|  |
| --- |
| 外文翻译 |
| **ElasticSearch**  **处理大量数据的高级快速搜索技术**  ·  **一、介绍**  Elasticsearch 是一个新的数据库，用于以非常高的可用性处理大量数据量，并在许多计算机上进行分布，使其容错且可扩展，同时保持一个简单但功能强大的 API，允许应用程序从对数据库的任何语言或框架访问。构建在 APACHE 的 LUCENE 之上的横向可扩展分布式数据库，通过简单而强大的 API 提供跨 TB 数据的完整搜索体验。许多公司使用ElasticSearch来帮助他们在应用程序中部署功能强大的搜索功能，这些功能易于设置、扩展和构建于云中。  **关于Lucene**  Apache 的 Lucene 是一个用于文本搜索的开源 Java 库。Lucene 项目已经发展了十多年，现已成为如何构建强大而易于集成的开源搜索库的标准参考。Lucene 作为搜索库，必须用接口包装，以便应用程序使用其功能。许多此类接口都是为不同的平台和用例（例如 SOLR）构建的。  但是，像 SOLR 这样的接口是为单个服务器可以处理索引和查询数据的全部工作负载而设计的。当数据量开始超过限制时，SOLR（和与 Lucene 类似的接口）变得难以使用：RDBMS 系统中发生的分片、复制和查询调度同样的问题在此上下文中开始出现。正如在RDBMS世界中存在各种处理这些困难的方法一样，在SOLR周围存在各种分片创建和分发的工具。  但是，正如大数据数据库的正确解决方案意味着从 RDBMS 迁移到 NoSQL 技术一样，扩展 Lucene 的正确解决方案是远离 SOLR 等工具，使用从头开始构建的工具，在水平内处理 TB 的数据可扩展、分布式和容错方式：ElasticSearch！  **二、LUCENE = ElasticSearch=** **大数据**  ElasticSearch最好被认为是Lucene的接口，专为BIGDATA设计， 从头开始。Lucene 为搜索数据提供的复杂功能集可通过ElasticSearch直接获得，因为 Lucene 最终是用于索引和查询数据的库。这也意味着，与 Lucene 一起工作的插件将在开箱即用的ElasticSearch中工作。ElasticSearch本身在 Lucene 周围提供的功能旨在使其成为大数据全文搜索的完美工具。  **三、了解ElasticSearch**  *A. 基本功能*  1） REST API：ElasticSearch通过 REST API 存储/检索对象。提供方便的 PUT、开机、获取和删除 API，实现版本检查（可选在 PUT 上），生成 ID（可选在开机自检），并允许您读取自己的写入（在 GET 上）。这就是它成为关键价值存储的原因。  2） 键值存储：在ElasticSearch中，每一段数据都有定义的索引和类型。可以将索引视为文档或数据库中的表的集合。但是，此处添加到索引的文档没有定义的结构和字段类型。对象具有类型，并转到索引中。因此，从 REST 的角度来看，任何对象的相对 uri 是 /{索引}/[type]/{id}。通过 REST API 在运行时创建索引和类型。  3） 多租户：您可以创建、更新、检索和删除索引。您可以根据每个索引配置分片和复制。这意味着ElasticSearch是多租户和相当灵活的。  4） 映射：ElasticSearch索引您存储的文档使用动态映射或您提供的映射（推荐）。这意味着您也可以通过搜索 API 查找回文档。分片和复制：通过副本（索引部件的副本）实现更好的可用性和性能。  *B. ElasticSearch中的搜索 API*  这是Lucene进来的地方。与 GET 不同，搜索不允许您立即读取自己的写入，因为索引需要时间来更新，并且批量复制和执行此操作的效率更高。  搜索 API 作为\_search资源公开，这些资源在服务器级别 （/\_search）、索引级别（//\_search）或类型级别(/{index}/{type}/\_search）中可用。因此，您可以跨多个索引进行搜索，因为 ElasticSearch 也跨多台计算机进行复制和分片。  返回搜索结果时，ElasticSearch 在结果集中包含一个\_source字段，该字段默认包含与结果关联的对象。这意味着查询就像执行多获取，即如果文档很大、查询昂贵且结果集很大，则查询费用昂贵。这意味着您必须仔细管理查询数据集的方式。  搜索 API 支持 GET 和 POST 方法。Post 作为不允许 json 正文作为 GET 请求一部分的客户端的备份而存在。您需要一个原因是 ElasticSearch 提供了特定于域的语言（当然，基于 json）来指定复杂的查询。您还可以在 GET 请求中使用 Lucene 查询语言与 q\_query 参数，但它的功能要小得多，并且只对简单内容有用。  *C. ElasticSearch中的群集*  默认情况下，ElasticSearch是群集的。这意味着，如果在同一网络中启动两个节点，它们将连接并成为群集。这不需要任何特殊配置。  ElasticSearch可以作为独立的单搜索服务器工作。然而，为了能够处理大数据集并实现容错，它可以在许多合作服务器上运行。总之，这些服务器称为 CLUSTER，每个服务器都称为节点。  通过索引分片（将其拆分为较小的单个部分）和 ElasticSearch 跨网络中任何可用的节点复制大量数据。通常，您以不同的方式配置它，以便在不同的环境中运行。  ElasticSearch专为亚马逊 AWS、Heroku 或您自己的数据中心等大型部署而构建。这意味着它附带了内置的监控功能、可插拔的体系结构，用于适应不同的环境，以及您需要在此类环境中运行的许多其他内容。这与SOLR形成了很好的对比，SOLR不会开箱即用地做这些事情。  *应用程序支持*  ElasticSearch不仅仅是一个搜索引擎，它不仅仅是一个搜索引擎。它是一个成熟的数据库，您可以在它之上构建整个前端应用程序。  ElasticSearch支持每个索引的多个索引（数据库）和多个映射（表）。此功能与ElasticSearch允许的复杂文档结构相结合，允许您构建支持应用程序的复杂数据模型。  此外，除了能够跨数据执行丰富的搜索查询外，Elasticsearch 还允许定义应用程序数据库的更"传统"操作：列出记录、创建记录、更新记录和删除记录。这些功能提供了在同一数据库之上构建传统数据库驱动的读/写应用程序所需的功能，该数据库允许您执行全文搜索和复杂查询，所有这些都具有从一开始内置的水平可扩展性。  *D. 通过 HTTP 与 JSON 合作*  JSON 通过 HTTP 实际上已成为系统中服务的"语言"（桥接语言）。尽管ElasticSearch是一个Java解决方案，但通过 HTTP 的 JSON 使人们能够非常轻松地使用 Ruby、Perl 和其他语言进行开发。  HTTP 是线格式，JSON 是有效负载。它很容易被任何语言所消耗。此外，您现在可以在 JSON 中说明响应，并将其直接流式传输到浏览器。  当您提供ElasticSearch搜索请求时，还可以请求直方图，例如过去一年中每天的推文数。它返回的结构，准备被扔进任何图表库。不仅仅是 HTTP 和 JSON，它是数据或对象结构，这使得它易于使用。  *ElasticSearch中的索引*  ElasticSearch能够实现快速搜索响应，因为它不是直接搜索文本，而是搜索索引。  这就像通过扫描书籍背面的索引来检索与关键字相关的书籍中的页面，而不是搜索书籍每一页的每个单词。这种类型的索引称为倒索引，因为它将以页面为中心的数据结构（page-gt;words）反转为关键字为中心的数据结构（word>页）。  ElasticSearch使用 Apache Lucene 创建和管理此反转索引。  在ElasticSearch中，文档是搜索和索引的单位。索引由一个或多个文档组成，文档由一个或多个字段组成。在数据库术语中，文档对应于表行，字段对应于表列。  E. 查询 DSL  查询 DSL 是ElasticSearch 使 Lucene 的查询语法可供用户访问的方式，允许使用 JSON 语法组合复杂的查询。  查询的主要结构大致为：  卷曲 -X POST " http://localhost:9200/blog/\_search?pretty=true" -d  {"from": 0,  "size": 10,  "query" : QUERY\_JSON,  FILTER\_JSON,  FACET\_JSON,  SORT\_JSON }’  **四、ElasticSearch与其他开源搜索引擎**  ElasticSearch使数据探索变得非常简单。  首先，有一个强烈的原则或建筑概念，事情应该是简单和简单的。当人们开始设置、部署和使用ElasticSearch时，很容易设置 2、4、6、10 - 根据需要设置尽可能多的节点。设置群集也很容易。  开始开发时，可以轻松地开始使用文档作为 JSON 文档。通过该 API，使用不同的语言（如 Java、Ruby、Perl、Python 等）变得容易。在运行时，ElasticSearch管理分布 — 添加节点非常简单，数据会自动重新分配。  如果您将搜索视为一个过程，ElasticSearch将超越自由文本。无论表单如何，用户和开发人员都希望从他们的数据中得到有价值的信息。虽然ElasticSearch可以很好地进行自由文本搜索，但您还需要结构化搜索、分析、聚合、数据方面的分面等。所有这些都很好地绑在一起。例如，假设您正在索引 Twitter 上的社会字符串，就像我们的很多客户一样。您可以轻松设置它，并提出诸如"找到所有关于总统的推文"之类的问题。这是一堆推文的免费文本搜索。然后你可以说，"在上个月从爱达荷州发微博时，找到所有关于总统的推文。我们通过添加位置和时间段来在这里撒上更多的结构。然后，我们可以要求它做同样的事情，但把它分解成每天的推文数，以便随着时间的推移看到趋势。因此，我们最终有了一个指标和几个维度来显示有价值的东西。设置初始查询后，我们可以轻松地将总裁的姓名更改为任何其他人员，ElasticSearch可实时反映结果设置。然后，我们可以从一个人转变为"流感流行"等话题，结果集再次实时反映realtime。  对于开发人员来说，通过ElasticSearch，数据探索变得多么容易，这非常强大。  **五.、ElasticSearch基本流程**  工作流可以简要地解释如下：  •上传或储存文件；文件可以是任何类型、任何大小、任何数量。  •然后JSON构建器将这些文档从各自的类型转换为JSON文档。  •现在，标记器的职责是将数据分解成单个单词。  •对这些词进行索引，并进行映射，以便将类似类型的词分组为一种映射类型。这确保了根据用户触发的查询更快地检索文本。  •解析器将解析查询，并相应地从索引文档中搜索和检索搜索文本。  **六、ElasticSearch：公司！**  谢伊·巴顿于2004年创立了Compass。 在考虑指南针的第三个版本时，他意识到有必要重写指南针的大部分内容，以"创建一个可扩展的搜索解决方案"。因此，他创建了"一个从开始构建的解决方案来分发"，并使用了一个通用接口，JSON通过HTTP，也适合编程Java以外的语言。谢伊·巴顿于2010年2月发布了第一版ElasticSearch。  ElasticSearch，该公司背后的流行的实时搜索和分析开源项目，被强调由ItWorks，全球技术咨询公司，作为领先的去搜索平台在其两年一次的技术雷达报告。  由成千上万的技术领导者阅读，这个包含是ElasticSearch的验证，它迄今已有超过250万次下载。世界各地的公司都在使用 Elasticsearch 来探索和理解比其他解决方案更容易、更经济地了解大型数据集。  2013 年技术雷达报告认可了ElasticSearch，包括：  • **易用性**: Elasticsearch平台是一个可扩展、多租户、水平可扩展的搜索解决方案。它允许快速而简单地索引和检索复杂的数据结构。  • **易操作性**: 该平台提供了一个优雅的操作模型，可以自动发现集群中的对等点、故障转移和复制。Elasticsearch可以通过插件系统进行扩展，从中可以轻松添加新功能。  • **可信任的社区**: 围绕Elasticsearch开源工具的用户非常活跃，Java、C#、Ruby和JavaScript等语言提供的客户机库数量就说明了这一点。  **七. 摘要**  数据一直流入您的系统。问题是...这些数据能多快成为洞察？使用ElasticSearch，实时是唯一的时间。  ElasticSearch允许您从小开始，但会随着您的业务而增长。它构建为水平缩放开箱即用。当您需要更多容量时，只需添加更多节点，让群集重新组织自身以利用额外的硬件。  ElasticSearch使用Lucene封面下提供任何开源产品中最强大的全文搜索功能。搜索附带多语言支持、强大的查询语言、对地理位置的支持、上下文感知即用建议、自动完成和搜索代码段。  复杂的真实世界实体可以作为结构化的 JSON 文档存储在ElasticSearch中。默认情况下，所有字段都编制索引，并且所有索引都可以在单个查询中使用，以快速返回结果。  它不断演变，文本分析的推出未来几天将会有新版本出现。  它最终将预示着全文搜索和文本分析领域的新曙光。 |
|  |