

33주차 발표\_민서연

---

# Elasticsearch

---

# 01 Elasticsearch

## Elasticsearch

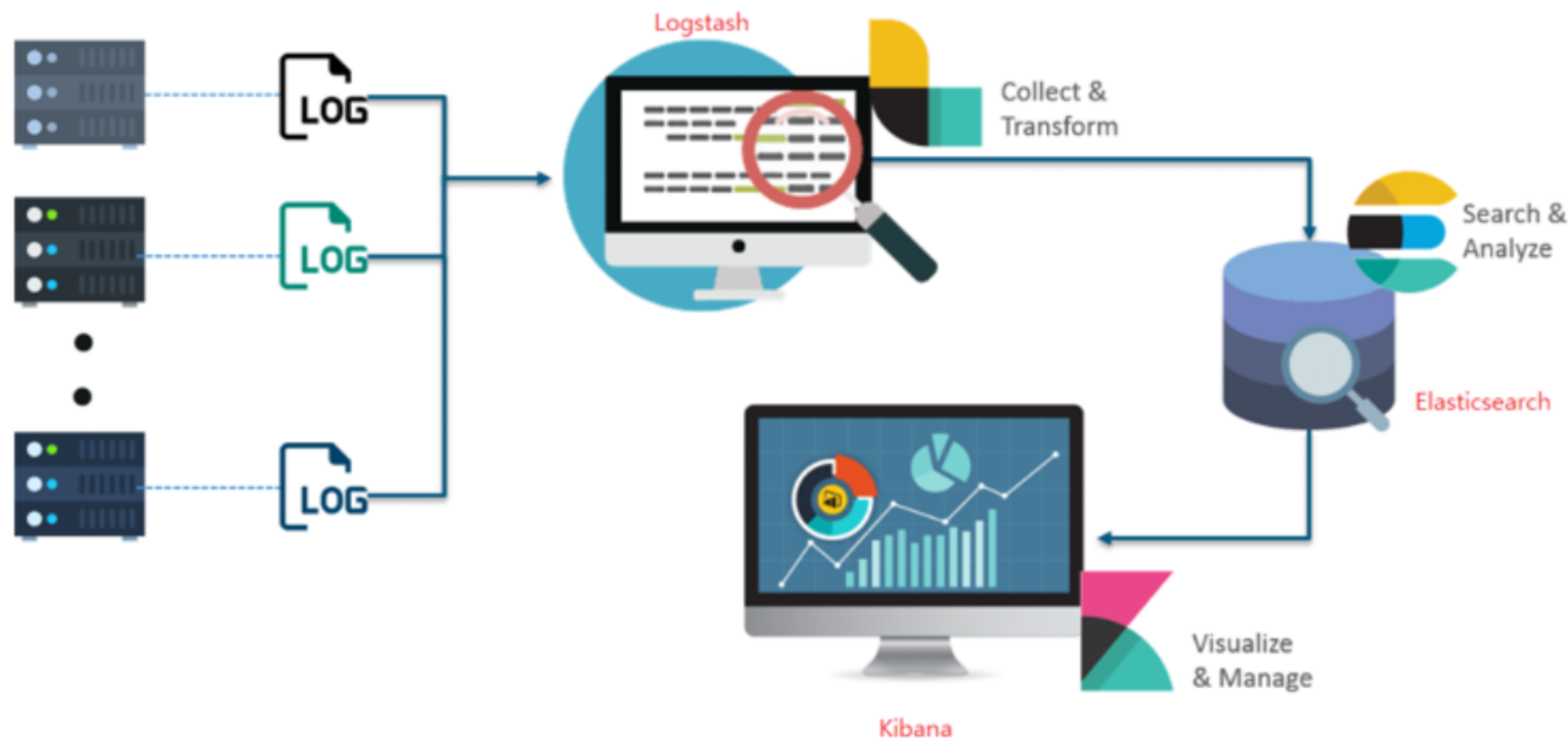
Elastic Stack의 핵심인 분산형 검색 및 분석 엔진

Apache Lucene( 아파치 루씬 ) 기반의 Java 오픈소스 분산 검색 엔진

루씬 라이브러리를 단독으로 사용할 수 있으며 방대한 양의 데이터를 신속하게 거의 실시간으로 저장, 분석, 검색 할 수 있다.

\* Apache Lucene( 아파치 루씬 )  
검색을 위한 라이브러리

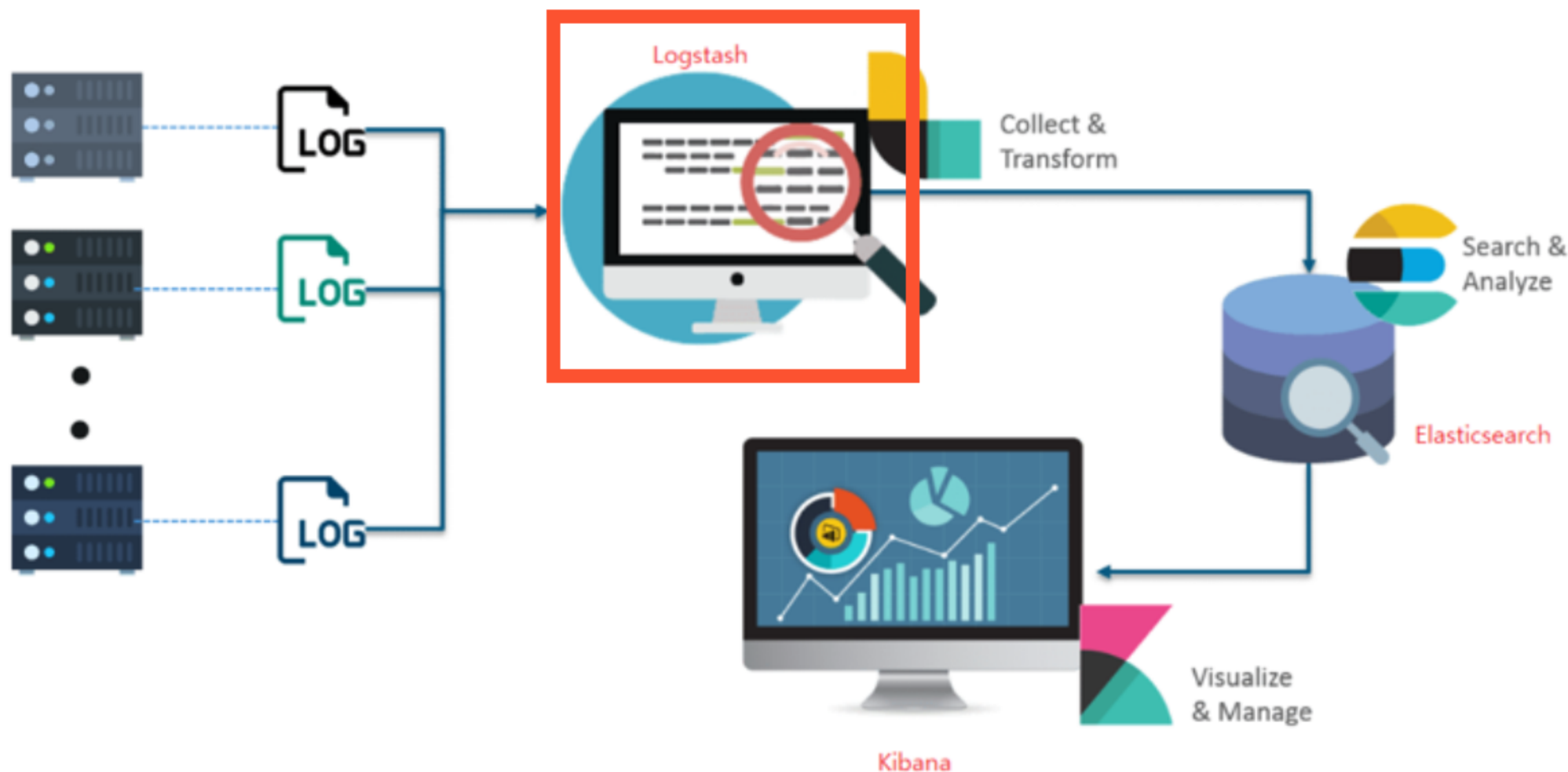
# 01 Elasticsearch



Elasticsearch는 검색을 위해 단독으로도 쓰이고 ELK( Elasticsearch / Logstatsh / Kibana )스택으로 사용 되기도 한다.

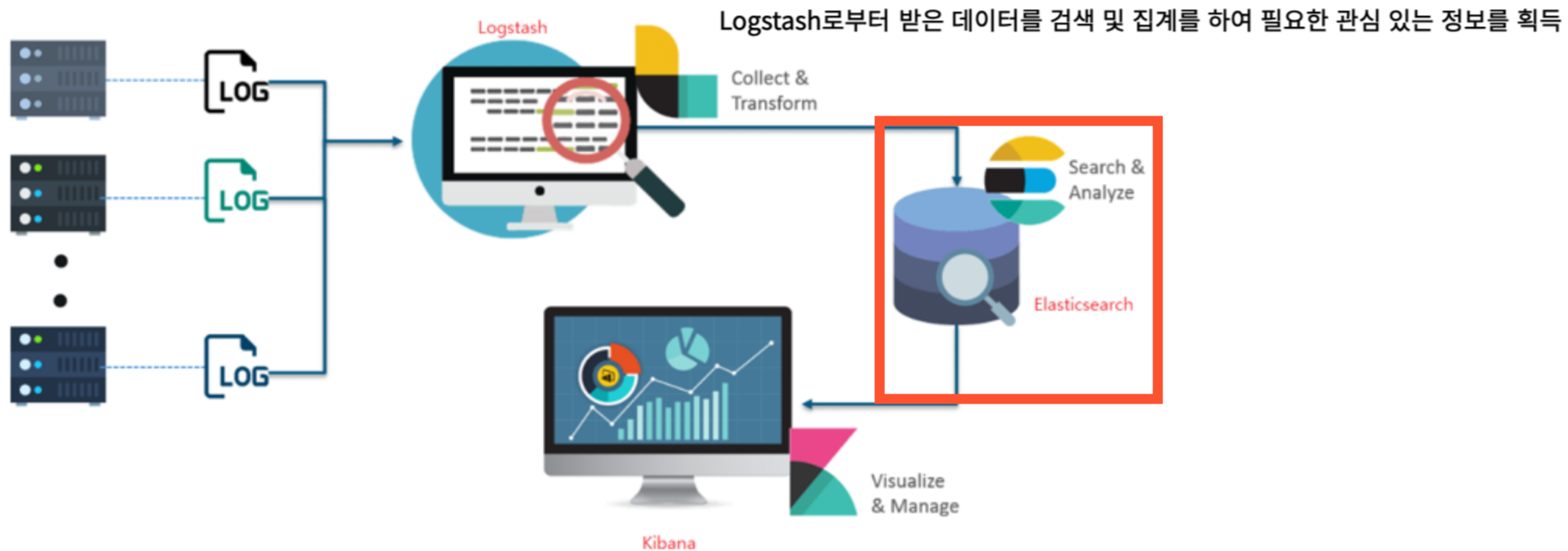
# 01 Elasticsearch

다양한 소스( DB, csv파일 등 )의 로그 또는 트랜잭션 데이터를 수집, 집계, 파싱하여 Elasticsearch로 전달



Elasticsearch는 검색을 위해 단독으로도 쓰이고 ELK( Elasticsearch / Logstatsh / Kibana )스택으로 사용 되기도 한다.

# 01 Elasticsearch



Elasticsearch는 검색을 위해 단독으로도 쓰이고 ELK( Elasticsearch / Logstatsh / Kibana )스택으로 사용 되기도 한다.

# 01 Elasticsearch




Elasticsearch의 빠른 검색을 통해 데이터를 시각화 및 모니터링

Elasticsearch는 검색을 위해 단독으로도 쓰이고 ELK( Elasticsearch / Logstatsh / Kibana )스택으로 사용 되기도 한다.

# 01 Elasticsearch

## Elasticsearch이 빠른 이유!

 inverted index( 역색인 )

index = 책의 목차

inverted index = 책 맨 뒤에 나와 있는 키워드 (찾아보기)

즉, Elasticsearch는 텍스트를 파싱해서 검색어 사전을 만든 다음, inverted index 방식으로 텍스트를 저장한다.



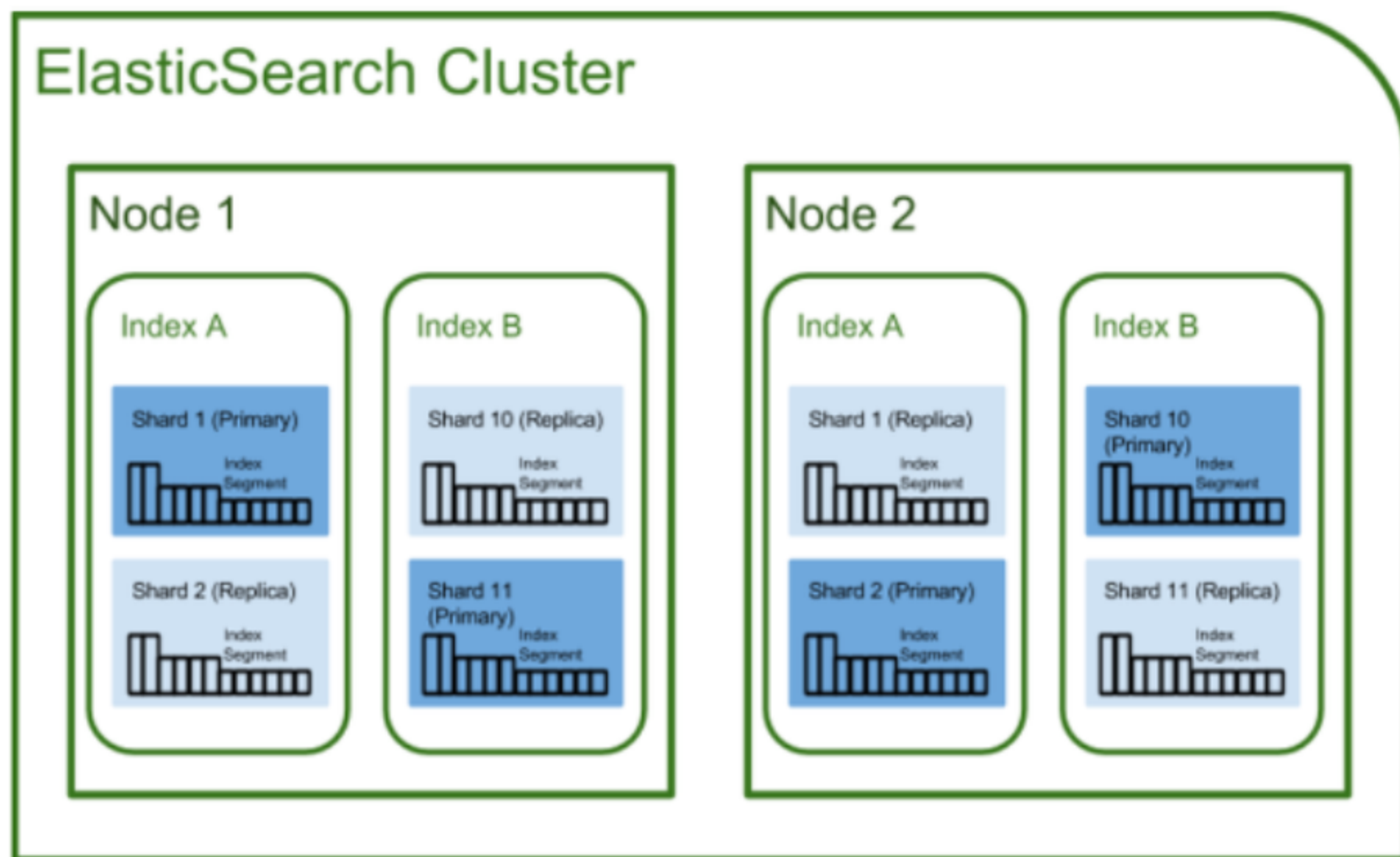
# 02 Elasticsearch와 관계형 DB



Relational Database	ElasticSearch
Database	Index
Table	Type
Row	Document
Column	Field
Index	Analyze
Primary key	_id
Schema	Mapping
Physical partition	Shard
Logical partition	Route
Relational	Parent/Child, Nested
SQL	Query DSL



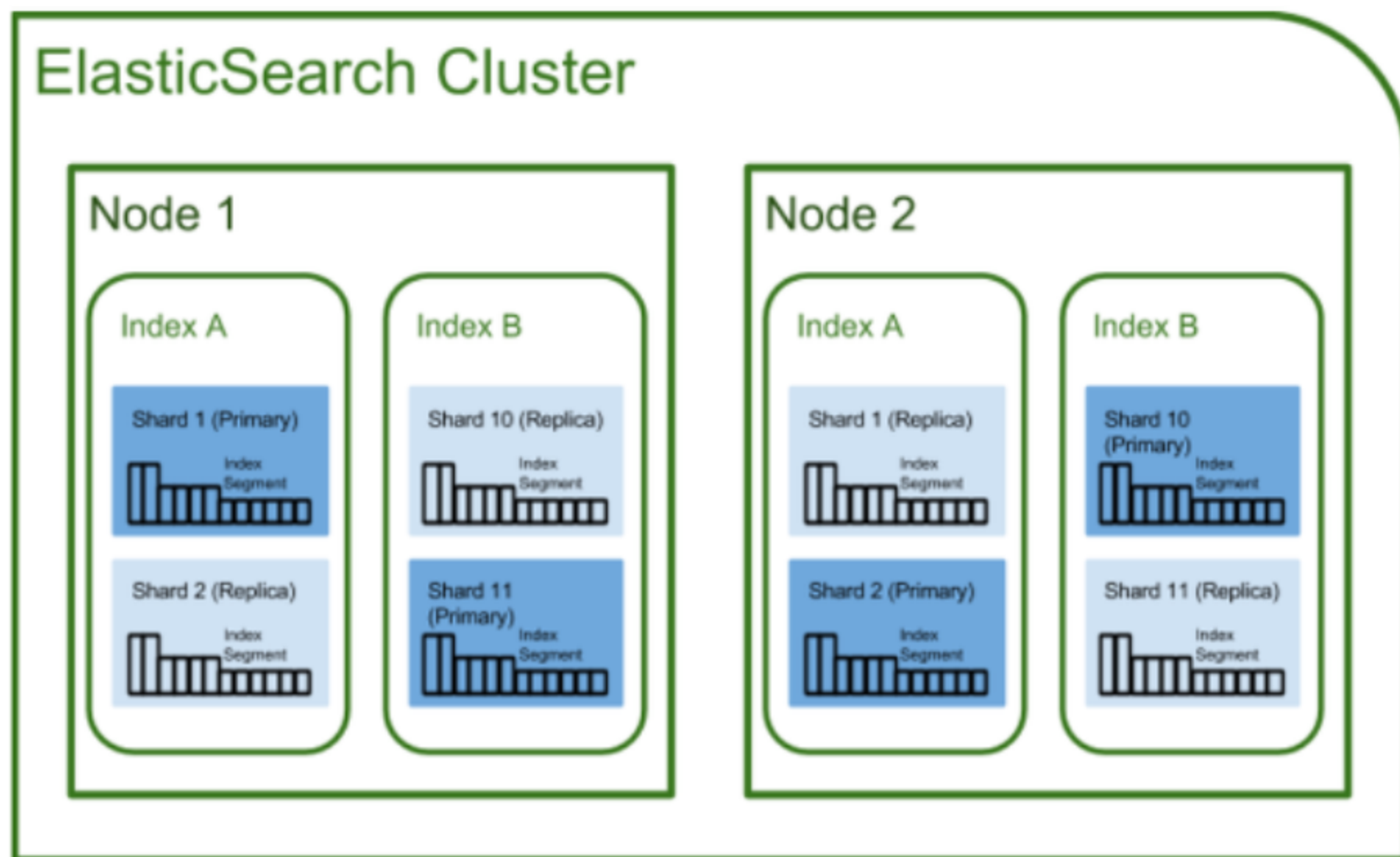
## 03 Elasticsearch 구조



### Cluster

- Elasticsearch에서 가장 큰 시스템 단위를 의미하며 최소 하나 이상의 노드로 이루어진 노드들의 집합
- 서로 다른 클러스터는 데이터의 접근, 교환을 할 수 없는 독립적인 시스템으로 유지 된다.
- 여러대의 서버가 하나의 클러스터를 구성할 수 있고 한 서버에 여러대의 클러스터가 존재할 수 있다.

## 03 Elasticsearch 구조



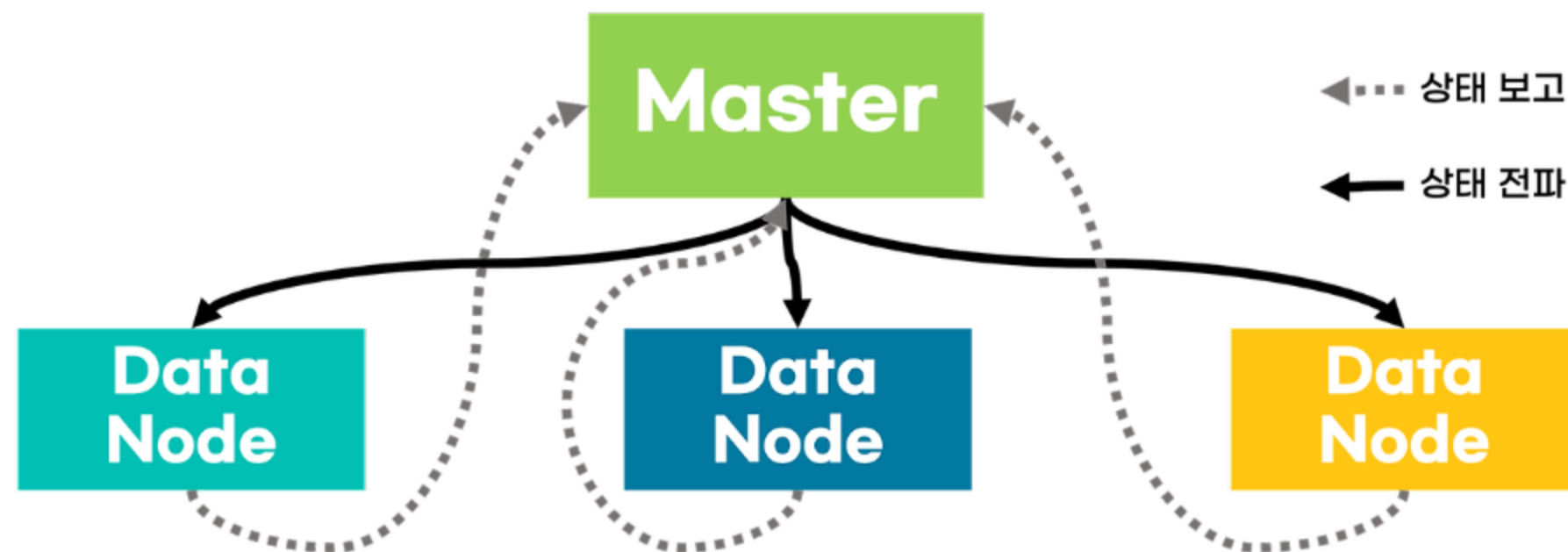
### Node

- Elasticsearch를 구성하는 하나의 단위 프로세스
- 역할에 따라 Master-eligible, Data, Ingest, Coordinating Only Node 노드로 구분할 수 있다.

## 03 Elasticsearch 구조

### Node

- Master-eligible  
: 클러스터를 제어하는 마스터로 선택할 수 있는 노드  
(인덱스 생성, 삭제 / 클러스터 노드들의 추적, 관리 /  
데이터 입력 시 어느 샤드에 할당할 것인지)



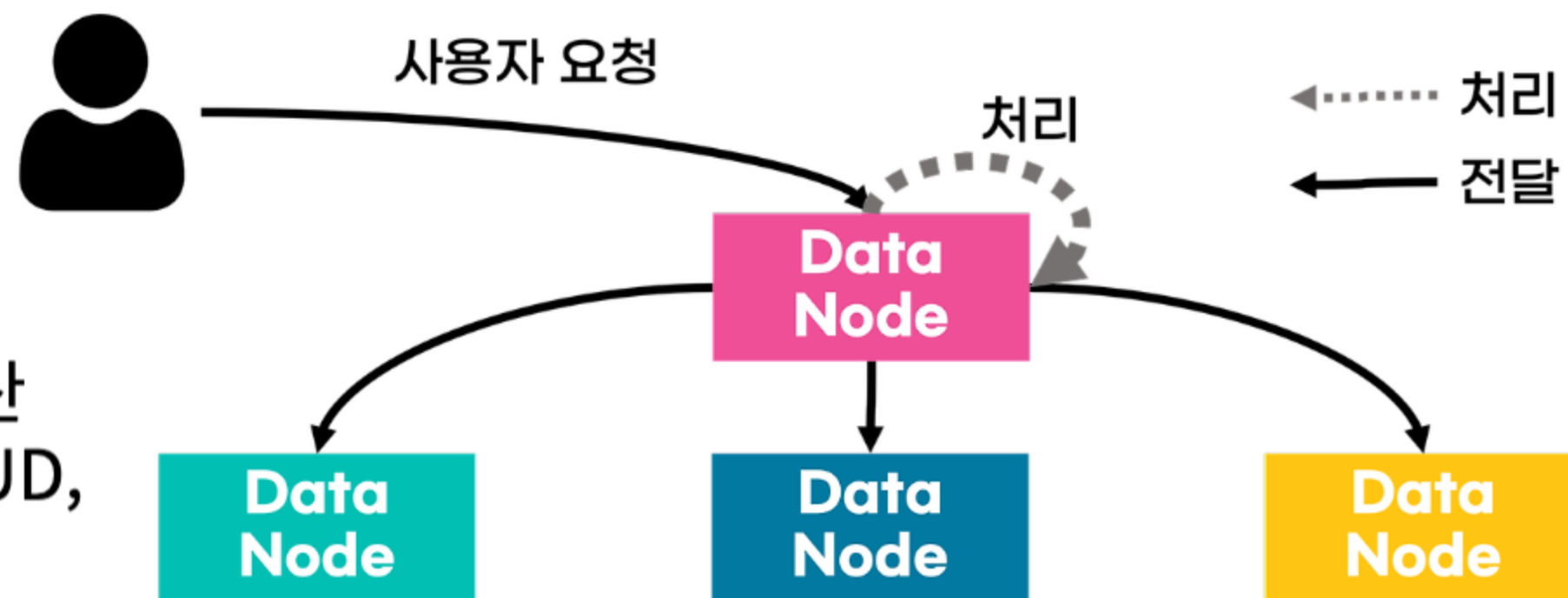
## 03 Elasticsearch 구조

### Node

- Data Node

: 사용자가 색인한 문서를 저장하고 검색 요청을 처리해 결과를 반환하는 역할을 수행하는 노드

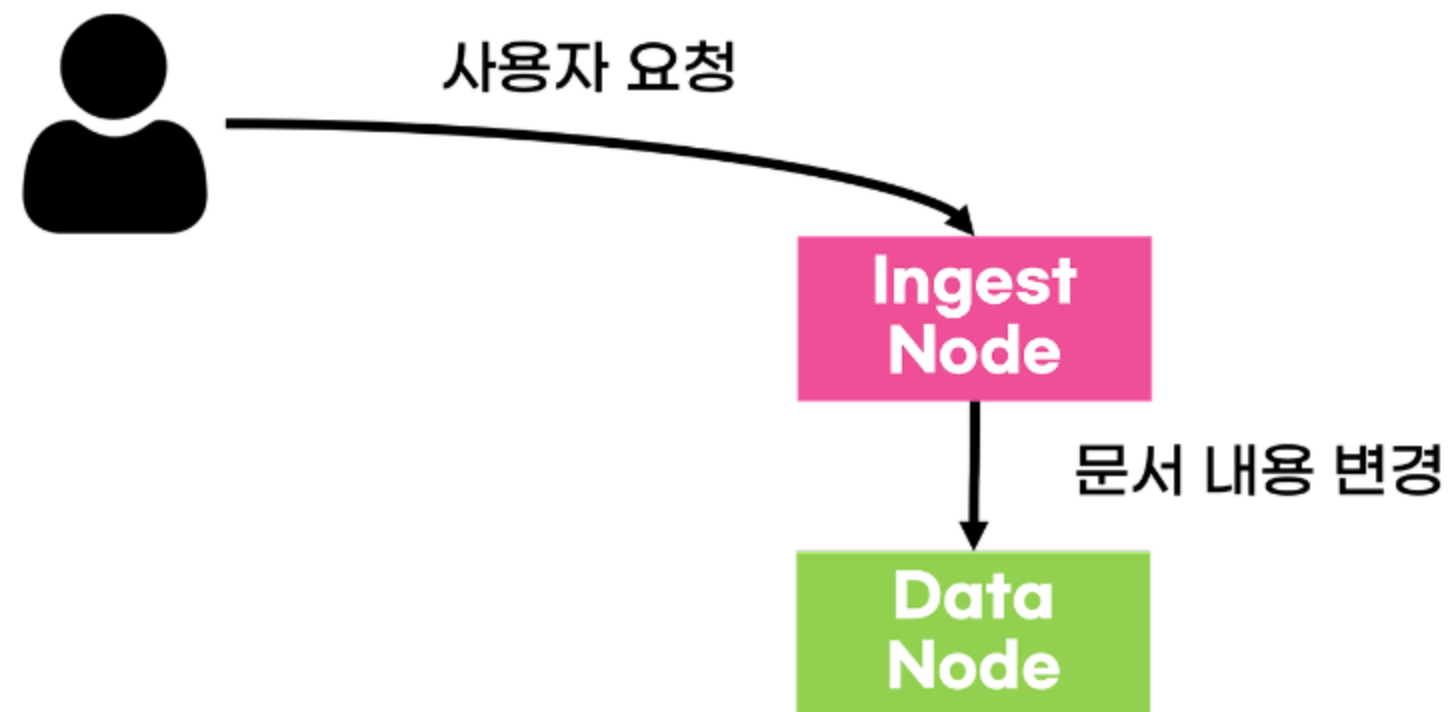
색인된 문서는 샤드 단위로 각 데이터 노드에 분산 저장되고 데이터 노드에 의해 데이터에 대한 CRUD, 검색, 집계 연산 등을 수행할 수 있다



## 03 Elasticsearch 구조

### Node

- Ingest Node  
: 일종의 데이터 전처리 파이프라인 역할을 수행하는  
노드

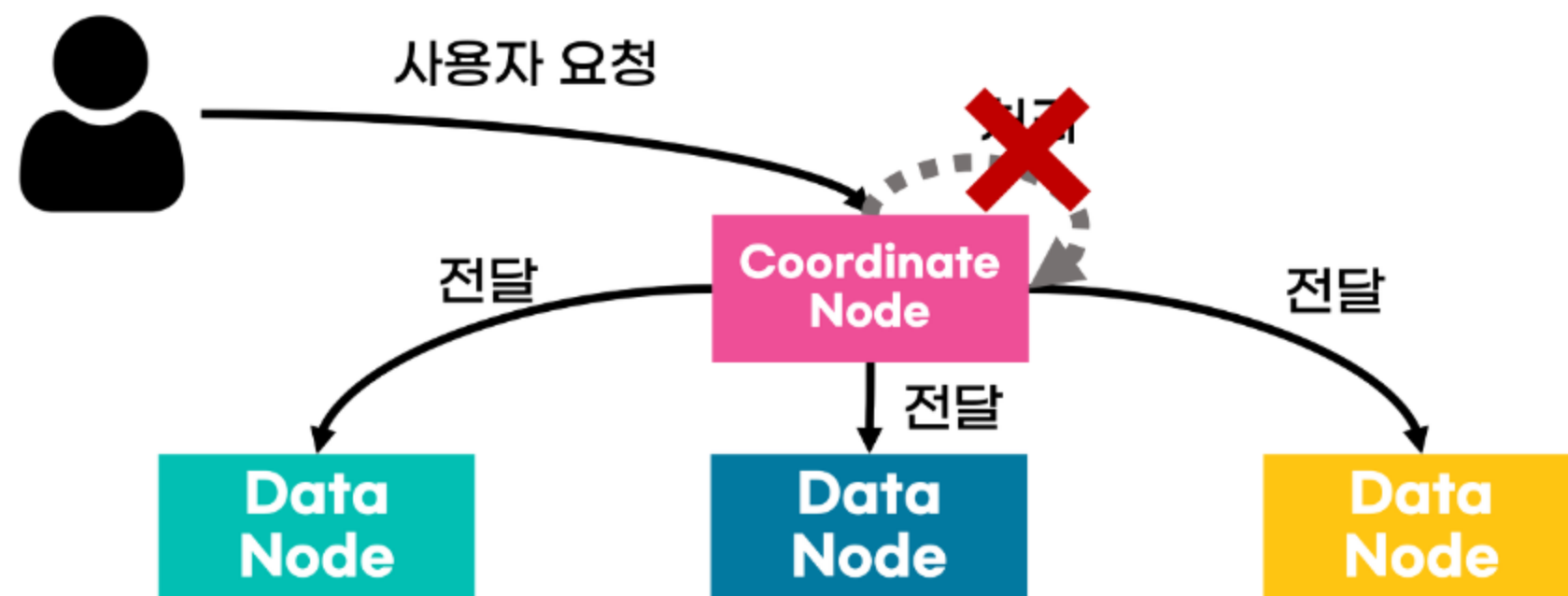


## 03 Elasticsearch 구조

### Node

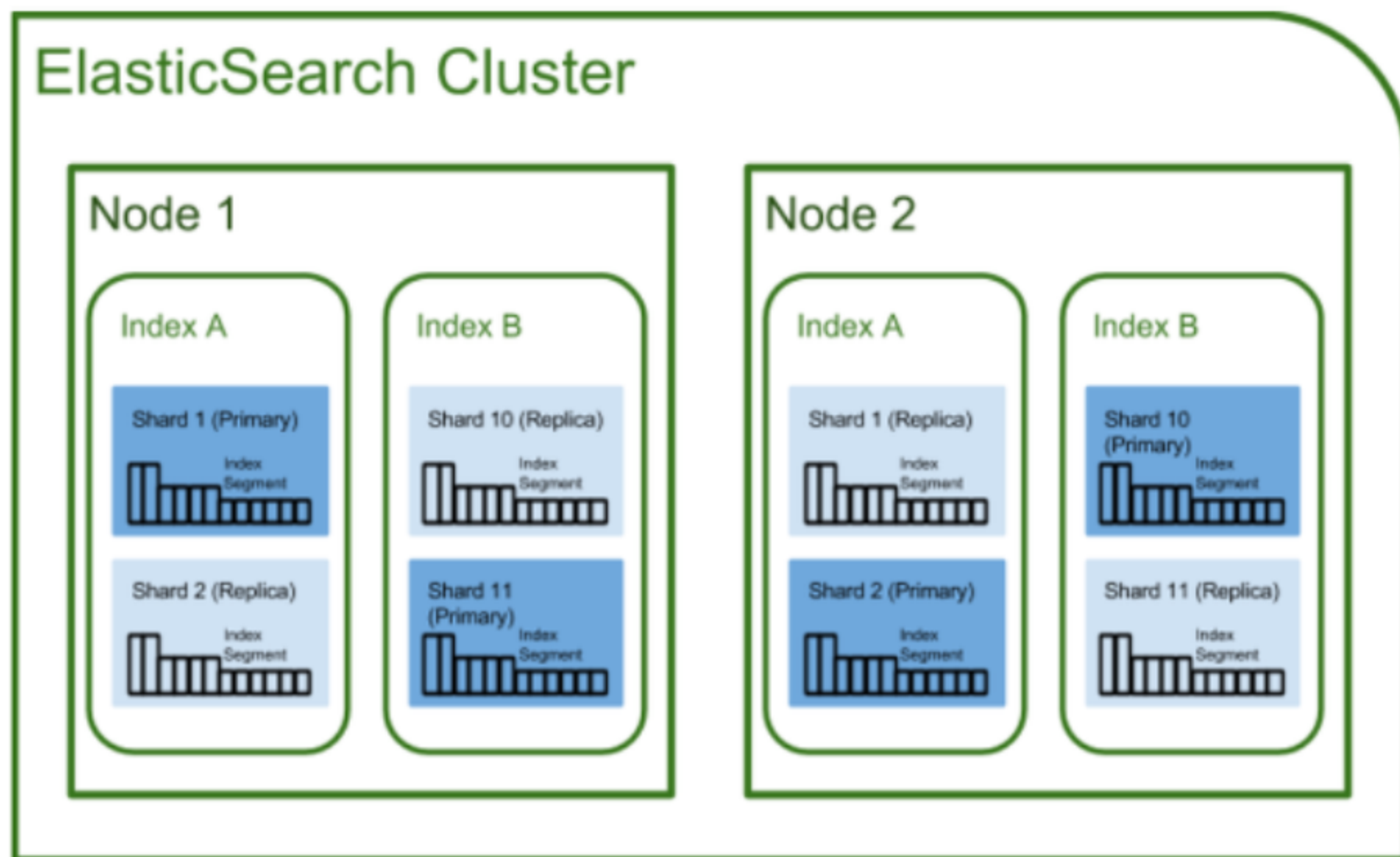
- **Coordinating Only Node**  
: 데이터 노드처럼 실제 데이터를 저장하거나 사용자의 요청에 대해 요청을 수행하진 않지만, 요청들에 대해 전달과 그 결과를 취합하는 역할을 하는 노드

로드 밸런서와 같은 역할





## 03 Elasticsearch 구조



### Index / Shard / Replica

- index는 RDBMS에서 database와 대응하는 개념
- 샤딩( sharding )은 데이터를 분산해서 저장하는 방법
- replica는 또 다른 형태의 shard.  
노드를 손실했을 경우 데이터의 신뢰성을 위해 샤드들을 복제



# 03 Elasticsearch 특징

- Scale out

샤드를 통해 규모가 수평적으로 늘어날 수 있다.

- 고가용성

Replica를 통해 데이터의 안정성을 보장한다.

- Schema Free

Json 문서를 통해 데이터 검색을 수행하므로 스키마 개념이 없다.

- Restful

데이터 CRUD 작업은 HTTP Restful API를 통해 수행한다.

Data CRUD	Elasticsearch Restful
SELECT	GET
INSERT	PUT
UPDATE	POST
DELETE	DELETE

## 04 출처

<https://www.elastic.co/guide/en/elasticsearch/reference/current/elasticsearch-intro.html>  
<https://victorydntmd.tistory.com/308>  
<https://tourspace.tistory.com/237>  
<https://www.edureka.co/blog/elk-stack-tutorial/>  
<https://www.slideshare.net/devview/2d1elasticsearch>  
<https://github.com/exo-archives/exo-es-search>  
<https://jeongxoo.tistory.com/12>