ES (Elastic Search)

ES简介

- ES是一个基于Lucene的搜索服务器，提供了一个分布式多用户能力的全中文搜索引擎
- 分布式文档数据库，每个字段都可以被索引
- 基于RESTful web接口，所有交互基于Json调用
- 一个快速的搜索解决方案
- 零配置，完全免费

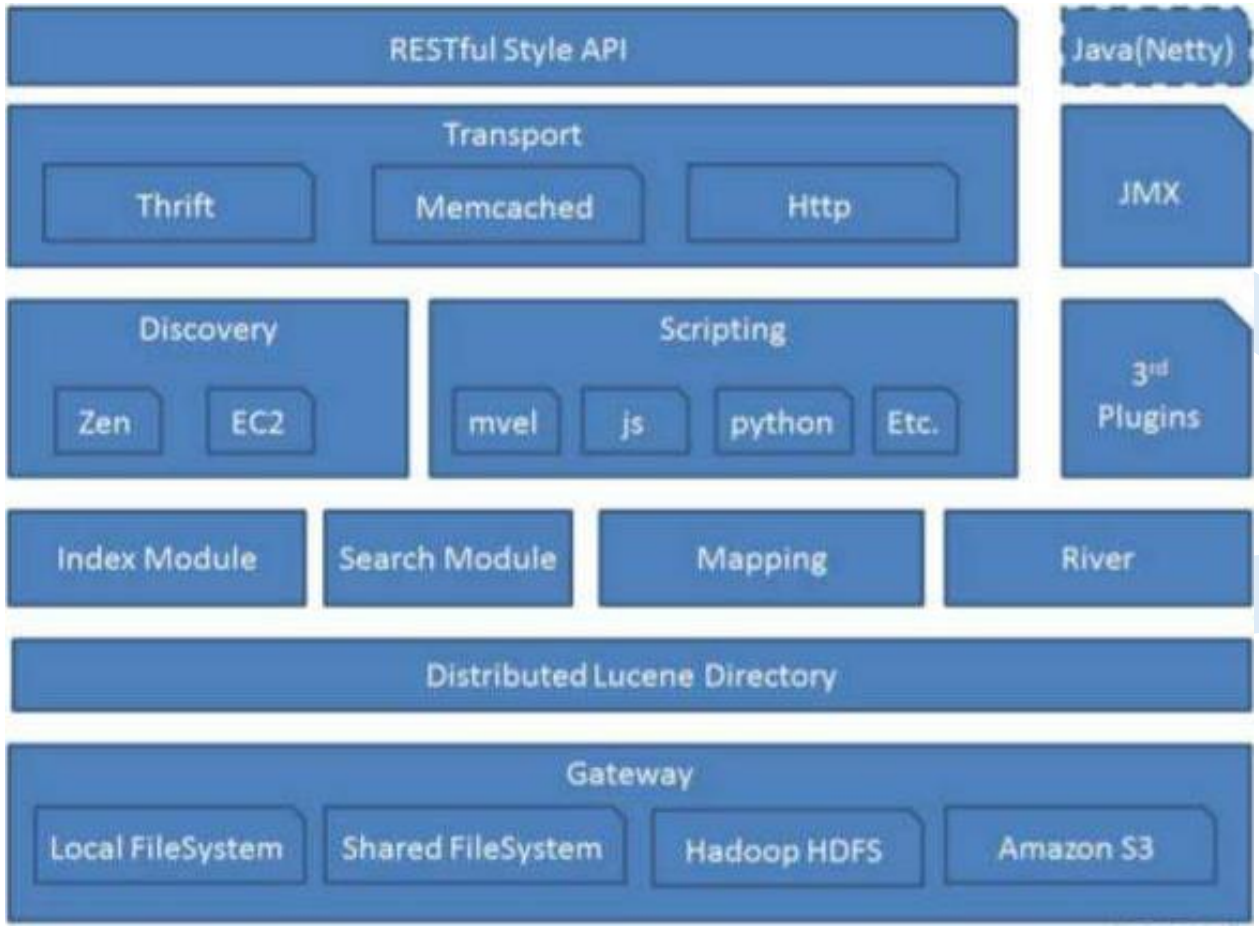
Vs. 数据库

- 与传统关系型数据库：
  - 无法对搜索出的结果打分
  - 无法洞悉搜索请求
  - 分词问题
  - 承担的数据规模有限
  - 开发效率低

- 与Nosql文档数据库
  - 复杂一对多的关系，需要创建多个关系表，进行互相映射，实现复杂度高

```
{
"title":"我们一块学习ES",
"tag":["Hadoop", ''Spark"],
"course":"大数据大规模检索学习",
"class":{
"name":"第十节",
"timelen":"45分钟",
"content":["ES搭建","scrapy实现"]
}
"from":"China"
}
```

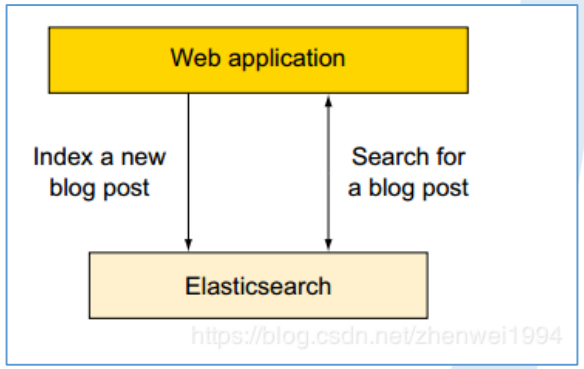
架构



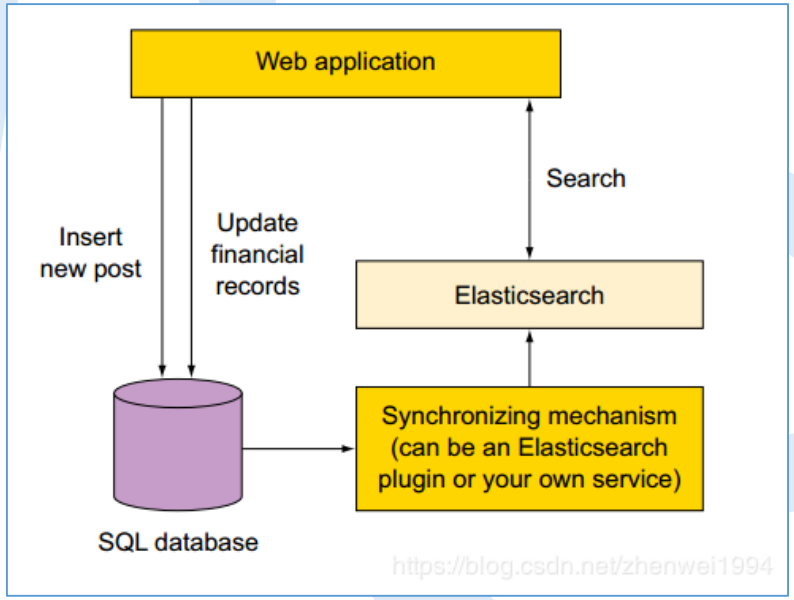
- API编程
- 协议交互
- 集群发现、脚本开发
- 索引、搜索、映射、插件
- Lucene里的一些列索引文件组成的目录
- ES索引快照的存储方式，默认内存

使用案例

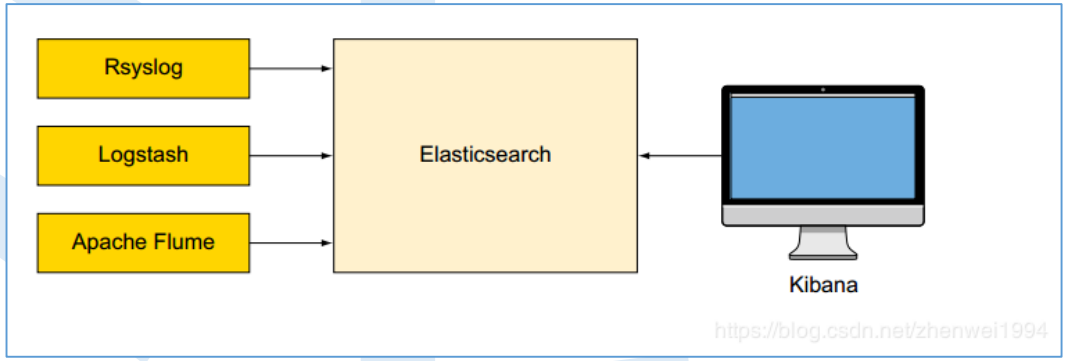
- 作为网站主要后端系统



- 添加到现有系统



- 现有解决方案的后端



文档

- 可以被索引的最基本的信息单位
  - 是自我包含的。一篇文档同时包含字段和他们的取值。
  - 是层次型的。文档中还可以包含新的文档，一个字段的取值可以是简单的，例如location字段的取值可以是字符串，还可以包含其他字段和取值，比如可以同时包含城市和街道地址。
  - 拥有灵活的结构。文档不依赖于预先定义的模式。也就是说并非所有的文档都需要拥有相同的字段，并不受限于同一个模式

```
{
  "name":"meeting",
  "location":"office",
  "organizer":"yanping"
}
```

```
{
  "name":"meeting",
  "location":{
    "name":"sheshouzuo",
      "date":"2019-6-28"
  },
  "memebers":["leio","shiyi"]
}
```

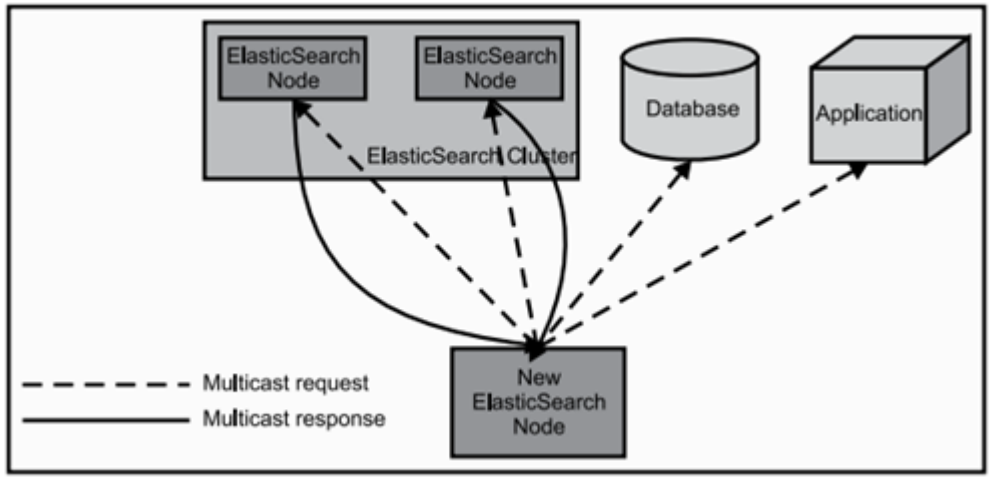
其他类型

- 类型
  - 文档的逻辑容器，类似于表格是行的容器。在不同的类型中，最好放入不同的结构的文档
- 字段
  - 每个文档，其实是以json形式存储的。而一个文档可以被视为多个字段的集合
- 映射
  - 每个类型中字段的定义称为映射。例如，name字段映射为String
- 索引
  - 索引是映射类型的容器一个ES的索引非常像关系型世界中的数据库，是独立的大量文档集合

与关系型数据库的结构对比

MySQL	ES
Database	Index
Table	Type
Row	Document
Column	Field
Schema	Mapping
Index	Everything is Indexed
SQL	Query DSL
SELECT * FROM Table…	GET http:// ...
UPDATE Table SET …	PUT http:// ...

物理设计



- 一个节点是一个ES的实例，在服务器上启动ES之后，就拥有了一个节点，如果在另一个服务器上启动ES，这就是另一个节点。甚至可以在一台服务器上启动多个ES进程，在一台服务器上拥有多个节点。多个节点可以加入同一个集群

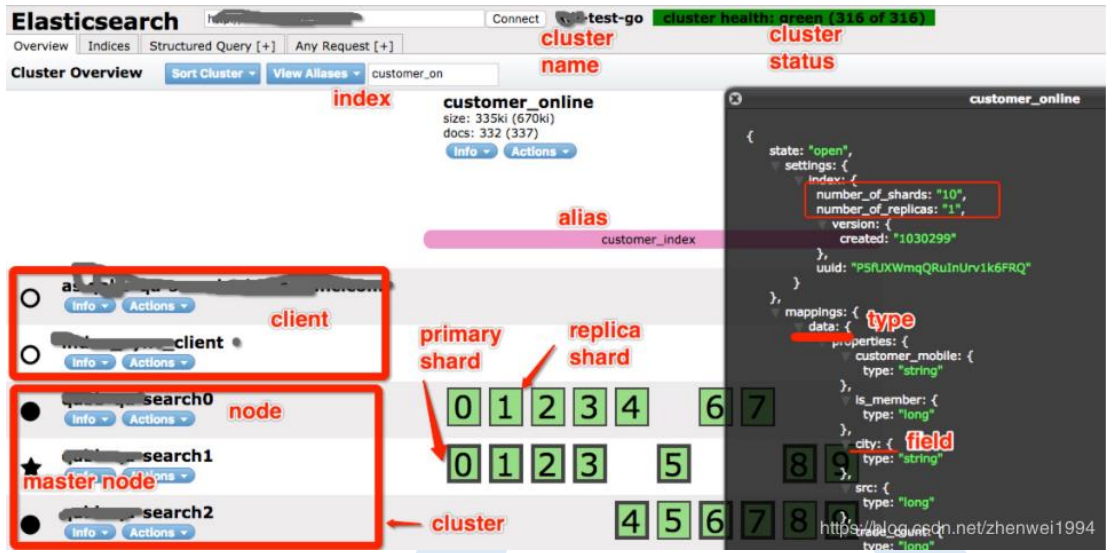
- 节点类型：
- Client_node：请求分发作用
- Master_node：主节点，增删分片均有该节点操作
- Date_node：只做搜索操作，分配由client_node决定，数据由master_node同步

分片

- 一个索引可以存储超出单个结点硬件限制的大量数据，需要考虑分片能力
- 每个分片本身也是一个功能完善并且独立的“索引”，这个“索引”可以被放置到集群中的任何节点上
- 分片重要性：
  - 允许水平分割：扩展内容容量
  - 容错恢复
    - 复制分片，多副本

- 默认，ES中的每个索引被分片5个主分片和1个复制，意味着索引将会有5个主分片和另外5个复制分片（1个完全拷贝），这样的话每个索引总共就有10个分片

插件



- node：即一个 Elasticsearch 的运行实例，多播或单播发现 cluster 加入
- cluster：包含一或多个拥有相同集群名称的 node，包含一个master
- index：类比关系型数据库里的DB，是一个逻辑命名空间。
- alias：可以给 index 添加零个或多个alias，通过 alias 使用index 和根据 index name 访问index一样
- type：关系数据库的Table。一个index可有多个type，一般仅配一个type
- mapping：数据库 schema 概念，mapping 定义了 index 中的 type
- document：类比关系数据库里的一行记录(record)，document 是 Elasticsearch 里的一个 JSON 对象，包括零个或多个field。
- field：类比关系数据库里的field，每个field 都有自己的字段类型。
- shard：是一个Lucene 实例。

shard、node、cluster 在物理上构成了 Elasticsearch 集群
field、type、index 在逻辑上构成一个index的基本概念

原理-倒排索引

- Term（单词）：一段文本经过分析器分析以后就会输出一串单词，这一个一个的就叫做Term
- Term Dictionary（单词字典）：它里面维护的是Term，可以理解为Term的集合
- Term Index（单词索引）：为了更快的找到某个单词，我们为单词建立索引
- Posting List（倒排列表）：倒排列表记录了出现过某个单词的所有文档的文档列表及单词在该文档中出现的位置信息，每条记录称为一个倒排项(Posting)。根据倒排列表，即可获知哪些文档包含某个单词，同时还有一些其它的信息，比如：词频（Term出现的次数）、偏移量（offset）等

例子

id	name	gender	age	address
1	张三	1	22	北京市朝阳区
2	李四	2	21	上海市徐汇区
3	王五	1	23	上海市虹口区

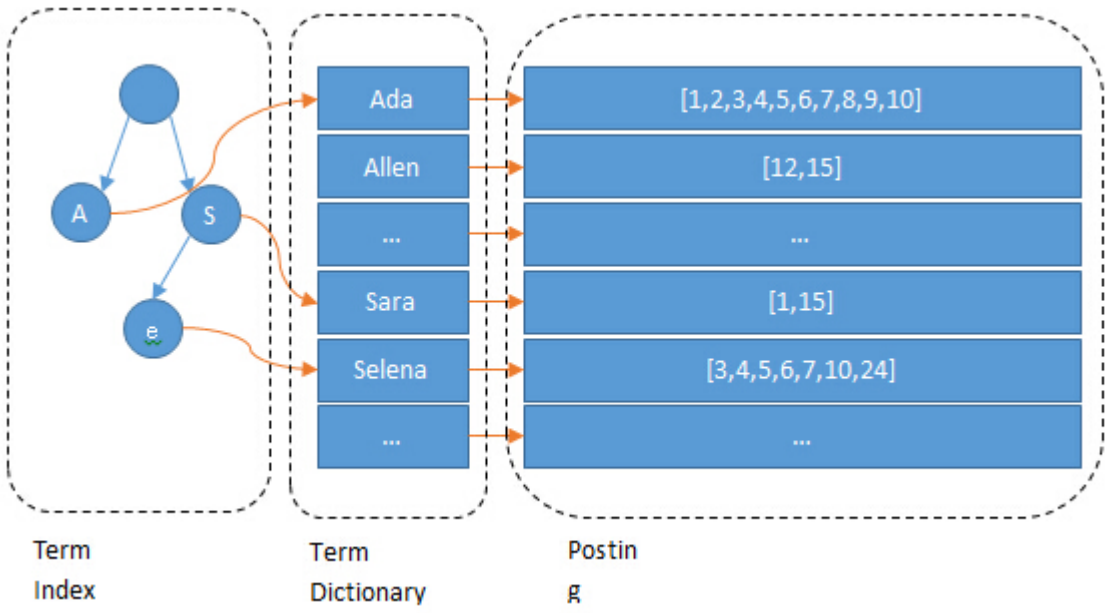
Term	Posting List
张三	1
李四	2
王五	3

Term	Posting List
21	2
22	1
23	3

Term	Posting List
1	[1, 3]
2	2

Term	Posting List
北京市	1
上海市	[2, 3]
徐汇区	2
虹口区	3
朝阳区	1

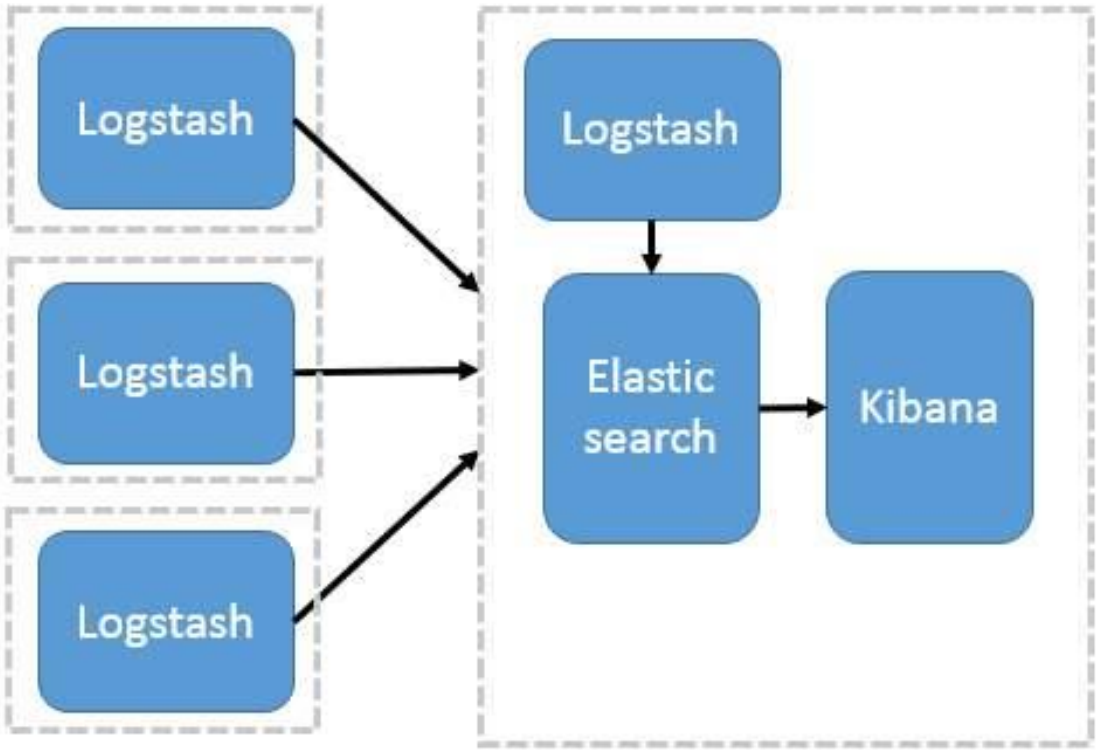
Elasticsearch分别为每个字段都建立了一个倒排索引



ELK

ELK = Elasticsearch, Logstash, Kibana 是一套实时数据收集，存储，索引，检索，统计分析及可视化解决方案

Logstash是一个用来搜集、分析、过滤日志的工具。它支持几乎任何类型的日志，包括系统日志、错误日志和自定义应用程序日志。



Q & A

@八斗学院