**搜索 / 分析**

Elasticsearch是Elastic Stack核心的分布式搜索和分析引擎。Logstash和Beats有助于收集,聚合和丰富您的数据并将其存储在Elasticsearch中。使用Kibana，你可以交互式地探索，可视化和共享对数据的见解，并管理和监视堆栈。

Elasticsearch是建立索引。搜索和分析魔术的地方。

Elasticsearch为所有类型的数据提供实时搜索和分析。无论是结构化文本还是非结构化文本，数字数据或地理空间数据，Elasticsearch都能以支持快速搜索的方式有效地对其进行存储和索引。您不仅可以进行简单的数据检索，还可以汇总信息来发现数据中的趋势和模式。随着数据和查询量的增长，Elasticsearch的分布式特性使你的部署可以随之无缝地增长。

提供快速和灵活性来处理各种用例中的数据：

* 将搜索框添加到应用或网站；
* 存储和分析日志，指标和安全事件数据；
* 使用机器学习自动实时建模数据行为；
* 使用Elasticsearch作为存储引擎来自动化业务工作流程；
* 使用Elasticsearch作为地理信息系统（GIS）管理，集成和分析空间信息；
* 使用Elasticsearch作为生物信息学研究工具来存储和处理遗传数据。

在Elasticsearch中处理数据，文档和索引的方式都是相同的。

#### 1、数据输入：文档和索引

Elasticsearch是一个分布式文档存储。Elasticsearch不会将信息存储为列数据的行，而是存储已序列化为JSON文档的复杂数据结构。当集群中有多个Elasticsearch节点时，存储的文档会分布在整个集群中，并且可以从任何节点立即访问。

存储文档时，将在1秒内几乎实时地对其进行索引和完全搜索。Elasticsearch使用称为倒排索引的数据结构，该结构支持非常快速的全文本搜索。反向索引列出了出现在任何文档中的每个唯一单词，并标识了每个单词出现的所有文档。

索引可以认为是文档的优化集合，每个文档都是字段的集合，这些字段是包含数据的键值对。默认情况下，Elasticsearch对每个字段中的所有数据建立索引，并且每个索引字段都具有专用的优化数据结构。例如，文本字段存储在倒排索引中，数字字段和地理字段存储在BKD数树中。使用按字段数据结构组合并返回搜索结果的能力使Elasticsearch如此之快。

Elasticsearch还具有无模式能力，这意味着可以为文档建立索引，而无明确指定如何处理文档中可能出现的每个不同字段。启用动态映射后，Elasticsearch自动检测并将新字段添加到索引。此默认行为使索引和浏览数据变得容易-只需开始建立索引文档，Elasticsearch就会检测布尔值，浮点数和整数值，日期和字符串并将其映射到适当的Elasticsearch数据类型。

可以定义规则以控制动态映射，并显式定义映射到适当的Elasticsearch数据类型。

定义自己的映射的优点：

* 区分全文字符串字段和精确值字符串字段；
* 执行特定于语言的文本分析；
* 优化字段以进行部分匹配；
* 使用自定义日期格式；
* 使用无法自动检测到的数据类型，例如geo\_point和geo\_shape

为不同目的以不同的方式对同一字段建立索引通常很有用。例如，可能希望将一个字符串字段索引为全文搜索的文本字段，以及作为排序或汇总数据的关键字字段。或者，您可能选择使用多个语言分析器来处理包含用户输入的字符串字段的内容。

在搜索时也会使用在索引期间应用于全文字段的分析链。当您查询全文字段时，对查询文本进行相同的分析，然后再在索引中查找术语。

#### 2、信息输出：搜索和分析

轻松访问基于Apache Lucence搜索引擎库构建的全套搜索功能。

Elasticsearch提供了一个简单，一致的REST API，用于管理您的集群以及索引和搜索数据。为了进行测试，您可以直接从命令行或通过Kibana中的开发者控制台轻松提交请求。在您的应用程序中，您可以将Elasticsearch客户端用于您选择的语言：Java、JavaScript、Go、.NET、PHP、Perl、Python或Ruby。

**（1）搜索数据**

Elasticsearch REST API支持结构化查询，全文查询以及结合了两者的复杂查询。结构化查询类似于您可以在SQL中构造的查询类型。例如，您可以搜索索引中的gender和age字段，employee按hire\_date字段对匹配项进行排序。全文查询会找到所有与查询字符串匹配的文档，并按相关性对它们进行归还-它们与你的搜索词的匹配程度如何。

除了搜索单个术语外，你还可以执行短语搜索，相似性搜索和前缀搜索，并获得自动完成建议。

是否要搜索地理空间或其他数字数据？Elasticsearch在支持高性能地理和数字查询的优化数据结构中索引非文本数据。

你可以使用Elasticsearch全面的JSON样式查询语言（Query DSL）访问所有这些搜索功能。您还可以构造SQL样式的查询，以在Elasticsearch内部本地搜索和聚合数据，并且JDBC和ODBC驱动程序使范围广泛的第三方应用程序可以通过SQL与Elasticsearch进行交互。

**（2）分析数据**

Elasticsearch聚合使你能够构建数据的复杂摘要，并深入了解关键指标，模式和趋势。通过汇总，你可以找到众所周知的“大海捞针”。

由于聚合利用了用于搜索的相同数据结构，因此它们也非常快。这使您可以实时分析和可视化数据。您的报告和仪表板会随着数据的更改而更新，因此您可以根据最新信息采取措施。

而且，聚合与搜索请求一起运行。你可以在单个请求中同时对相同数据搜索文档，过滤结果并执行分析。

#### 3、可扩展性和弹性：集群、节点和碎片

Elasticsearch旨在始终可用，并可以根据您的需求进行扩展。它是通过自然分布来实现的。您可以将服务器（节点）添加到集群以增加容量，Elasticsearch会自动在所有可用节点之间分配数据和查询负载。无需大修您的应用程序，Elasticsearch知道如何平衡多节点集群以提供扩展性和高可用性。节点越多越好。

这是如何运作的？在幕后，Elasticsearch索引实际上只是一个或多个物理碎片的逻辑分组，其中每个碎片实际上是一个独立的索引。通过在多个分片之间的索引中分配文档，并在多个节点之间分配这些分片，Elasticsearch可以确保冗余，这既可以防止硬件故障，又可以在将节点添加到集群中时提高查询能力。随着集群的增长（或收缩），Elasticsearch会自动迁移碎片以重新平衡集群。

分片有两种类型：主数据库和副本数据库。索引中的每个文档都属于一个主分片。副本分片是主分片的副本。副本可提供数据的冗余副本，以防止硬件故障并增加处理读取请求（如搜索或检索文档）的能力。

创建索引时，索引中主碎片的数量是固定的，但是副本碎片的数量可以随时更改，而不会中断索引或查询操作。

**（1）这取决于......**

在分片大小和为索引配置的主分片数量方面，存在许多性能方面的考虑和权衡取舍。分片越多，维护这些索引的开销就越大。分片大小越大，当Elasticsearch需要重新平衡集群时，分片移动所需的时间就越长。

查询很多小的分片会使每个分片的处理速度更快，但是更多的查询意味着更多的开销，因此查询较小数量的大分片可能会更快。简而言之...要视情况而定。

作为起点：

* 旨在将平均分片大小保持在几GB到几十GB之间。对于具有基于时间的数据的用例，通常会看到20GB到40GB范围之内的碎片。
* 避免庞大的碎片问题。节点可以容纳的分片数量与可用对空间成比例。通常，每GB堆空间中的分片数量应少于20。

确定用例最佳配置的最佳方法是通过使用自己的数据和查询进行测试。

**（2）在灾难情况下**

出于性能原因，群集内的节点必须位于同一网络上。跨不同数据中心中的节点在群集中平衡碎片的时间太长了。但是高可用性架构要求您避免将所有鸡蛋都放在一个篮子里。如果一个位置发生重大故障，则另一个位置的服务器需要能够接管。无缝地。答案？跨集群复制（CCR）。

CCR提供了一种方法，可以自动将索引从主群集同步到可以用作热备份的辅助远程群集。如果主群集发生故障，则辅助群集可以接管。您还可以使用CCR创建辅助群集，以接近地理位置的方式向用户提供读取请求。

跨集群复制是主动-被动的。主群集上的索引是活动的领导者索引，并处理所有写请求。复制到辅助群集的索引是只读跟随者。

**（3）**

与任何企业系统一样，您需要工具来保护，管理和监视Elasticsearch集群。集成到Elasticsearch中的安全性，监视和管理功能使您可以将Kibana 用作控制中心来管理集群。类似的特征数据汇总和指标生命周期管理 可帮助您明智随着时间的推移管理您的数据。