SolrCloud的搭建及使用

1. 工具包准备
2. 服务容器 Zookeeper zookeeper-3.4.6
3. Solr-6.1.0

下载地址：\\Sf00591793a\分享内容\SolrCloud搭建及使用

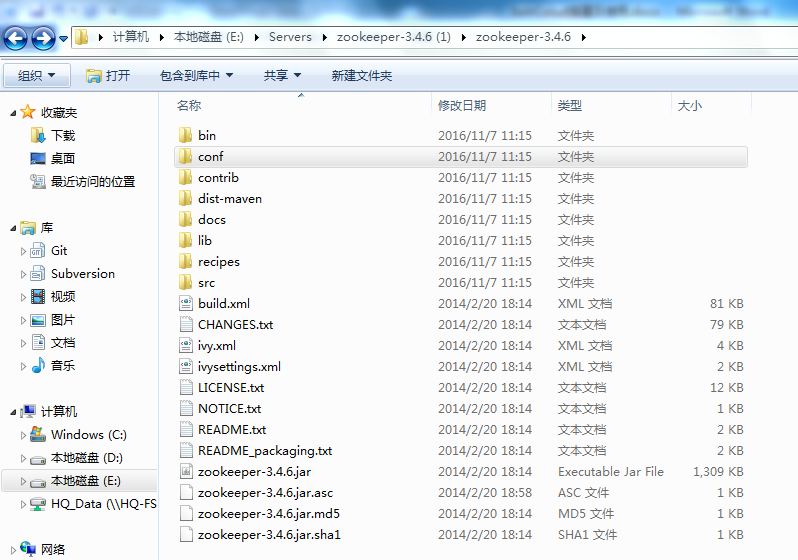
2、 环境准备

以下为windows环境的搭建过程，linux环境大部分相似可自行实践。

### Zookeeper单机伪集群搭建：

#### 1、解压Zookeeper压缩包

解压后可直接bin目录启动zkServer.cmd 但为单机模式，需要配置集群模式。



#### 2、新建配置文件

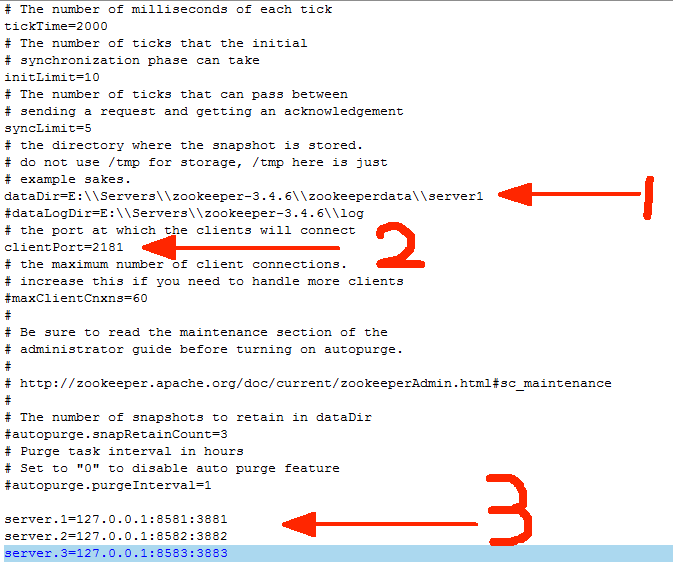
进入conf目录 ，复制zoo\_sample.cfg 并重命名为zoo1.cfg

按如下图重新配置zoo1.cfg，其中

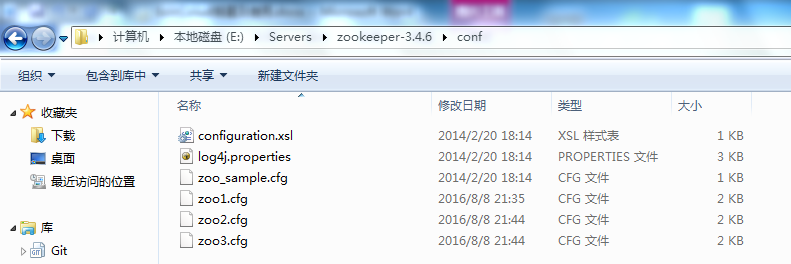
1代表修改log日志存放路径（需要唯一）

2代表端口号（需要唯一）

3集群配置（相同）



重复第2步操作，满足奇数个即可，以下新建了zoo2.cfg，zoo3.cfg。

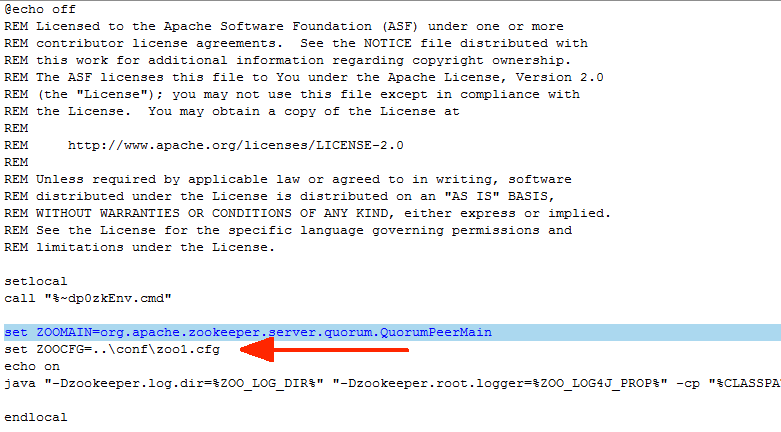


#### 3、新建启动文件

进入bin目录 新建启动配置。复制zkServer.cmd并重命名为zkServer1.cmd

新增 set ZOOCFG=..\conf\zoo1.cfg 这一行

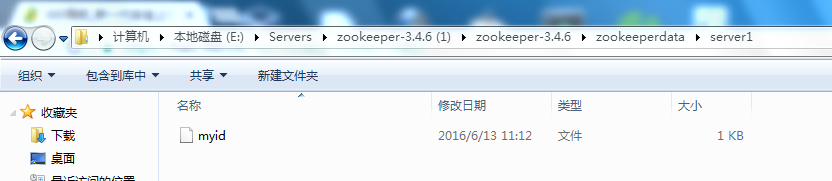
同理，复制第三步，新建zkServer2.cmd，zkServer3.cmd，并指向对应不同的配置文件zoo2.cfg，zoo3.cfg。



#### 4、新建日志文件目录

ZK目录下新建文件夹 命名为/zookeeperdata/serverX（与zoo.cfg里的配置统一）

并在各个文件夹里新建myid文件，分别填入1、2、3。



#### 5、启动Zookeeper集群

进入bin目录，分别启动zkServer1.cmd，zkServer2.cmd，zkServer3.cmd。

连接Zookeeper客户端命令: ZkCli.cmd –server 127.0.0.1:2181

出现下图则代表ZK集群启动成功。

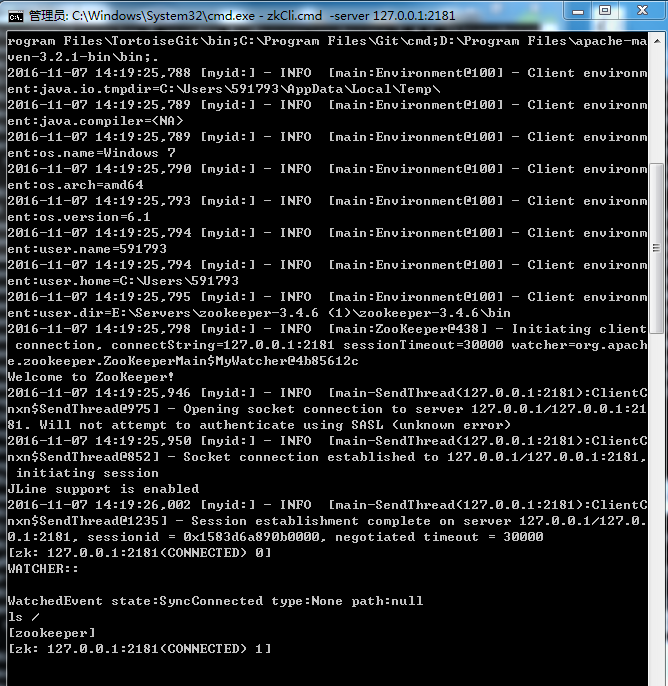
客户端查看zookeeper节点信息命令: ls / or ls2 /

查看节点内容命令: get /XXX/XXX

删除节点命令： delete /XXX

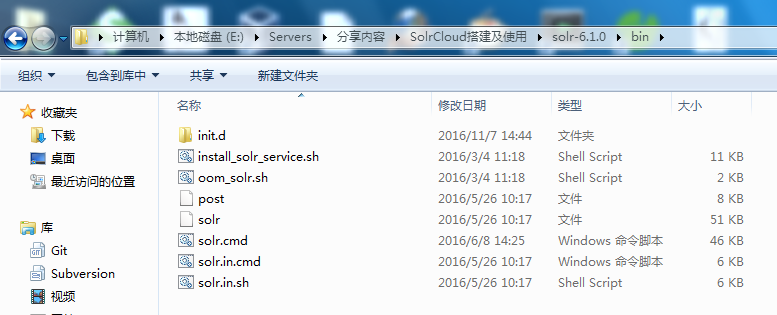
设置节点内容命令： set /zk "zkbak"

查看ZK状态命令: zkServer.cmd status

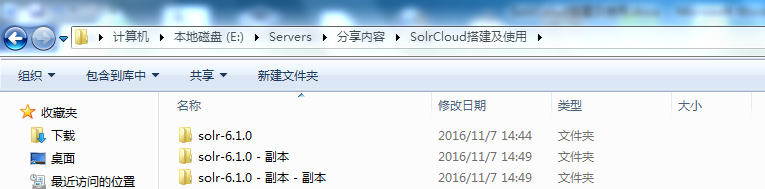


### Solr集群搭建

#### 1、解压Solr压缩包



复制solr-6.1.0文件夹2份，如下图。



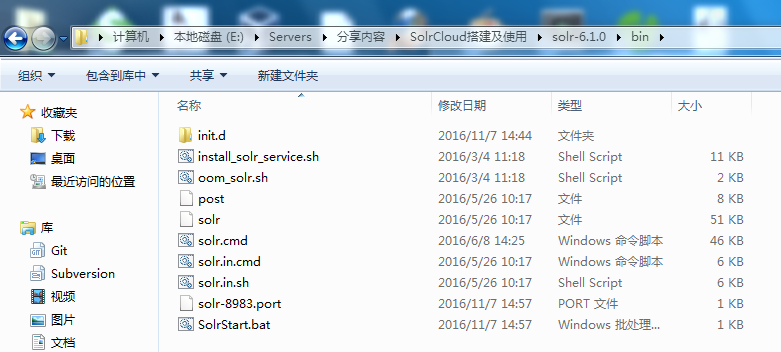
#### 2、新建Solr启动文件

进入bin目录，新建solr启动bat文件，也可以命令启动。

命令：solr start -c -z 127.0.0.1:2181 -p 8983

127.0.0.1:2181 代表zk集群配置的节点

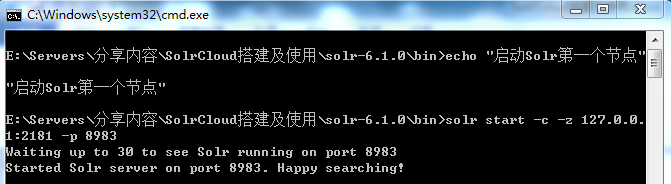
8983 代表solr占用的端口



同上，在其他solr文件夹的bin目录新建启动方式，注意修改Zookeeper的节点与solr端口号。

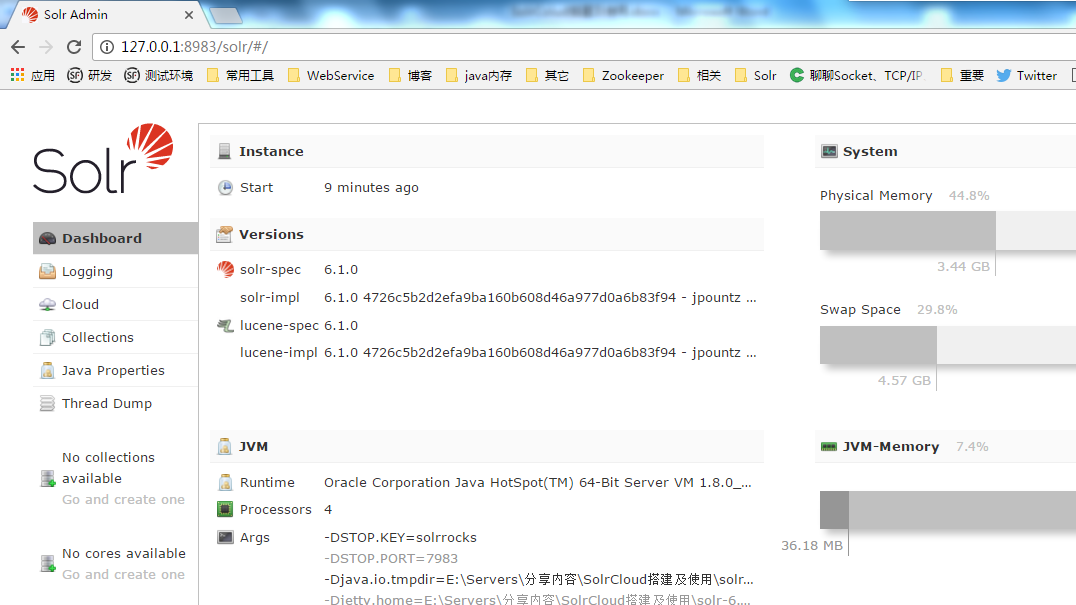
#### 3、启动Solr

点击启动SolrStart.bat



浏览器输入<http://127.0.0.1:8983/solr/#/>

出现如下Solr的管理端界面，则代表启动成功。



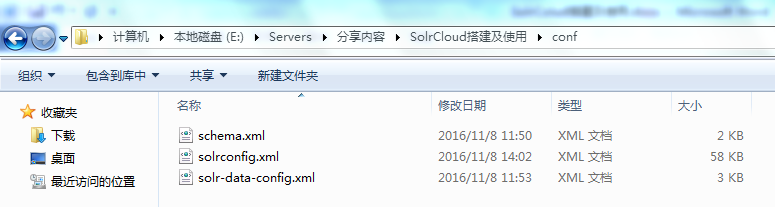
3、 基础开发

完成上述所有准备工作后，后面可以针对具体的需求进行开发。

本文档针对客户档案全网查询为例。

### 1、配置文件

新建配置文件夹conf，并从Solr提供的样例中获取配置文件solrconfig.xml、solr-data-config.xml到conf文件夹中。

在Solrconfig.xml删除依赖未用到的部分（不删solr建core会报错）：

<searchComponent name=”elevator” class="solr.QueryElevationComponent" >

<requestHandler name="/elevate" class="solr.SearchHandler" startup="lazy">

2、修改数据库信息配置文件solr-data-config.xml：



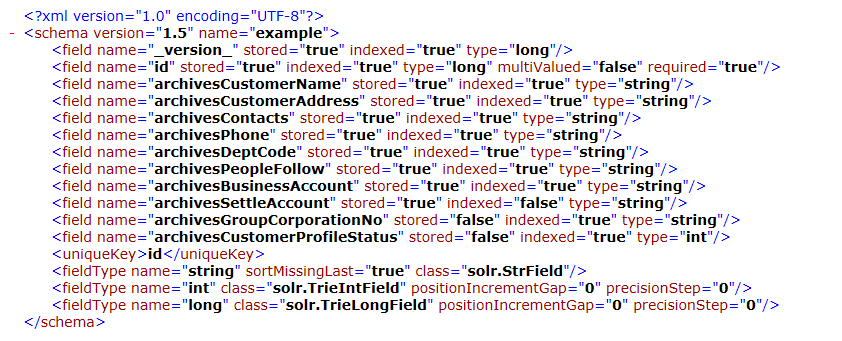
Query: 查询数据库表符合记录数据 (全量)

DeltaQuery: 增量索引查询主键ID

DeltaImportQuery: 增量索引查询导入的数据

核心思想是：通过内置变量“${dih.delta.id}”和 “${dataimporter.last\_index\_time}”来记录本次要索引的id和最近一次索引的时间。

3、新建schema.xml文件，针对solr-data-config.xml匹配对应的字段。



### 2、上传配置文件

进入solr的bin目录，上传第一步的配置文件至Zookeeper中。

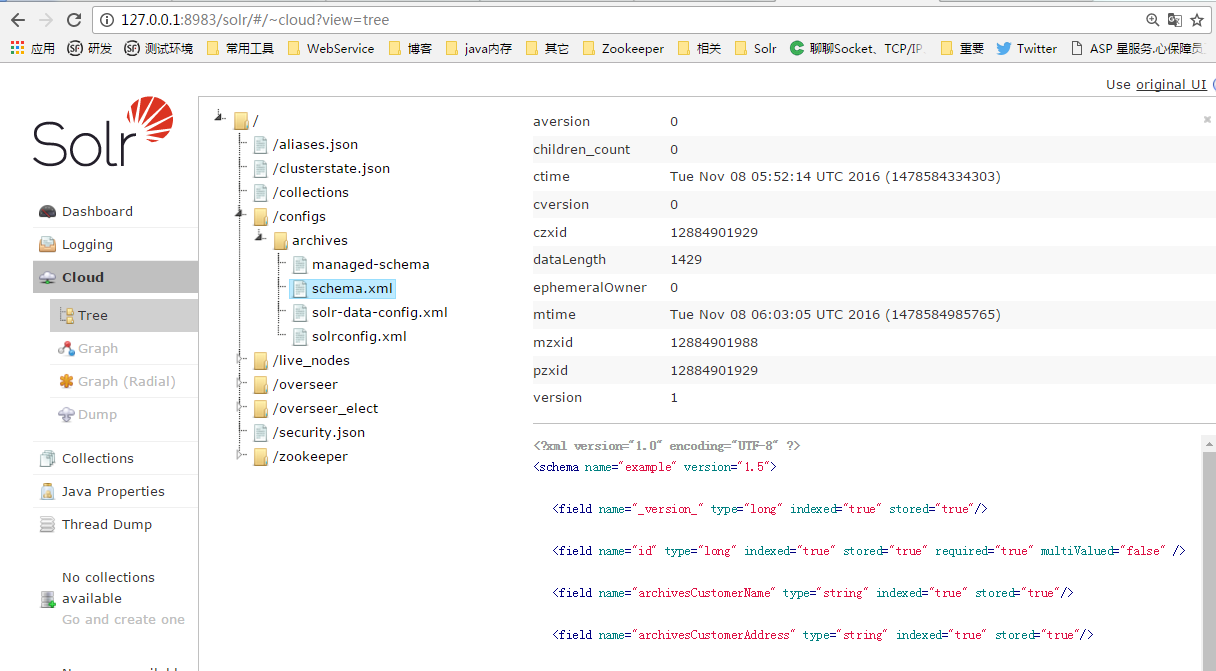
命令：

solr zk -upconfig -d E:\Servers\分享内容\SolrCloud搭建及使用\conf -n archives -z 127.0.0.1:2181,127.0.0.1:2182,127.0.0.1:2183

solr zk -upconfig -d d:\user\01083449\桌面\技术手册\技术收藏整理\solr\conf -n opportunities -z 127.0.0.1:2181,127.0.0.1:2182,127.0.0.1:2183

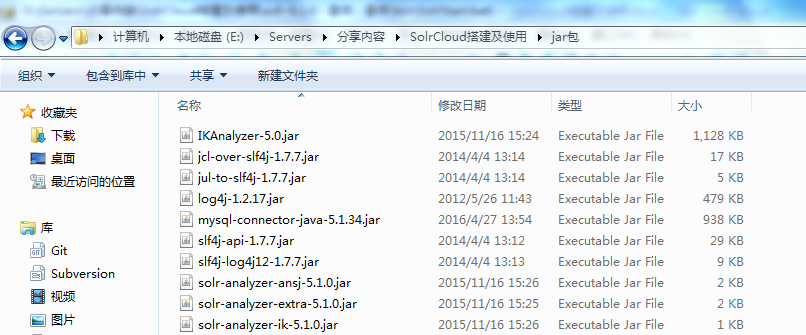
solr zk -upconfig -d d:\user\01083449\桌面\技术手册\技术收藏整理\solr\archives -n archives -z 127.0.0.1:2181,127.0.0.1:2182,127.0.0.1:2183

通过登录solr管理端或者通过zk客户端查看是否上传成功



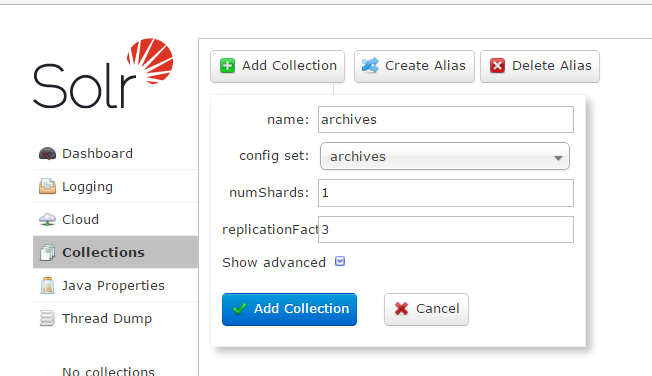
### 3、加入jar包

进入solr-6.1.0\server\solr-webapp\webapp\WEB-INF\lib目录，加入以下必须的jar包。

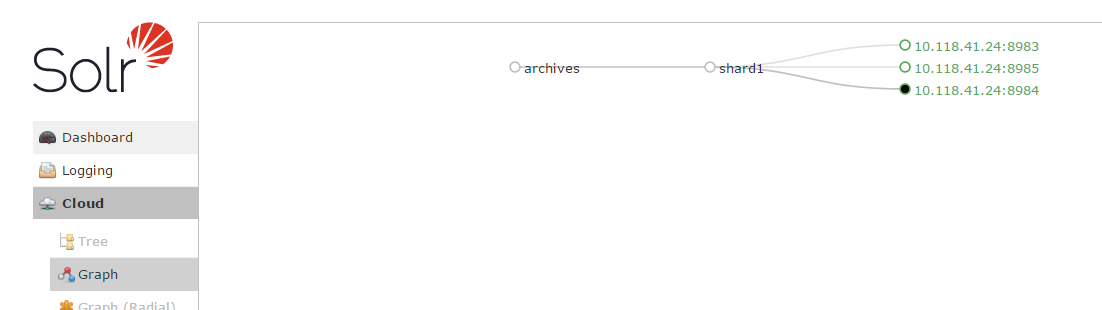


### 4、新建Core

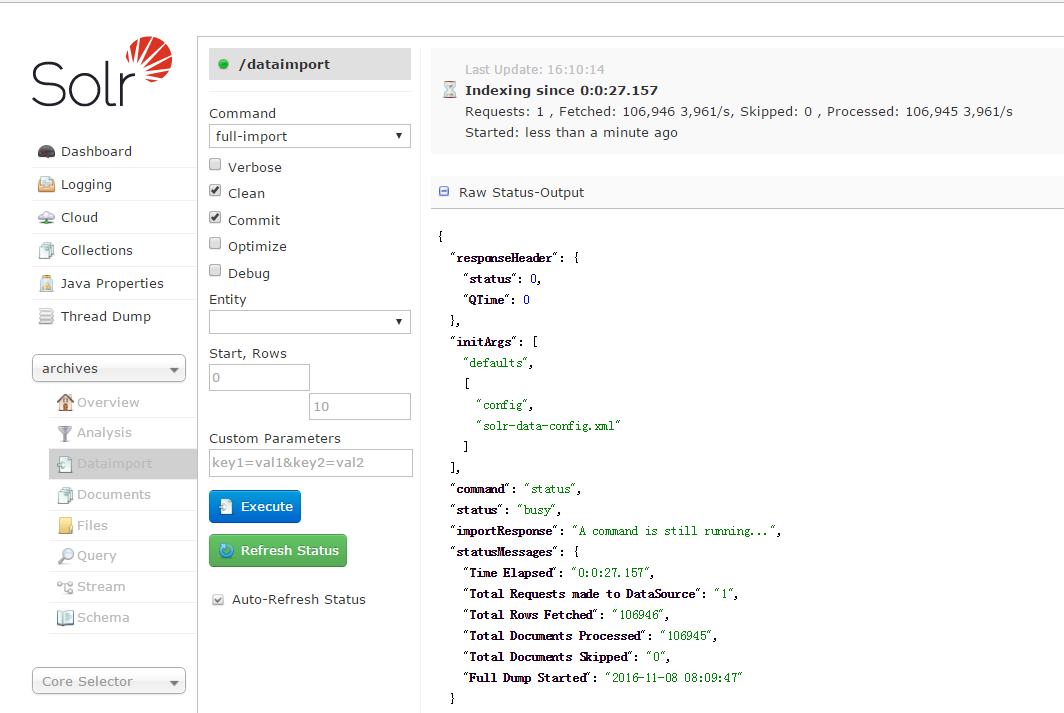
新建archives命名的core，其中建3个副本。



新建后，查看是否创建成功。



### 5、全量导入数据



### 6、扩展项

#### 中文分词

使用开源的IKAnalyzer分词包，lib中引入必须的jar包，详见第3点。

修改schema.xml，新增中文分词字段类型。

<fieldType name="text\_ik" class="solr.TextField">

<analyzer type="index">

<tokenizer class="org.apache.lucene.analysis.ik.IKTokenizerFactory" useSmart="false"/>

<filter class="solr.LowerCaseFilterFactory"/>

<filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true" expand="false"/>

</analyzer>

<analyzer type="query">

<tokenizer class="org.apache.lucene.analysis.ik.IKTokenizerFactory" useSmart="false"/>

<filter class="solr.LowerCaseFilterFactory"/>

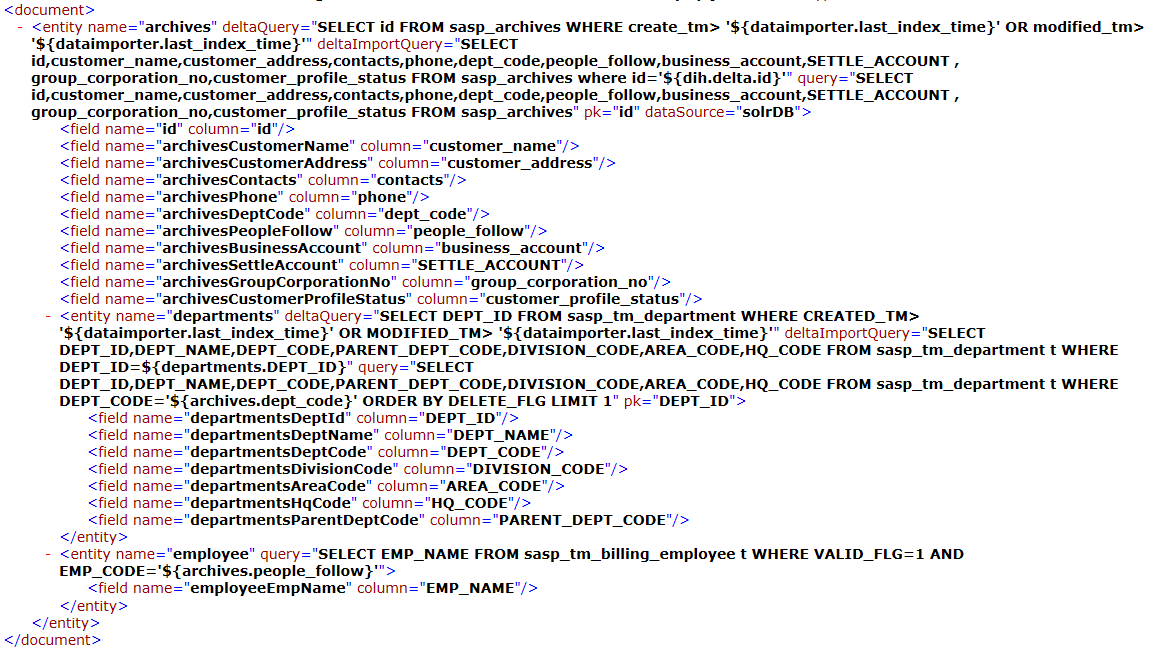
<filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true" expand="true"/>

</analyzer>

</fieldType>

#### 表关联配置

Solr-data-config.xml:



#### 增量更新

1、引入apache-solr-dataimportscheduler的jar包。

2、进入solr-6.1.0\server\solr-webapp\webapp\WEB-INF 目录，修改WEB-INF目录下面的web.xml文件，添加

<listener>

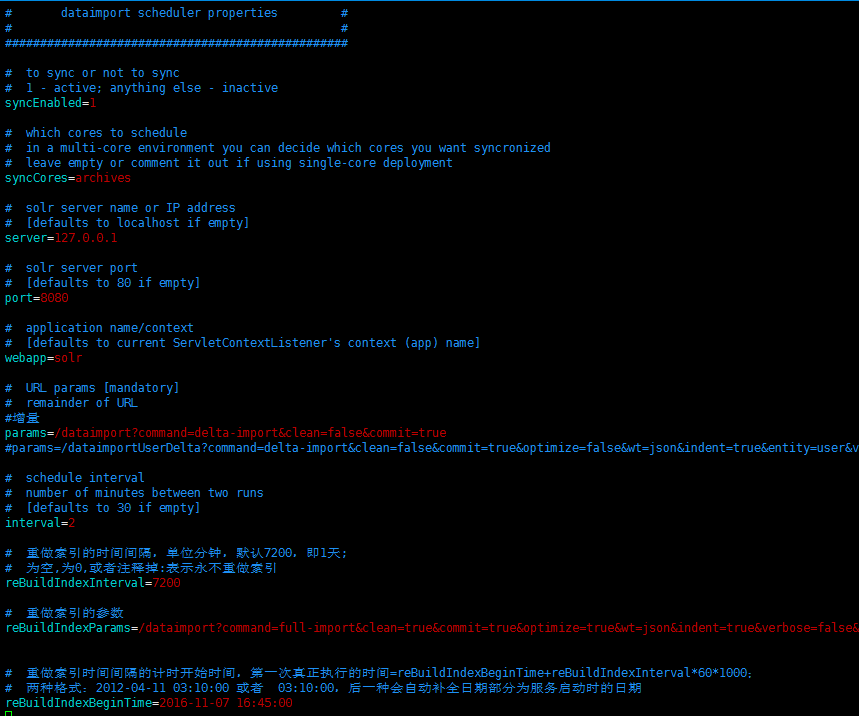
<listener-class>

org.apache.solr.handler.dataimport.scheduler.ApplicationListener

</listener-class>

</listener>

3、在solr-6.1.0\server\solr目录下新建conf文件夹以及配置文件dataimport.properties。



#### Java后台交互

后台引入SolrJ的jar包：

<dependency>

<groupId>org.apache.solr</groupId>

<artifactId>solr-solrj</artifactId>

<version>6.1.0</version>

</dependency>

具体代码详见Base包案例。