# ElasticSearch

### 一、安装

1. OS

版本:centos7.3

2、javasdk

版本:9.0

参考文档:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk9-downloads-3848520.html>

备注:java环境变量

export JAVA\_HOME=/apps/java/jdk-9.0.1

export JRE\_HOME=${JAVA\_HOME}/jre

export CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib

export PATH=${JAVA\_HOME}/bin:$PATH

3、ES

版本:6.1.2

安装目录:/apps/es

参考文档：

<https://www.elastic.co/guide/en/elasticsearch/reference/6.1/install-elasticsearch.html>

安装命令:

wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.1.2.tar.gz

tar -zxvf elasticsearch-6.1.2.tar.gz

### 二、运行

cd /apps/es

1. 用户

adduser es

passwd es

chown –R es /apps/es/elasticsearch-6.1.2

su es

备注:es不允许以root用户执行

1. 修改配置

vi config/elasticsearch.yml

修改如下:

path.data: /apps/es/elasticsearch-6.1.2/data #数据目录

path.logs: /apps/es/elasticsearch-6.1.2/logs #日志目录

network.host: 10.0.0.80 #host地址

http.port: 9200 #端口

1. 执行

./bin/elasticsearch –d –p pid

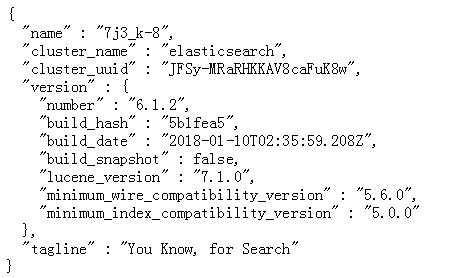
说明:

p: 进程ID将写入pid文件,结束ES执行命令 kill `cat pid`

备注：第一次启动时应去掉-d选项，以排查启动时的错误。

验证:

访问: <http://ip:9200/>，出现下图:



常见错误:

参考 http://blog.csdn.net/weini1111/article/details/60468068

1. max file descriptors [4096] for elasticsearch process is too low, increase to at least [65536]

解决:

su

vi /etc/security/limits.conf

添加内容:

\* soft nofile 65536

\* hard nofile 131072

\* soft nproc 2048

\* hard nproc 4096

1. max virtual memory areas vm.max\_map\_count [65530] is too low, increase to at least [262144]

解决:

su

vi /etc/sysctl.conf

添加内容:

vm.max\_map\_count=655360

执行命令：

sysctl -p

### 三、插件

#### 1、kibana

用于访问索引和操作ES的工具箱。更高级的功能如数据分析、报表、监控没有用到。

项目地址:

<https://github.com/elastic/kibana>

安装

wget https://artifacts.elastic.co/downloads/kibana/kibana-6.1.2-linux-x86\_64.tar.gz

tar –zxvf

修改配置

vi config/kibana.yml

server.host: "10.0.0.80" #host地址

server.port: 5601

elasticsearch.url: <http://10.0.0.80:9200> #es地址

运行

./bin/kibana &

备注:第一次运行应去掉&，以排查错误

检测:

http://ip:5601

#### 2、IK分词器

项目地址:

<https://github.com/medcl/elasticsearch-analysis-ik>

分词选项

1. ik\_max\_word:细粒度，列出所有的可能
2. ik\_smart:粗粒度

安装

wget https://github.com/medcl/elasticsearch-analysis-ik/releases/download/v6.1.2/elasticsearch-analysis-ik-6.1.2.zip

拿到下载目录,demo: /apps/ik/elasticsearch-analysis-ik-6.1.2.zip

在es的安装目录执行

./bin/elasticsearch-plugin install file:///apps/ik/elasticsearch-analysis-ik-6.1.2.zip

校验：

1. ik\_max\_word分词

POST /\_analyze

{

"analyzer":"ik\_max\_word",

"text":"我想带你去骑单车"

}

结果: 我|想|带你去|带你|去骑|骑单车|单车

1. ik\_smart分词

POST /\_analyze

{

"analyzer":"ik\_smart",

"text":"我想带你去骑单车"

}

结果: 我|想|带你去|骑单车

配置扩展词库

cd /apps/es/elasticsearch-6.1.2/config/analysis-ik

vi IKAnalyzer.cfg.xml

修改

<entry key="ext\_dict">mykeyword.dic</entry>

案例

POST /\_analyze

{

"analyzer":"ik\_max\_word",

"text":"马云"

}

未配置之前:马|云

配置之后:马云

备注:如果配置了扩展词库，某些词没有生效，比如“特步”,”三只小熊”，可以放在main.dic.

在配置网络词库时候报

**java.security.AccessControlException: access denied (java.net.SocketPermission**

**需要修改配置文件**

%JAVA\_HOME%/conf/security/java.policy

增加：

**permission java.net.SocketPermission "词库IP:词库端口","accept";**

**permission java.net.SocketPermission "词库IP:词库端口","listen";**

**permission java.net.SocketPermission "词库IP:词库端口","resolve";**

**permission java.net.SocketPermission "词库IP:词库端口","connect";**

#### 3、同义词配置

1、ES自带同义词filter，新建同义词词典，如synonyms.txt，URF8编码，测试时，不带文件头。内容如下（注意每个,号之间有空格）:

Iphone, apple

2、在synonyms.txt copy至es安装目录下的config文件夹。并设置synonyms.txt为es运行用户权限:

如:

chown –R es /apps/es/elasticsearch-6.1.2/config/synonyms.txt

3、在建立索引时，配置分词器和过滤器，如：

{ "index" ,

new

{

number\_of\_replicas = 1,

number\_of\_shards = 1,

analysis = new {

analyzer = new {

ext\_ik\_smart = new {

tokenizer = "ik\_smart",

filter = new string[]{ "synonym"}

},

ext\_ik\_max\_word = new {

tokenizer = "ik\_max\_word",

filter = new string[]{ "synonym"}

}

},

filter = new {

synonym = new {

type = "synonym",

synonyms\_path = "synonyms.txt"

//synonyms = new string[]{

//"iphone, apple",

//}

}

}

}

}

}

上面定义了两个分词器: ext\_ik\_smart, ext\_ik\_max\_word，在定义映射时，需要使用同义词的字段需指定该分词器。

自更新：

1、安装插件 <https://github.com/bells/elasticsearch-analysis-dynamic-synonym/tree/master>

目前不支持es的高版本，又程序完成。

### 四、客户端

1、框架 elasticsearch.net

地址: <https://github.com/elastic/elasticsearch-net>

Nuget: install-package nest

五、查询案例

<https://www.iteblog.com/archives/1768.html>

<http://m635674608.iteye.com/blog/2274337> : 热门度

### 五、理论

#### 1、评分机制

参考:

<https://www.elastic.co/guide/cn/elasticsearch/guide/current/relevance-intro.html>

查看某条记录的查询评分

GET [索引名]/[类型名]/[id]/\_explain

[查询语句]….

A、TF: 词频 term frequency

参考: <https://www.elastic.co/guide/cn/elasticsearch/guide/current/scoring-theory.html>

某个词在字段中出现次数的平方根。在添加映射时，可以禁用TF。可以参考Demo中的“keyword”字段配置。注：如果只有某个字段不计算词频，则它的分数远低于其他的字段。

B、IDF:逆向文档频率 inverse document frequency

词在集合所有文档里出现的频率是多少？频次越高，权重 *越低* 。

索引中文档数量除以所有包含该词的文档数，然后求其对数。

C、字段长度 field-length norm

归一化:归一化的值做为字段长度的权重值，词越多，权重越低。

字段长度归一值（ norm ）是字段中词数平方根的倒数，

可以在字段映射时，禁用。

D、控制分值

参考: <https://www.elastic.co/guide/cn/elasticsearch/guide/current/function-score-query.html>

Function\_score: 它允许为每个与主查询匹配的文档应用一个函数，以达到改变甚至完全替换原始查询评分 \_score 的目的。

#### 2、文本查询

参考:

<https://www.elastic.co/guide/cn/elasticsearch/guide/current/term-vs-full-text.html>

Term: 对搜索词做精确查询，搜索倒排索引中已经分词的term。

Match:对搜索词分词后，按term的方式查询。

参考: <https://www.elastic.co/guide/cn/elasticsearch/guide/current/match-query.html>

#### 3、人类语言

1、停用词: stopword

2、分词

3、同义词

参考: https://www.elastic.co/guide/cn/elasticsearch/guide/current/using-synonyms.html