**ELASTIC SEARCH**

**INTRODUCTION OF ELASTIC SEARCH**

 **Web Application:** This represents the user interface where users interact with the search system (optional).

 **Load Balancer:** Distributes incoming search requests across the Elasticsearch cluster for better performance and scalability.

 **Elasticsearch Cluster:** This is the core of the system, consisting of multiple Elasticsearch nodes working together.

 **Client Applications:** These are applications that interact with the Elasticsearch cluster using APIs for indexing and searching data.

 **Shards:** Data is divided and distributed across these shards, which reside on different nodes in the cluster. This allows for scalability and parallel processing of search queries.

 **APIs:** Elasticsearch provides various APIs for interacting with the cluster:

* **Search API:** Allows searching for documents based on keywords, phrases, or specific criteria.
* **Indexing API:** Used to add, update, or delete documents within an index.
* **Bulk API:** Enables efficient indexing of large datasets by sending multiple documents in a single request
* Elasticsearch là một search engine.
* Elasticsearch được kế thừa từ Lucene Apache
* Elasticsearch thực chất hoặt động như 1 web server, có khả năng tìm kiếm nhanh chóng (near realtime) thông qua giao thức RESTful
* Elasticsearch có khả năng phân tích và thống kê dữ liệu
* Elasticsearch chạy trên server riêng và đồng thời giao tiếp thông qua RESTful do vậy nên nó không phụ thuộc vào client viết bằng gì hay hệ thống hiện tại của bạn viết bằng gì. Nên việc tích hợp nó vào hệ thống bạn là dễ dàng, bạn chỉ cần gửi request http lên là nó trả về kết quả.
* Elasticsearch là 1 hệ thống phân tán và có khả năng mở rộng tuyệt vời (horizontal scalability). Lắp thêm node cho nó là nó tự động auto mở rộng cho bạn.
* Elasticsearch là 1 open source được phát triển bằng Java

**5 BENEFITS:**

 **Scalability and Performance:** Compared to traditional relational databases, Elastic Search excels in handling large and ever-growing datasets. Its distributed architecture allows horizontal scaling by adding more nodes to the cluster, efficiently handling increasing data volumes and search queries. This is in contrast to relational databases, where scaling often involves expensive hardware upgrades.

 **Real-time Search:** Unlike some databases with batch indexing processes, Elastic Search offers near real-time search capabilities. This means data becomes searchable almost instantaneously after indexing, providing users with up-to-date information. This is valuable for applications requiring real-time search functionality, such as e-commerce product searches or news aggregators.

 **Flexible Data Schema:** Elastic Search is known for its schema-less approach. Unlike relational databases with rigid schemas defined upfront, Elastic Search allows you to store data in a flexible JSON format. This makes it ideal for handling unstructured and semi-structured data like logs, emails, and social media content, which can be challenging to model in traditional databases.

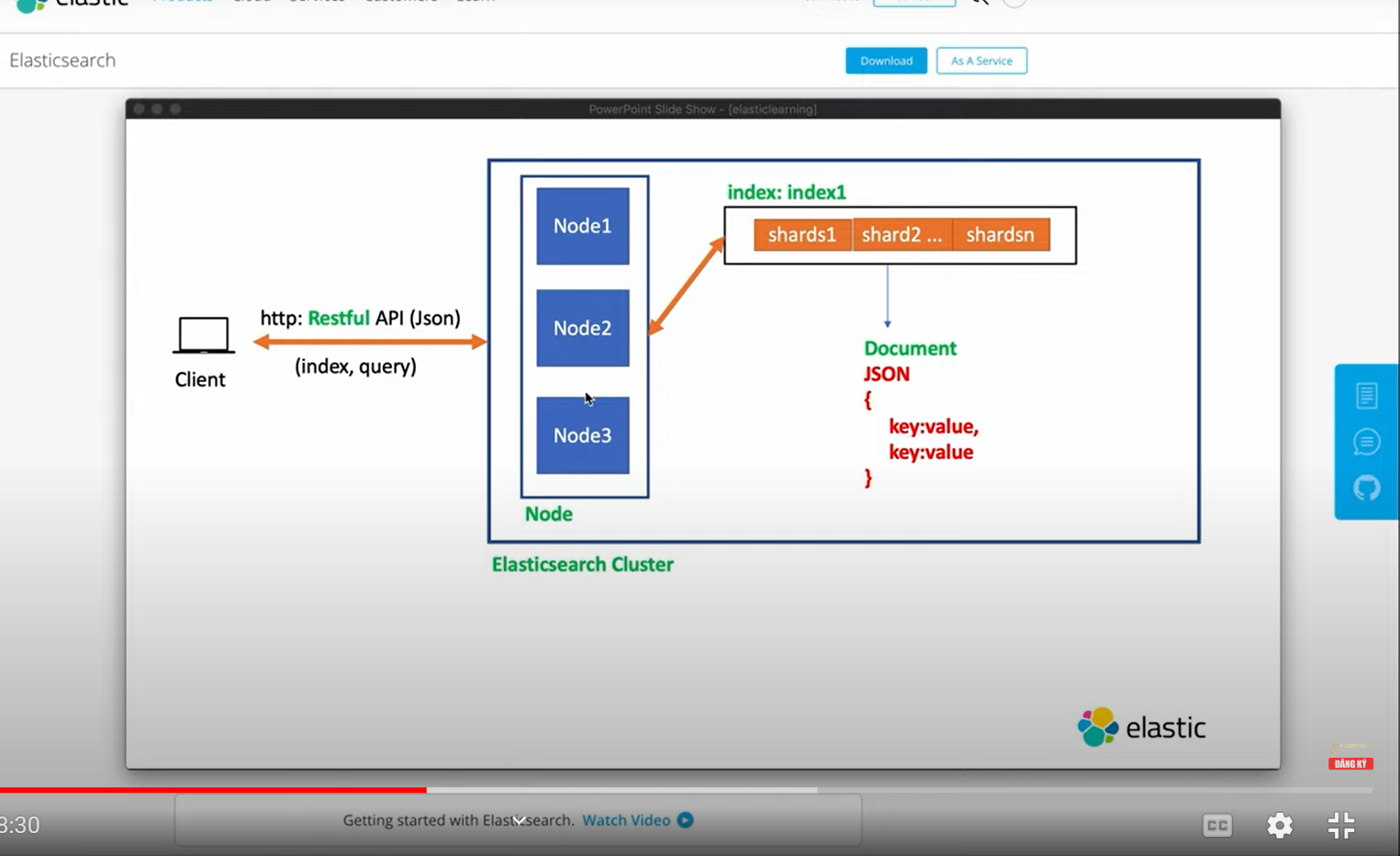
 **Powerful Search Capabilities:** Elastic Search boasts a rich set of query features beyond simple keyword searches. It supports full-text search, allowing searches across entire documents, as well as various query types like wildcard searches, phrase searches, and aggregations for grouping and analyzing search results. This flexibility empowers users to find relevant information within large datasets with greater precision.

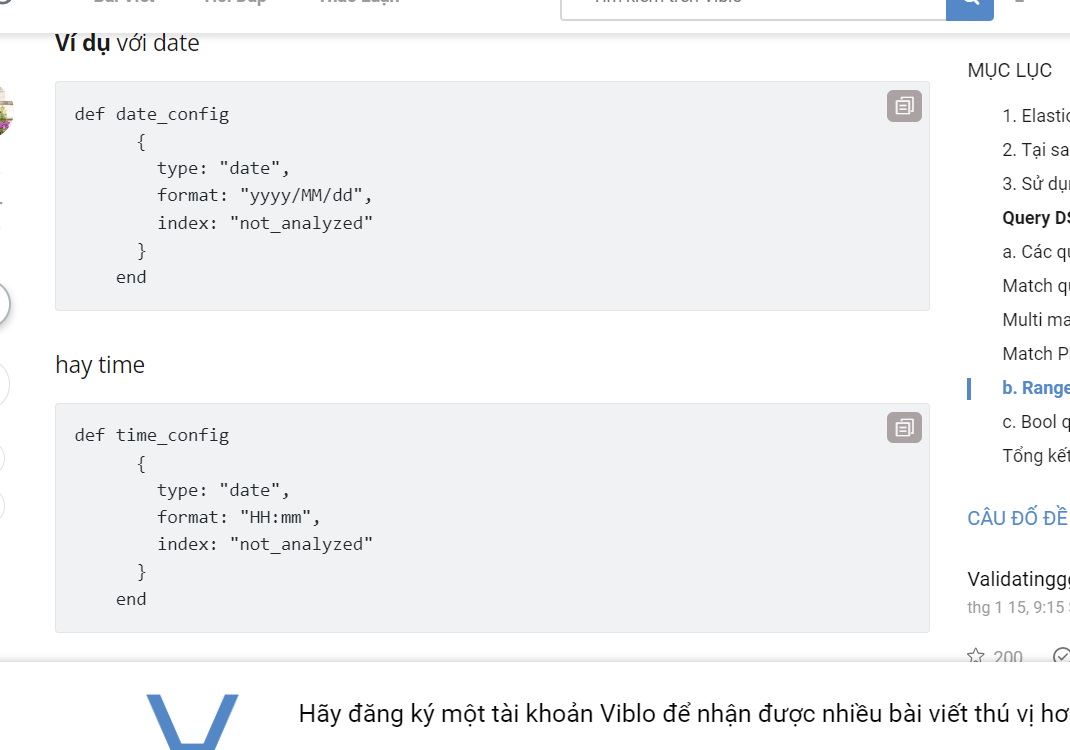
 **Distributed Architecture and Fault Tolerance:** Elastic Search's distributed nature offers inherent fault tolerance. Data is sharded and replicated across multiple nodes in the cluster. If a node fails, the remaining nodes can continue to service search requests without significant downtime. This redundancy ensures high availability and protects against hardware failures.

 **Open-Source and Cost-Effective:** A significant advantage of Elastic Search is its open-source nature. This translates to cost savings compared to proprietary search solutions. While there is a commercial version with additional features and support, the open-source version offers a robust foundation for building search applications. Additionally, the large and active open-source community provides support and resources for learning and troubleshooting.

**Data Flow:**

1. Users upload documents (images/PDFs) through the client application.
2. The web server receives the upload request and forwards it to the Tesseract OCR component.
3. The document is sent to Tesseract OCR for text extraction.
4. The extracted text is passed to the Indexer component.
5. The Indexer prepares the data for indexing in Elastic Search, including adding metadata or formatting the text according to the defined mapping.
6. The prepared data is sent to the Elastic Search cluster for indexing.
7. Documents are stored and distributed across shards within the cluster.
8. Users interact with the search interface (hosted on the web server) to search for indexed documents based on keywords or phrases within the extracted text content.





**Sources**: <https://www.elastic.co/blog/how-to-ingest-data-into-elasticsearch-service>

<https://www.ijsr.net/archive/v10i6/SR21529233126.pdf>

<https://viblo.asia/p/gioi-thieu-ve-elastic-phan-1-gioi-thieu-co-ban-ve-elastic-search-va-huong-dan-cai-dat-co-ban-step-by-step-1VgZvVNmZAw>

<https://topdev.vn/blog/elasticsearch-la-gi/>

<https://viblo.asia/p/phan-1-nhung-khai-niem-co-ban-trong-elasticsearch-V3m5WzzglO7>

## <https://viblo.asia/p/cach-tao-index-va-mot-so-truy-van-pho-bien-trong-elasticsearch-1VgZvO39lAw#_index-elasticsearch-la-gi-0>

<https://viblo.asia/p/elasticsearch-va-query-dsl-p1-OeVKBdY0lkW#_match-query-5>

<https://viblo.asia/p/elasticsearch-va-query-dsl-p2-4dbZN4ky5YM>

<https://viblo.asia/p/xay-dung-webapp-doc-bien-so-xe-su-dung-yolov6-tesseract-ocr-streamlit-vlZL9bx8VQK>