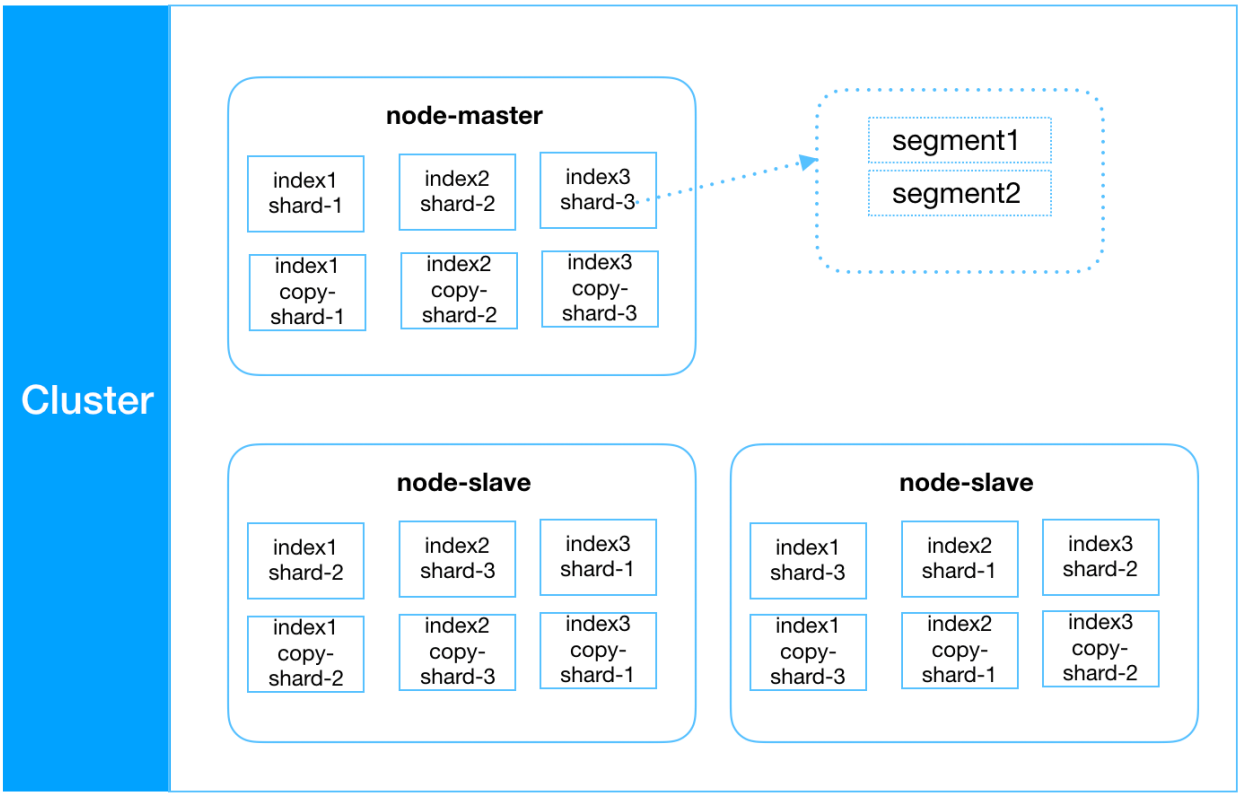
简介

> ElasticSearch是一个高度可扩展的开源全文搜索和分析引擎。它允许您快速、近实时地存储、搜索和分析大量数据。它通常被用作驱动具有复杂搜索功能和需求的应用程序的底层引擎/技术。适用于需要大量存储、快速检索、智能分析、复杂聚合查询和可视化要求的业务。

架构

ES 在架构上主要分为**集群、节点、索引、分片、段**这五层结构。集群(cluster)包含了若干个 ES 节点，节点(node)角色分为 master 与 slave；一个节点包含了若干个索引的部分数据，索引(index)可类比于关系型数据库中的表；一个索引包含若干个分片，ES 7.X 之前默认为5个分片，ES 7.X之后默认为1个分片；一个分片(shard)是一个完整的 lucene Index，其中包含了若干个段（segment）；段中则包含着 ES 最底层的数据结构如**倒排索引、docValue、stored field、cache** 等 。架构图如下所示：



生产中设计方案

采用 ES 生态的方法根据不同的应用场景会有不尽相同的方案，但是大体上的模式为 **数据源+中间件+数据传输+ES** 的方案，比如在集中式日志平台中通常会采用 **Filebeat+Kafka+Logstash+ES 的方案。**为了保证集群的高可用，通常在集群的规划上

会构建主备两个集群，利用中间件将数据双写到两个集群，这样一旦主集群出现故障可以立马切换到备用集群，保证服务不被中断。历史数据的不断增加，会导致集群的存储成本不断增长和搜索性能的下降，而在实际应用中通常近期数据访问频繁，而过去数据很少访问，因此在数据的存储上会根据时间将数据区分为冷数据和热数据，采用不同的配置方案达到提高搜索性能并且降低存储成本的目的。如下图所示：

集群的监控

为了实时掌控 ES 集群的健康状况，需要能够对 ES 集群进行监控，监控的层面包括 **集群、节点、索引、分片以及segment**，ES 本身提供了什么全面的监控 API，通过调用这些 API 可以很直观的了解到集群是否健康，节点内存使用是否过高，索引的文档数量是否达到极限等。但是这些通常无法满足实际的需求，比如某个索引文档写入是否出现突峰或者低谷，文档是否出现了新增字段等。这是我们需要自主开发或者引入第三方开源的监控组件，比较主流的有 **Yelp** 开发的 **Elastalert**，其设计的告警规则如下：

```
frequency：在 Y 时间里匹配事件高于 X 个；
spike：事件的增长率与下降率是否在阈值范围内；
flatline： 在 Y 时间内匹配事件低于 X 个；
blacklist/whitelist：匹配字段是否属于黑名单或白名单；
change：匹配字段在一段时间内是否改变；
new_term：匹配新出现的字段
cardinality：匹配某一字段在一段时间内出现不同值是否在一定的阈值范围内
metric_aggregation：匹配一段时间窗口内的聚合指标是否在阈值范围内
```

这些规则通常能够满足大部分的监控场景需求，对于个性化的需求，也可以通过其提供的开发文档进行定制化开发。