

|  |
| --- |
| STRUCTURE ELASTICSEARCH INDEXING WHILE IMPORTING HETEROGENEOUS DATA FORMATS  2018 |
|  |
| November 9  A joint paper by JP Tokyo & Co and MRIIRS  Authored by: <Nikhil> Reviewed by: Rajesh Nath, <DR. Madhulika> |

**Contents**

**1.) Introduction, purpose and heading………………………………………………………….3**

**2.) Description, Platforms and Technologies…….……………………………………………7**

**3.) Findings and verification….……………………………………………………………………….8**

**4.) About……………………………………………………………………………………………………….9**

**5.) Project artifacts….……………………………………………………………………………………10**

**6.) References………………………………………………………………………………………………11**

Introduction

This white Paper is a report of the overall work that has been done and mainly focuses on “Structure Elasticsearch indexing when importing heterogeneous data formats” using different platforms.

About Elasticsearch

Elasticsearch is an exceptionally adaptable open-source full-content search and analytics engine. It enables you to store, seek, and dissect enormous volumes of information rapidly and in real time. It is for the most part utilized as the basic engine/innovation that powers applications that have complex search highlights and prerequisites.

Here are a few sample use-cases that Elasticsearch could be used for:

* You run an online web store where you enable your clients to scan for items that you offer. For this situation, you can utilize Elasticsearch to store your whole item list and stock and give look of and autocomplete recommendations to them.
* You need to gather log or exchange information and you need to examine and mine this information to search for patterns, insights, outlines, or inconsistencies. For this situation, you can utilize Logstash (some portion of the Elasticsearch/Logstash/Kibana stack) to gather, total, and parse your information, and afterward have Logstash feed this information into Elasticsearch. When the information is in Elasticsearch, you can run inquiries and collections to mine any data that is important to you.
* You run a price altering platform which permits value astute clients to indicate a standard like "I am willing to purchase a particular electronic item and I need to be informed if the cost of device falls beneath $X from any seller inside the following month". For this situation you can import seller costs, push them into Elasticsearch and utilize its reverse-search (Percolator) capacity to coordinate value developments against client inquiries and in the end drive the alarms out to the client once coordinates are found.

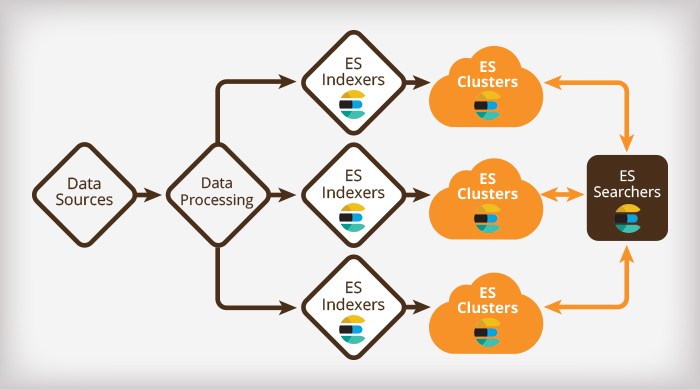
Description

This Data model of Elastic search is used to search queries as quick as could be expected under the circumstances in order to stay in the ceaseless stream of getting the outcomes. Elasticsearch is a database that stores, recovers, and oversees report situated and semi-organized information.

Items that include web based business and web crawlers with tremendous databases are confronting issues, for example, item data recovery taking too long. This prompts poor client encounter and thus kills potential clients.

Slack in pursuit is credited to the social database utilized for the item plan, where the information is scattered among numerous tables and recovery of significant client data require bringing the information from them. The Relational Database works nearly ease back with regards to gigantic information and getting list items through questions from the database. Organizations these days searching for exchange ways where the information put away so that the recovery is brisk. This can be accomplished by embracing NOSQL as opposed to RDBMS for putting away information. Elasticsearch is one such NOSQL disseminated database. Elasticsearch depends on adaptable information models to manufacture and refresh guests profiles to meet the requesting remaining burden and low inactivity required for constant commitment.

Social database works relatively ease back with regards to enormous information and getting indexed lists through questions from the database. (There are approaches to upgrade this like ordering yet then there are connected confinements like we can't list each field. Column updates to vigorously filed tables would require some investment. Individuals likewise scale their RDBMS vertically to enhance execution.) This is an issue is overwhelmed by Elasticsearch. Beneath figure indicates how RDBMS in a perfect world function for seeking things from the Database.



There are a couple of ideas that are center to Elasticsearch. Understanding these ideas from the start will enormously help facilitate the learning procedure.

NEAR REALTIME (NRT)

Elasticsearch is a near real time search platform. This means there is a slight inertness (ordinarily one moment) from the time you record a report until the point that the time it becomes searchable.

CLUSTER

A cluster is a gathering of at least one hubs (servers) that together holds your whole information and gives combined ordering and pursuit capacities over all hubs. A cluster is recognized by a one of a kind name which naturally is "elasticsearch". This name is imperative on the grounds that a node must be a piece of a group if the node is set up to join the cluster by its name.

INDEX

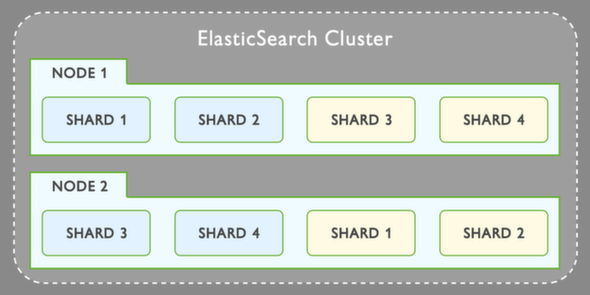
A index is an accumulation of archives that have to some degree comparable qualities. For instance, you can have an index for client information, another index for an item inventory, but another index for request information. An index is recognized by a name (that must be all lowercase) and this name is utilized to allude to the index when performing ordering, inquiry, refresh, and erase tasks against the archives in it.

In a single cluster, you can characterize any number of indexes as you need.

SHARDS AND REPLICAS

A index can conceivably store a lot of information that can surpass the equipment furthest reaches of a solitary hub. For instance, a solitary record of a billion archives taking up 1TB of plate space may not fit on the circle of a solitary index or might be too ease back to serve seek demands from a solitary index alone.

To take care of this issue, Elasticsearch gives the capacity to subdivide your index into numerous pieces called shards. When you make an index, you can just characterize the quantity of shards that you need. Every shard is in itself a completely useful and a free "index" that can be facilitated on any node in the cluster.



Platforms and Technologies

The platforms and technologies used while making the project are discussed below:

* Elasticsearch: It is an [open source](https://whatis.techtarget.com/definition/open-source) [search engine](https://whatis.techtarget.com/definition/search-engine) built on top of [Apache Lucene](https://whatis.techtarget.com/definition/Apache-Lucene) and released under an [Apache](https://whatis.techtarget.com/definition/Apache) license. It is [Java](https://www.theserverside.com/definition/Java)-based and can search and index document files in diverse formats.
* Docker: Docker is an open platform that assists with all inclusive circulation of uses. It has turned into a standard for particular kinds of holder virtualization frameworks and has been embraced by different organizations as a product compartment technique.
* cURL: cURL is a PC programming venture giving a library and command line tool for exchanging information utilizing different conventions. The cURL venture produces two items, libcurl and cURL. It was first unchained in 1997.
* Java Development kit(JDK): The Java Development Kit (JDK) is a software development environnment utilized for creating Java applications and applets. It incorporates the Java Runtime Environment (JRE), a mediator/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and different devices required in Java advancement.
* Logstash: Logstash is a tool for overseeing events and logs. At the point when utilized conventionally, the term envelops a bigger arrangement of log accumulation, handling, stockpiling and looking exercises.

LIBRARIES

* PYPDF2: PyPDF2 is an unadulterated python PDF library equipped of splitting, combining, editing, and changing the pages of PDF files.It can recover content and metadata from PDFs and also consolidate whole documents.
* **OS-** The OS module in Python provides a way of using operating system dependent functionality. The functions that the OS module provides allows you to interface with the underlying operating system that Python is running on – be that Windows, Mac or Linux.
* **Glob**-The glob module finds all the pathnames matching a specified pattern according to the rules used by the Unix shell, although results are returned in arbitrary order**.**

**Findings and verifications**

The aim was to import heterogeneous data into Elasticsearch so that it can further be used to create a database and later be embedded into ES for the required purposes.

Heterogeneous data and findings were the following:

* Imported xlsx, xls and csv files to Elasticsearch
* Explained how to load JSON files into Elasticsearch
* Explained how to copy SQL server data into Elasticsearch using Logstash

All the work that was being done was in constant supervision of our respective mentors and reviewers. Also periodic updates were provided to the mentors so that the work moves in the right direction following all the required protocols.

Also the work used to be discussed with our respective teams and at last all the work and findings were consolidated.

**ABOUT**

1. **CONTRIBUTOR: NIKHIL**

****

A dedicated person who wants to keep on learning new things and stay relevant down the line over the coming years and try to be a step ahead of this challenging world.

* Class representative in college since last 3 years have very well managed it till now.
* Student Placement Coordinator, Manav Rachna International institute of research and studies
* Campus Ambassador, TATA consultancy services
* Was a part of the Panel discussion on the topic IOT among various experts
* Published 2 research papers and currently working on 2 more
* Good in programming languages like C,C++,HTML,MYSQL
* Favoured to subjects like Data structures,Database Concepts, Operating Systems
* Also interested in Fashion shows, already done 2,was a shows topper in one of them and winner of the other

1. **Mentors**

* **Mr.Rajesh Nath-**Associate Vice-President of Jp Tokyo and co.

rajesh.nath@jptokyo.co.jp

* **Dr.Madhulika-**Associate professor in MRIIRS

[Madhulika.fet@mriu.edu.in](mailto:Madhulika.fet@mriu.edu.in)

* **JP Tokyo-**info@jptokyo.co.jp
* [**MRIIRS-**delhi@mrei.ac.in](mailto:MRIIRS-delhi@mrei.ac.in)

**PROJECT ARTIFACTS**

1. *Parses XLSX/XLS/CSV files into ElasticSearch using column titles from a specified row combined with data in columns on each row*

Tested with ElasticSearch 5.6.2 and 6.4.2.

Running with docker

docker run -it -p 5252:5252 -e es\_port=9200 -e es\_host=localhost codingchili/excelastic

Note: there is currently no support for TLS or basic authentication when using the docker image.

Running the JAR

Running the application, filename and index is required, to import from the terminal run:

java -Xmx2g -jar excelastic.jar <fileName> <indexName> --mapping mappingName --clear

If running with --clear, then the existing index will be cleared before the import starts.

To run with the web interface, run the following in your terminal:

java -Xmx2g -jar excelastic.jar

When the application successfully connects to the ElasticSearch server, the browser will automatically open a new tab.

If any connection errors occur check that the ElasticSearch listen port matches with the elastic\_port in the configuration file. Make sure that ElasticSearch is running by directing your browser at localhost:9200.

Compiling a new fatjar and run tests,mvn clean package

├── configuration.json

The configuration file is placed in the same directory as the jar. An example of the configuration:

{ "web\_port": 0, // the port the web interface listens on "elastic\_port": 9200, // the port elasticsearch listens on "elastic\_host": "localhost", // address to elasticsearch "elastic\_tls": false, // set to true to use tls when indexing "authentication": false, // sends an "Authentication" header if true. "basic": "username:password" // if authentication is true this is used as basic authentication. }

If no configuration file is present the values in the above example will be used. Note that the comments cannot be included in the configuration file.

If no configuration file is present a new configuration file will be created using the default values listed here.

1. *Loading JSON files into Elasticsearch*

You have to install the curl binary in your PC first. You can download it from here.

After that unzip it into a folder. Lets say C:\curl. In that folder you'll find curl.exefile with several .dll files.

Now open a command prompt by typing cmdfrom the start menu. And type cd c:\curlon there and it will take you to the curl folder. Now execute the curl command that you have.

One thing, windows doesn't support single quote around around the fields. So you have to use double quotes. For example I have converted your curl command like appropriate one.

curl -H "Content-Type: application/json" -XPOST "http://localhost:9200/indexname/typename/optionalUniqueId" -d "{ \"field\" : \"value\"}"

3) *Copying SQL server data into Elasticsearch using Logstash*

STEPS:

1)Install Java Development Kit (JDK)

2)Install JDBC Driver for SQL Server.

3)Set CLASSPATH for the driver.

4)Create an Elasticsearch Index to Import Data to.

5)Configure LogStash configuration file.

6)Run LogStash.

7)Verify in Kibana.

**REFERENCES**

* <https://www.elastic.co/products/elasticsearch>
* <https://www.elastic.co/guide/en/logstash/current/index.html>
* <https://www.elastic.co/guide/en/elastic-stack/current/index.html>
* <http://www2.lixil.co.jp/rp/dfw/exsas6/webcatalog/kentikumuke.aspx?cate_cd=K1021&_ga=2.14736374.1555774946.1539170516-874854234.1539170516>
* [www.google.com](http://www.google.com)
* <https://youtu.be/x2KuhN1RZ4s>