# C:\Users\sz\Desktop\dfs\zsCat.pngzsCat

开源分布式框架zaCat 群 559182393

# zaCat文件搜索solr引擎

# Solr概述

什么是Solr

Solr 是Apache下的一个顶级开源项目，采用Java开发，它是基于Lucene的全文搜索服务器。Solr提供了比Lucene更为丰富的查询语言，同时实现了可配置、可扩展，并对索引、搜索性能进行了优化。

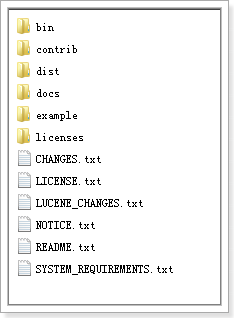
Solr可以独立运行，运行在Jetty、Tomcat等这些Servlet容器中，Solr 索引的实现方法很简单，用 POST 方法向 Solr 服务器发送一个描述 Field 及其内容的 XML 文档，Solr根据xml文档添加、删除、更新索引 。Solr 搜索只需要发送 HTTP GET 请求，然后对 Solr 返回Xml、json等格式的查询结果进行解析，组织页面布局。Solr不提供构建UI的功能，Solr提供了一个管理界面，通过管理界面可以查询Solr的配置和运行情况。

## 下载

从Solr官方网站（http://lucene.apache.org/solr/ ）下载Solr4.10.3，根据Solr的运行环境，Linux下需要下载lucene-4.10.3.tgz，windows下需要下载lucene-4.10.3.zip。

Solr使用指南可参考：https://wiki.apache.org/solr/FrontPage。

下载lucene-4.10.3.zip并解压：



bin：solr的运行脚本

contrib：solr的一些贡献软件/插件，用于增强solr的功能。

dist：该目录包含build过程中产生的war和jar文件，以及相关的依赖文件。

docs：solr的API文档

example：solr工程的例子目录：

* example/solr：

该目录是一个包含了默认配置信息的Solr的Core目录。

* example/multicore：

该目录包含了在Solr的multicore中设置的多个Core目录。

* example/webapps：

该目录中包括一个solr.war，该war可作为solr的运行实例工程。

licenses：solr相关的一些许可信息

# Solr的安装及配置

## 运行环境

solr 需要运行在一个Servlet容器中，Solr4.10.3要求jdk使用1.7以上，Solr默认提供Jetty（java写的Servlet容器），本教程使用Tocmat作为Servlet容器，环境如下：

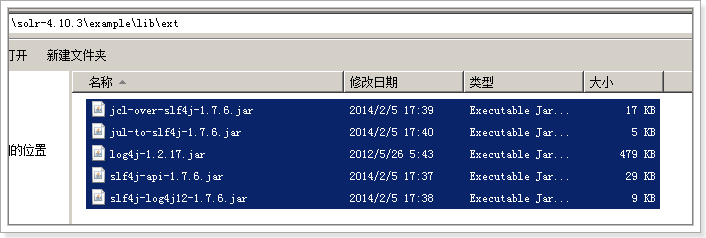
Solr：Solr4.10.3

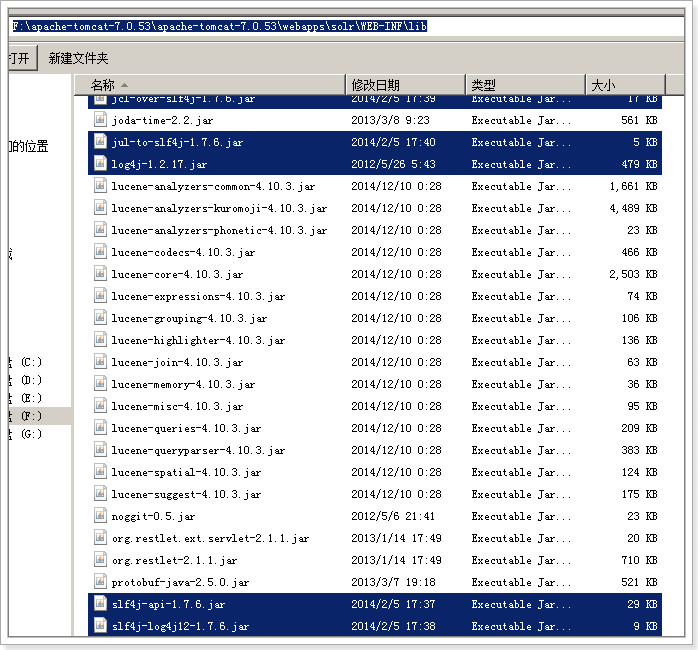
Jdk：jdk1.7.0\_72

Tomcat：apache-tomcat-7.0.53

## Solr整合tomcat

1. 将dist\solr-4.10.3.war拷贝到Tomcat的webapp目录下改名为solr.war
2. 启动tomcat后，solr.war自动解压，将原来的solr.war删除。
3. 拷贝example\lib\ext 目录下所有jar包到Tomcat的webapp\solr\WEB-INF\lib目录下





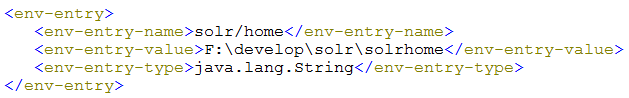
1. 拷贝log4j.properties文件

在 Tomcat下webapps\solr\WEB-INF目录中创建文件 classes文件夹，

复制Solr目录下example\resources\log4j.properties至Tomcat下webapps\solr\WEB-INF\classes目录

1. 创建solrhome及配置solrcore的solrconfig.xml文件
2. 修改Tomcat目录 下webapp\solr\WEB-INF\web.xml文件，如下所示：

设置Solr home



<!--配置jndi告诉solr工程我们的solrhome的位置-->

<env-entry>

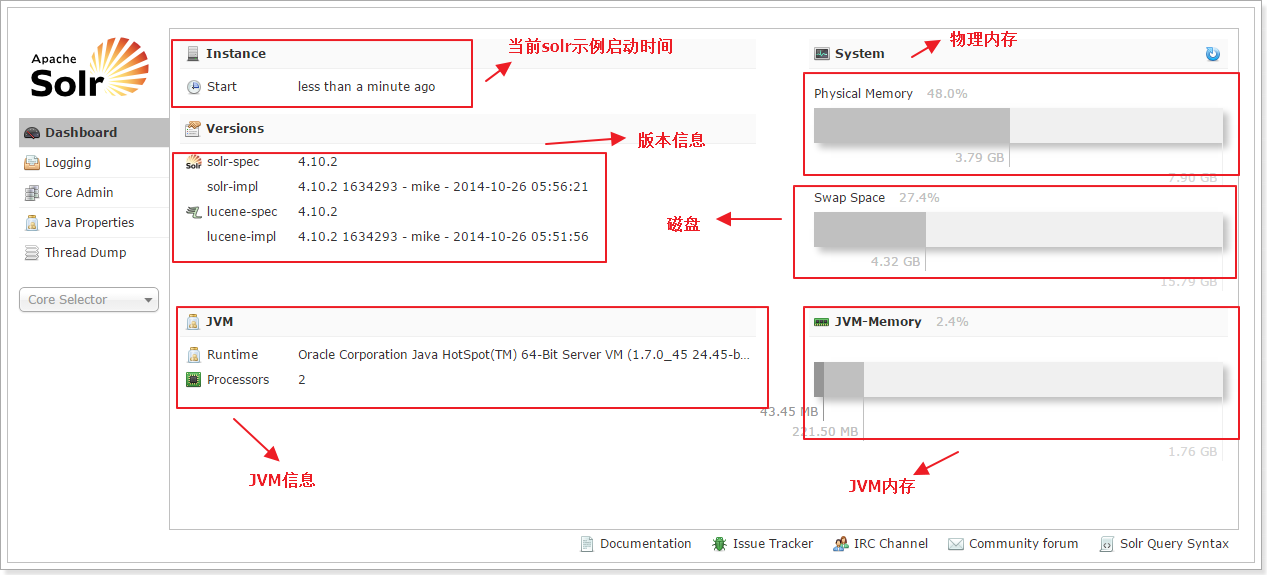
<env-entry-name>solr/home</env-entry-name>

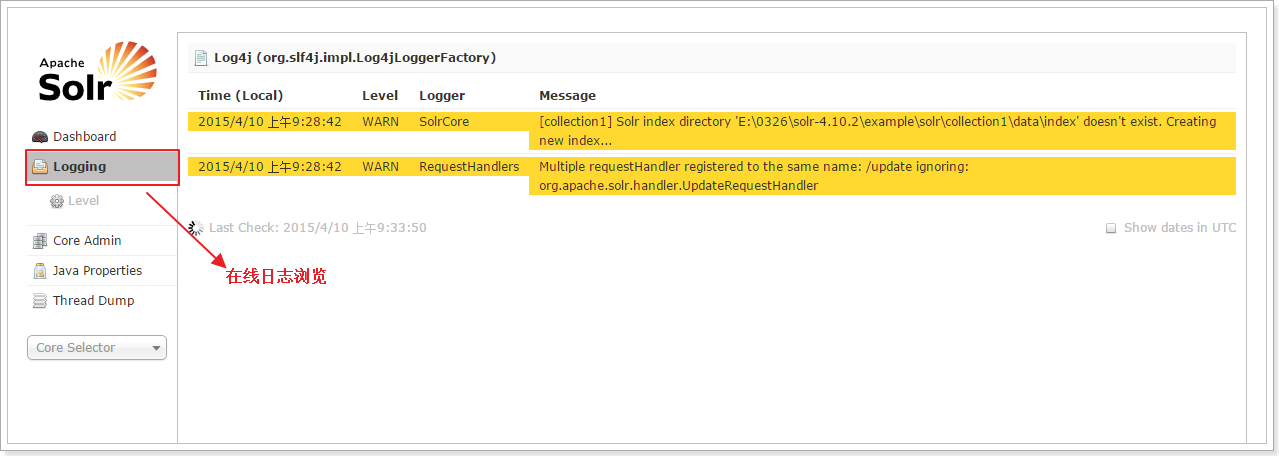
<env-entry-value>D:/temp/solr/solrhome</env-entry-value>

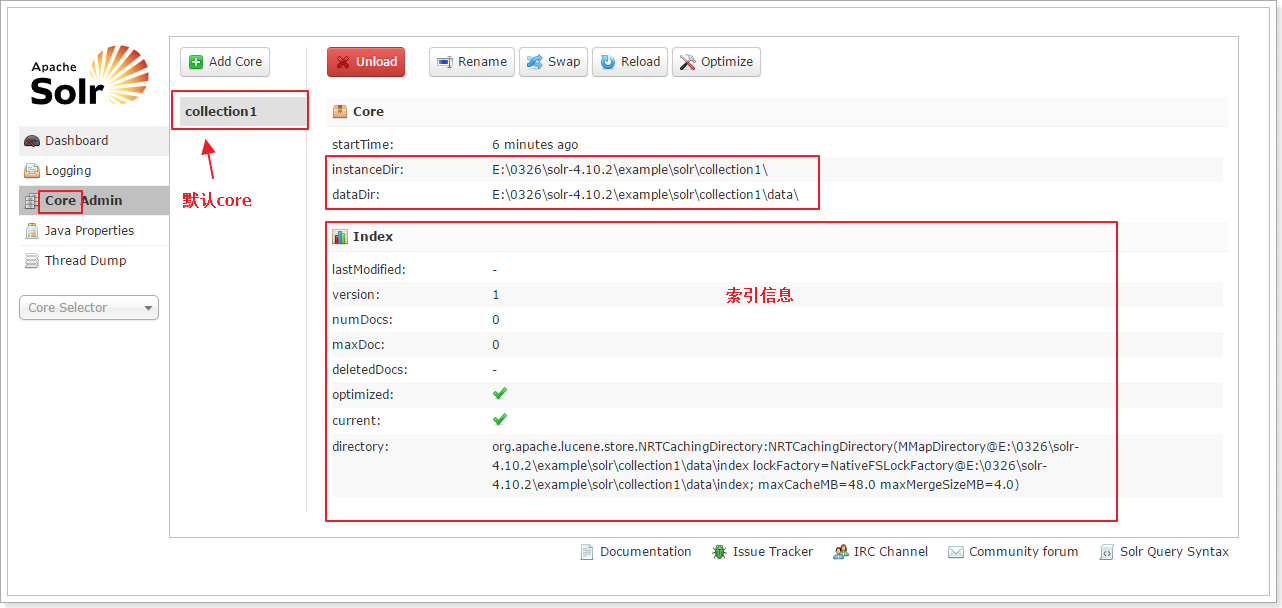
<env-entry-type>java.lang.String</env-entry-type>

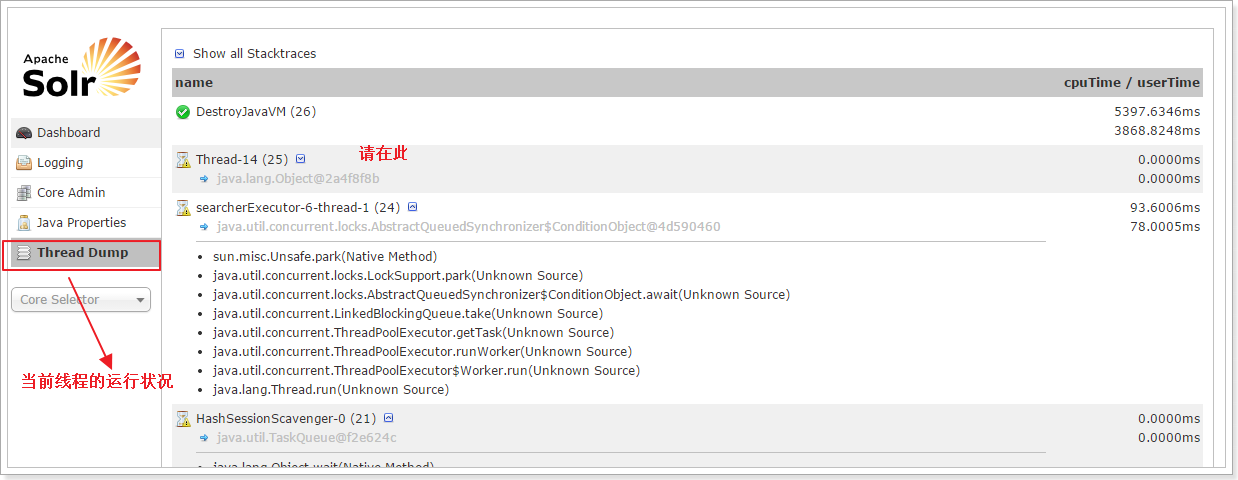
</env-entry>

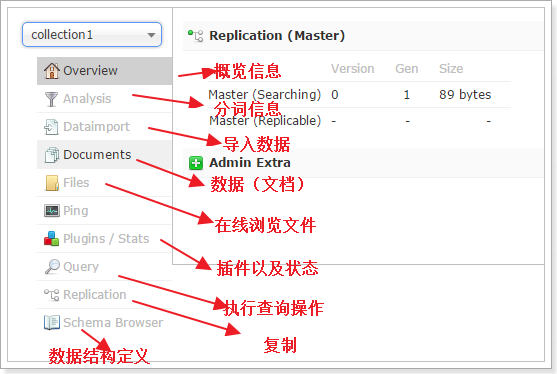
# Solr界面功能











# 安装中文分词器

## 安装步骤

### 第一步：配置IKAnalyzer的jar包

拷贝IKAnalyzer的文件到Tomcat下Solr目录中

将IKAnalyzer2012FF\_u1.jar拷贝到 Tomcat的webapps/solr/WEB-INF/lib 下。

### 第二步：IKAnalyzer的配置文件

在Tomcat的webapps/solr/WEB-INF/下创建classes目录

将IKAnalyzer.cfg.xml、ext\_stopword.dic mydict.dic copy到 Tomcat的

webapps/solr/WEB-INF/classes

**注意：ext\_stopword.dic 和mydict.dic必须保存成无BOM的utf-8类型。**

### 第三步：修改schema.xml文件

修改schema.xml文件

修改Solr的schema.xml文件，添加FieldType：

|  |
| --- |
| <fieldType name="text\_ik" class="solr.TextField">  <analyzer class="org.wltea.analyzer.lucene.IKAnalyzer"/>  </fieldType> |

### 第四步：设置业务系统Field

设置业务系统Field

|  |
| --- |
| <field name="item\_title" type="text\_ik" indexed="true" stored="true"/>  <field name="item\_sell\_point" type="text\_ik" indexed="true" stored="true"/>  <field name="item\_price" type="long" indexed="true" stored="true"/>  <field name="item\_image" type="string" indexed="false" stored="true" />  <field name="item\_category\_name" type="string" indexed="true" stored="true" />  <field name="item\_desc" type="text\_ik" indexed="true" stored="false" />  <field name="item\_keywords" type="text\_ik" indexed="true" stored="false" multiValued="true"/>  <copyField source="item\_title" dest="item\_keywords"/>  <copyField source="item\_sell\_point" dest="item\_keywords"/>  <copyField source="item\_category\_name" dest="item\_keywords"/>  <copyField source="item\_desc" dest="item\_keywords"/> |