# ElasticSearch学习笔记

## ElasticSearch安装

本文是在Windows的环境下安装ElasticSearch。

### java环境要求

jdk：安装不同的ElasticSearch需要不同版本的jdk，本文中采用的是较高版本的ElasticSearch，jdk为8。

java -version 可以查看电脑中安装的jdk版本信息。

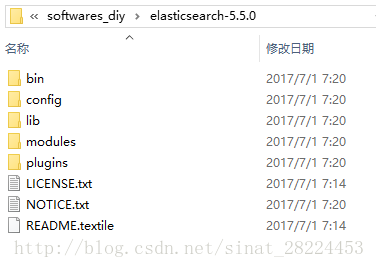
* 1. 安装ElasticSearch

ElasticSearch下载地址：<https://www.elastic.co/downloads/elasticsearch>

下载ZIP格式。

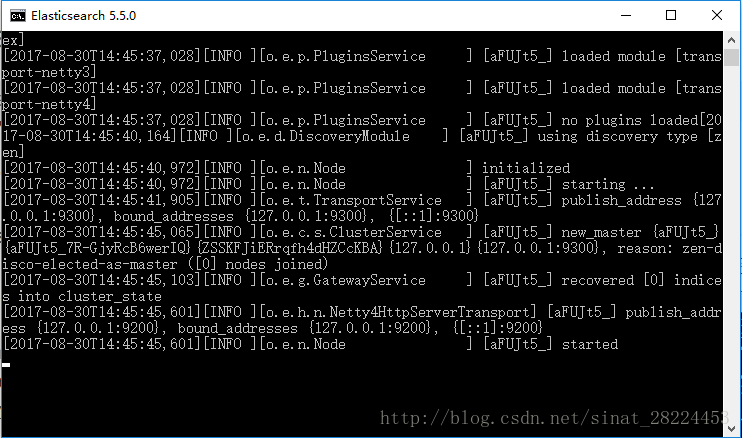
#### 解压

可选择任意的盘或文件夹下，解压ElasticSearch的压缩包。



#### 安装

进入bin目录下，双击执行elasticsearch.bat



进入 [http://localhost:9200 页面，出现以下页面，则说明安装成功。](http://localhost:9200页面，出现以下页面，则说明安装成功。)



页面上json里的 name 是节点的名字，集群名称是 elasticsearch，还有其他版本信息,想停止的话，ctrl+c

### 1.2、ElasticSearch中文版

下载地址：<https://github.com/medcl/elasticsearch-rtf（本文安装的是英文版，原版）>

### 1.3、ElasticSearch设置为服务启动

启动时通过cmd直接在elasticsearch的bin目录下执行elasticsearch

这样直接启动的话集群名称会默认为elasticsearch，节点名称会随机生成。

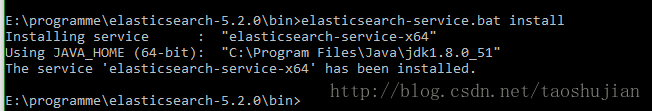
停止就直接在cmd界面按Ctrl+C

其实我们也可以将elasticsearch设置为windows系统服务：

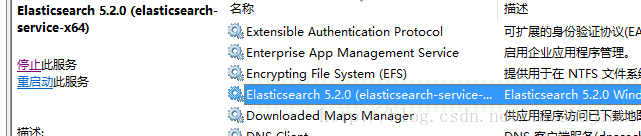
elasticsearch的bin目录下有一个elasticsearch-service.bat

进入bin目录下执行:

elasticsearch-service.bat install



然后在系统服务中可以看到Elasticsearch已成为系统服务。



elasticsearch-service.bat后面还可以执行这些命令

install: 安装Elasticsearch服务

remove: 删除已安装的Elasticsearch服务（如果启动则停止服务）

start: 启动Elasticsearch服务（如果已安装）

stop: 停止服务（如果启动）

manager:启动GUI来管理已安装的服务

## 安装head插件

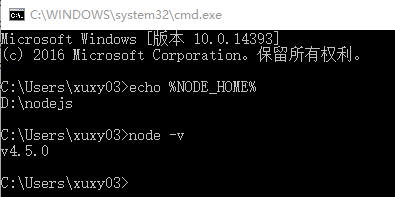
### 环境

* Windows10企业版X64
* JDK-1.8
* ElasticSearch-5.0.0
* node-v4.5.0-x64.msi
* git客户端

### 步骤

安装node到D盘。如D:\nodejs。

把NODE\_HOME设置到环境变量里(安装包也可以自动加入PATH环境变量)。测试一下node是否生效：



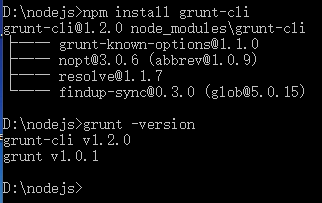
### 安装grunt

grunt是一个很方便的构建工具，可以进行打包压缩、测试、执行等等的工作，5.0里的head插件就是通过grunt启动的。因此需要安装grunt：

注意：路径切到D:\nodejs下。

命令：npm install -g grunt-cli

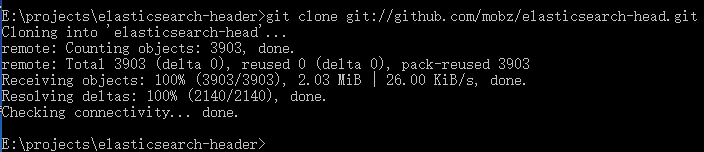
-g代表全局安装。安装路径为C:\Users\yourname\AppData\Roaming\npm，并且自动加入PATH变量。安装完成后检查一下：



把head插件的源码git clone下来：

git clone git://github.com/mobz/elasticsearch-head.git

效果如图：



### 修改head源码

由于head的代码还是2.6版本的，直接执行有很多限制，比如无法跨机器访问。因此需要用户修改两个地方：

目录：head/Gruntfile.js：

connect: {

server: {

options: {

port: 9100,

hostname: '\*',

base: '.',

keepalive: true

}

}

}

增加hostname属性，设置为\*

修改连接地址：

目录：head/\_site/app.js

修改head的连接地址:

this.base\_uri = this.config.base\_uri || this.prefs.get("app-base\_uri") || "http://localhost:9200";

把localhost修改成你es的服务器地址，如：

this.base\_uri = this.config.base\_uri || this.prefs.get("app-base\_uri") || "http://10.10.10.10:9200";

### 运行head

修改elasticsearch的参数

修改一下es使用的参数。编辑config/elasticsearch.yml：

# 换个集群的名字，免得跟别人的集群混在一起

cluster.name: es-5.0-test

# 换个节点名字

node.name: node-101

# 修改一下ES的监听地址，这样别的机器也可以访问

network.host: 0.0.0.0

# 默认的就好

http.port: 9200

# 增加新的参数，这样head插件可以访问es

http.cors.enabled: true

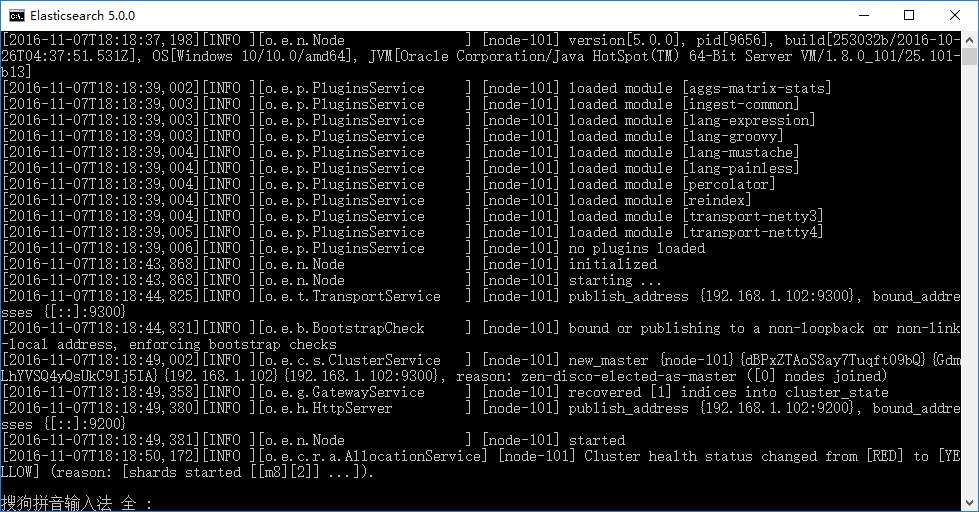
http.cors.allow-origin: "\*"

注意，设置参数的时候:后面要有空格！

开启ES-5.0.0：

D:\ElasticSearch-5.0.0\bin\elasticsearch.bat

启动效果：



然后在head源码目录中，执行npm install 下载的包：

npm install

效果如图：

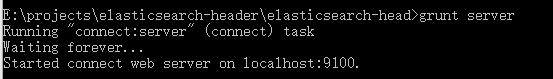
IMG_256

初次运行安装可能会报警告或错误。可以重新运行一次npm install。

最后，在head源代码目录下启动nodejs：

grunt server

效果如图：

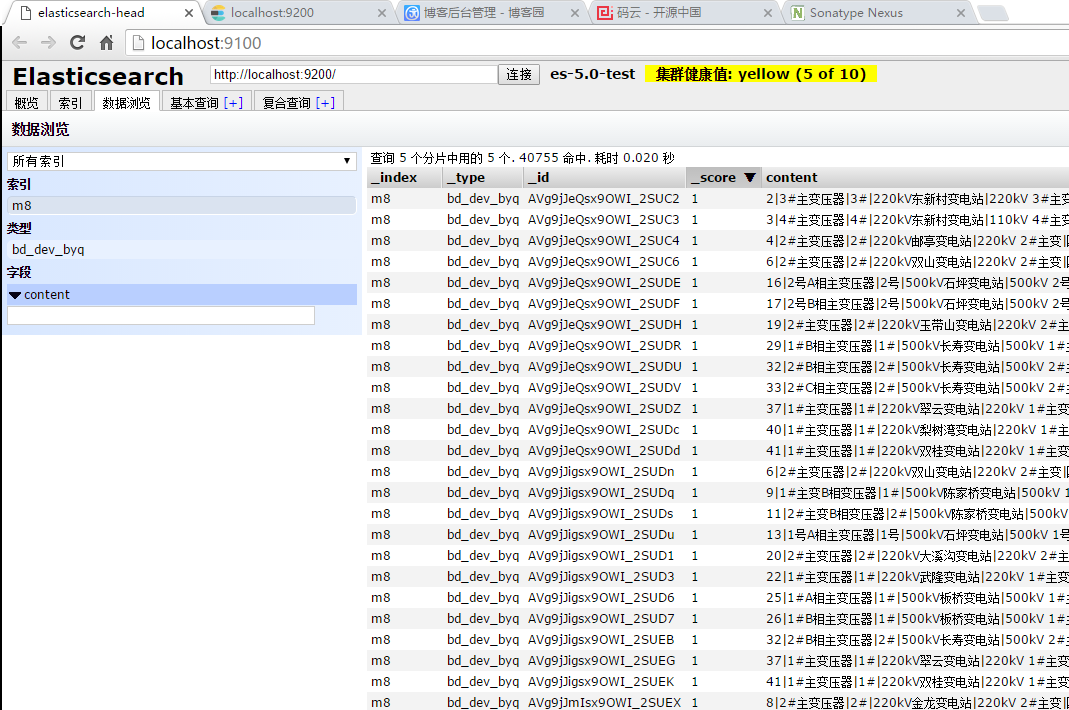


访问:target:9100

这个时候，访问http://localhost:9100就可以访问head插件了：



因为之前已在ES中建立了一个索引m8，因此可以看到数据：



## ElasticSearch中文分词器IK

### 3.1、安装中文分词器IK

一ElasticSearch安装中文分词器

* 1. 中文分词器下载地址：

gitclone https://github.com/medcl/elasticsearch-analysis-ik

1.2 切换分支： gitcheckout tags/v5.2.0

1.3mvn 打包： mvn clear package

1.4将target/releases/elasticsearch-analysis-ik-5.2.0.zip拷贝到

ElasticSearch安装目录下的 plugins文件夹下，并创建目录ik，将该压缩包解压到ik目录下

1.5 重启ES

### 3.2、ik分词器基础

#### 3.2.1、ik分词器精细度说明

##### 3.2.1.1、ik\_max\_word

会将文本做最细粒度的拆分

POST /ecommerce/\_analyze

{

"text":"深秋嫩绿的垂柳",

"analyzer":"ik\_max\_word"

}

它会拆分成"深秋","嫩绿","垂柳","绿","垂","柳"

##### 3.2.1.2、ik\_smart

会做最粗粒度的拆分

POST /ecommerce/\_analyze

{

"text":"深秋嫩绿的垂柳",

"analyzer":"ik\_smart"

}

"深秋","嫩绿","垂柳"

#### 3.2.2 IK分词器的使用

##### 3.2.2.1、 首先建立映射

PUT/ecommerce

{

"mappings": {

"music":{

"properties": {

"brand": {

"type":"text",

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

"fields": {

"keyword": {

"type":"keyword",

"ignore\_above": 256

}

}

},

"desc": {

"type":"string",

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

"fields": {

"keyword": {

"type":"keyword",

"ignore\_above": 256

}

}

},

"sales":{

"type":"long"

},

"price":{

"type":"float"

},

"review":{

"type":"integer"

},

"color":{

"type":"string",

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

"fields": {

"keyword": {

"type":"keyword",

"ignore\_above": 256

}

}

},

"origin":{

"type":"string",

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

"fields": {

"keyword": {

"type":"keyword",

"ignore\_above": 256

}

}

},

"c\_date":{

"type":"date"

}

}

}

}

}

##### 3.2.2.2、 然后添加数据

PUT /ecommerce/music/1

{

"brand":"雅马哈",

"desc":"正品YAMAHA雅马哈F310 F600初学者入门男女学生民谣木吉他 41寸",

"color":"红",

"origin":"美国",

"sales":1914,

"price":799.00,

"review":786,

"c\_date":"2015-12-21"

}

PUT /ecommerce/music/2

{

"brand":"雅马哈",

"desc":"雅马哈FG800单板民谣木吉他 FGX800C电箱吉他FG700S升级41/40寸",

"color":"黄",

"origin":"美国",

"sales":102,

"price":1590.00,

"review":349,

"c\_date":"2016-09-12"

}

##### 3.2.2.3、 分词分析

POST /ecommerce/\_analyze

{

"text":"美德威MP2000X智能电钢琴88键重锤成人电子钢琴家用数码电钢专业",

"analyzer":"ik\_max\_word"

}

##### 3.2.2.4、 查询数据

POST /ecommerce/music/\_search

{

"query":{

"match": {

"desc":"罗兰"

}

}

}

POST /ecommerce/music/\_search

{

"query":{

"constant\_score": {

"filter": {

"term":{

"desc":"吉他"

}

}

}

}

}

### 3.3、配置更新IK

IK默认提供的很多分词，并不能满足我们所有的情况，不同公司，可能都有自己的数据，而且还有一些最近流行的一些新词也是没有。所以这就需要我们更新词库。

我们首先需要进到IK的config目录，有一个IKAnalyzer.cfg.xml的配置文件：

main.dic：ik原生内置的中文词库

quantifier.dic：放了一些单位相关的词

suffix.dic：放了一些后缀

surname.dic：中国的姓氏

stopword.dic：英文停用词

在custom目录下也放了些词库

ext\_stopword.dic：扩展的停用词，我们可以用来添加中文停用词

### 3.4、IK词库的热更新

我们如果每次需要手动去更新词库，每次都要重启ElasticSearch，如果集群数量太多，比如几百台 怎么去弄，所以我们有必要再不重启ElasticSearch的情况下热更新。

#### 3.4.1、热更新的方案

方案一： 修改ik分词器源码，然后手动支持从mysql中每隔一定时间，自动加载新的词库。

方案2： 基于ik分词器原生支持的热更新方案，部署一个web服务器，提供一个http接口，通过modified和tag两个http响应头，来提供词语的热更新。

一般公司都会采用第一种，第二种官方都不建议，因为不稳定

核心思想就是不停的reload你想更新的词典文件，比如ext\_stop.dic或者main.dic，然后读取数据库里面的配置更新。