# 环境要求

Zookeeper版本：zookeeper-3.4.8

JDK版本： jdk1.8.

Solr版本：solr-6.4.1

Tomcat版本：tomcat8

# 应用部署

已经打包Tomcat压缩文件<apache-tomcat-8.5.11-solr.zip>



## 打包过程：

1. 将solr-6.4.1\server\solr-webapp\webapp下面的所有文件复制到tomcat/webapps/solr/目录下
2. 将 solr-6.4.1\server\lib\下面所有的\*.jar 复制到tomcat/webapps/solr/WEB-INF/lib/目录下
3. 将 solr-6.4.1\server\lib\ext\下面所有的\*.jar 复制到tomcat/webapps/solr/WEB-INF/lib/目录下
4. 将 solr-6.4.1\server\solr 下的所有文件复制到tomcat/solrhome/目录下
5. 增加tomcat/bin/catalina.sh中的JAVA\_HOME和zk的配置

JAVA\_OPTS="-DzkHost=127.0.0.1:2181"

JAVA\_HOME="C:/DevTool/jdk1.8.0\_92/"

## 配置修改

1. 修改tomcat/solrhome/solr.xml中的hostPort为tomcat的访问端口（例如tomcat访问端口为8080，那么修改配置如下）

<int name="hostPort">10100</int>

1. 修改tomcat/webapps/solr/WEB-INFO/web.xml修改solr/home的值（改目录为打包过程第四部的solrhome目录），例如：

<env-entry>

<env-entry-name>solr/home</env-entry-name>

<env-entry-value>C:\Users\Silver\Desktop\apache-tomcat-8.5.11-10100\solrhome</env-entry-value>

<env-entry-type>java.lang.String</env-entry-type>

</env-entry>

1. 设置日志输出

tomcat/webapps/solr/WEB-INF/classes/目录下增加log4j.properties文件 ，并更改solr.log=D:/logs/solr/



## 设置分词（IKAnalyzer分词）

Git下载源码工程：https://github.com/cj96248/ik-analyzer-solr6.git

将工程打包出JAR包，并且放在tomcat/webapps/solr/WEB-INF/lib/目录下

配置参考：

<http://www.tuicool.com/articles/faYFBzA>

原理：在字段上面进行了分词处理，solr服务器数据在插入会调用分词进行索引，查询的时候调用分词进行查询

## 可能遇到的问题

1. ClassNotFund -- com.codahale.metrics.MetricSet

-- solr-6.4.1\server\lib\除了jetty之外的jar包都放到 tomcat/webapps/solr/WEB-INFO/lib/下

1. HTTP Status 403 - Access to the requested resource has been denied

-- 注释掉tomcat/webapps/solr/WEB-INFO/web.xml中的所有<security-constraint>标签

1. Xid out of order. Got Xid 52 with err 0 expected Xid 51 for a packet with details: clientPath:null serverPath:null finished:false header:: 51,14

-- 升级zookeeper版本为3.4.8

1. UnsupportedClassVersionError: org/apache/solr/servlet/SolrDispatchFilter : Unsupported major.minor version 52.0

-- Solr工程需要JDK8以上，因此需要设置JAVA\_HOME为JDK8

## 程序封装

配置文件主要：

#solr搜索引擎配置

# zookeeper地址 公用

system.zookeeper.address=127.0.0.1:2181

# app应用环境 公用

system.appenv.name=dev\_online

# solr 搜索区分业务

search.solr.core.name=help\_center

# solr 每次接口调用超时时间

search.solr.timeout=1000

# solr 配置，如果search.solr.core.name在集群中不存在，那么初始化solr的配置

#search.solr.configpath=solr\_help\_center\_config/ 示例目录为：classpath:solr\_help\_center\_config/

search.solr.configpath=solr\_help\_center\_config/

# solr 如果search.solr.core.name在集群中不存在，那么初始化集群的分片数量，构造函数中使用，建议与solr集群数量同步

search.solr.numshards=3

1. 区分搜索场景，例如帮助中心和产品搜索需要有不同的业务场景隔离，通过solr的collection实现
2. 区分环境，不同的环境的帮助中心，也需要进行隔离，通过封装collection的实现，实现不同的环境下相同的业务场景collection是不一样的
3. 初始化设置：

由于要支持不同场景初始化设置，因此封装了配置上传以及collection自动创建功能

1. 存储查询

目前提供的API是通过将对象转化为JSON格式存储到一个字段（改字段配置了IK分词，在存储和查询的过程中会自动分词）中进行操作