada uno es un índice autocontenido.

Elasticsearch usa la **redundancia** para proteger contra fallos de hardware, además cuando un clúster crece, el fragmento de la base de datos migra de forma automática para balancear el clúster.

Tipos de fragmentos:

- **Primarios.**
- **Replicas:** Copia el fragmento primario, usando la redundancia.
Las réplicas también permiten el aumento de la capacidad de lectura para las consultas.

Algunas consideraciones

El uso de fragmentos implica:

- Entre más fragmentos, más difícil o costoso será el mantenimiento.
- Entre más grandes sean los fragmentos, más difícil balancear el clúster.
- Consultar muchos fragmentos pequeños podría hacer el procesamiento más rápido, pero es más complicado, por lo que, dependiendo, consultar pocos fragmentos más grandes podría ser más rápido.

Debido a esto, se sabe que lo mejor en realidad depende, lo ideal es hacer una revisión y análisis de los datos para escoger la solución óptima.

Cross-cluster replication

Forma automática de sincronizar los índices del nodo primario al secundario, esto para mantener la disponibilidad de la base de datos en caso de fallas, además estos índices pueden usarse para la lectura de datos.

Kibana

Es un centro de control para manejar los clúster.

Resumen de [What is Elasticsearch?](#)