[Nagarro]-[Ecommerce]

DAR Document

Nagarro Software Pvt. Ltd.

Akshat Aggarwal

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| Version | Date | Author/Contributor | Comments |
| 1.0 | October 23, 2024 | Akshat Aggarwal | |  | | --- | |  |  |  | | --- | | Initial draft | |
| 2.0 | December 08, 2024 | Akshat Aggarwal | Updated Comparison Analysis matrix |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1 Introduction 4](#_Toc421883713)

[1.1 Objective and scope of document 4](#_Toc421883714)

[2 Requirements at a Glance 5](#_Toc421883715)

[3 Available tools 6](#_Toc421883716)

[3.1 ElasticSearch 6](#_Toc421883717)

[3.1.1 Features 6](#_Toc421883718)

[3.1.2 Pricing 6](#_Toc421883719)

[3.2 Solr 6](#_Toc421883720)

[3.2.1 Features 6](#_Toc421883721)

[3.2.2 Pricing 6](#_Toc421883722)

[4 Comparison Analysis 7](#_Toc421883723)

[4.1 Point Matrix 7](#_Toc421883724)

[4.2 Comparison 1: Speed, Performance and Ease of Setup 7](#_Toc421883725)

[4.3 Comparison 2: Scalability, Flexibility and Faceted Filtering 7](#_Toc421883726)

[5 Recommendation 8](#_Toc421883727)

[6 Assumptions 9](#_Toc421883728)

[7 Risks 10](#_Toc421883729)

[8 Appendix 11](#_Toc421883730)

[8.1 References 11](#_Toc421883731)

# Introduction

This document provides a comparative analysis between ElasticSearch and Solr to enable faster and more efficient product searches on the e-commerce platform. Both technologies are widely used for implementing search functionality and are critical for improving user experience by delivering relevant results with minimal latency.

## Objective and scope of document

The objective of this document is to analyze the suitability of ElasticSearch and Solr for the e-commerce product search functionality. The scope includes evaluating both technologies based on performance, scalability, flexibility, and relevance ranking, leading to a recommendation on which technology will offer the best overall search experience.

# Requirements at a Glance

* **Fast Search Performance:** The search engine should deliver low-latency results even during high-traffic periods.
* **Scalability:** The system should be able to scale horizontally to handle increasing amounts of product data and user requests.
* **Faceted Search & Filtering:** Users should be able to filter search results based on product attributes.
* **Relevance and Ranking:** The search engine should return relevant products based on keywords, descriptions, and user intent.
* **High Availability:** The system should ensure minimal downtime, providing high availability even during infrastructure failures.

# Available tools

* ElasticSearch
* Solr

## ElasticSearch

ElasticSearch is a distributed, RESTful search engine built on top of the Apache Lucene library. It is designed for high-speed, full-text search and analytics across distributed datasets.

### Features

* Real-time indexing and search capabilities
* Horizontal scalability through sharding
* Supports full-text search, including stemming, synonyms, and fuzzy search
* Advanced analytics and aggregations
* RESTful API with JSON-based queries
* Cluster management, replication, and fault tolerance
* Strong community support and extensive plugins

### Pricing

* Open source with enterprise features available through Elastic (paid version for additional functionality like security and machine learning).

### Solr

Solr is an open-source search platform built on Apache Lucene, designed to handle large-scale search applications with structured and unstructured data. It is widely used for enterprise-level search solutions.

### Features

* Full-text search with faceted search and filtering
* Highly configurable schema with strong support for structured data
* Built-in support for replication and sharding (SolrCloud)
* Strong analytics capabilities with advanced faceted navigation
* Wide support for structured and semi-structured data
* REST-like API for search queries

### Pricing

* Open source, with most features available without a paid license.
* Infrastructure costs depend on the environment used for hosting (e.g., cloud, on-premises).

# Comparison Analysis

This section presents a comparative analysis of the features of ElasticSearch and Solr based on key performance metrics relevant to the e-commerce platform.

## Weightage Matrix

|  |  |
| --- | --- |
| Feature | Points |
| Performance | 5 |
| Scalability | 5 |
| Speed | 5 |
| Faceted Search and Filtering | 4 |
| Flexibility (Schema) | 4 |
| Ease of Setup | 3 |

## Comparison 1: Speed, Performance and Ease of Setup

|  |  |  |
| --- | --- | --- |
| Feature | ElasticSearch | Solr |
| Performance | **5** | **4** |
| Speed | **5** | **3** |
| Ease of Setup and Maintenance | **5** | **3** |

## Comparison 2: Scalability, Flexibility and Faceted Filtering

|  |  |  |
| --- | --- | --- |
| Feature | ElasticSearch | Solr |
| Scalability | **5** | **4** |
| Flexibility | **5** | **4** |
| Faceted Search & Filtering | **4** | **5** |

# Recommendation

Based on the comparative analysis, ElasticSearch is recommended for faster and better product searching on the e-commerce platform. Its superior performance in distributed environments, real-time indexing capabilities, and more flexible configuration make it a more robust choice for handling both unstructured product data and real-time search requirements. While Solr offers strong faceted search capabilities, ElasticSearch’s overall speed, ease of scaling, and better query flexibility make it the better option for this use case.

# Assumptions

1. The e-commerce platform requires real-time search capabilities to handle a large volume of products and dynamic user queries.
2. High availability and fault tolerance are key requirements.
3. The database will handle variety of unstructured data depending upon the products, necessitating flexible schema support.
4. Faster Retrival of data is the most important aspect of it.

# Risks

* **ElasticSearch**: Possible increased complexity in managing large clusters at scale without using managed services like Elastic Cloud.
* **Solr**: Greater manual effort required to configure and maintain large, distributed environments, potentially leading to slower deployment and scaling.

# Appendix

## References

1. https://www.elastic.co/
2. https://solr.apache.org/
3. https://logz.io/blog/solr-vs-elasticsearch/