Apache Solr的安装和使用

# 1.Solr 部署和使用

1. 下载Solr

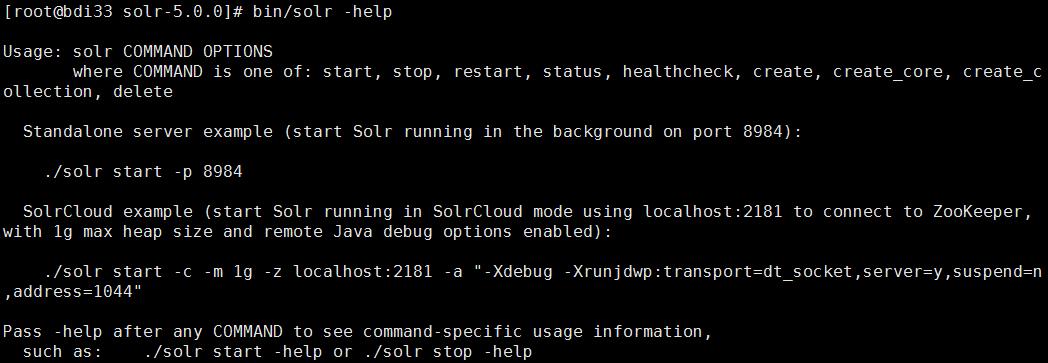
$wget http://archive.apache.org/dist/lucene/solr/5.0.0/solr-5.0.0.tgz

1. 解压

$cd ~/

$tar zxv solr-5.0.0.tgz

1. 命令



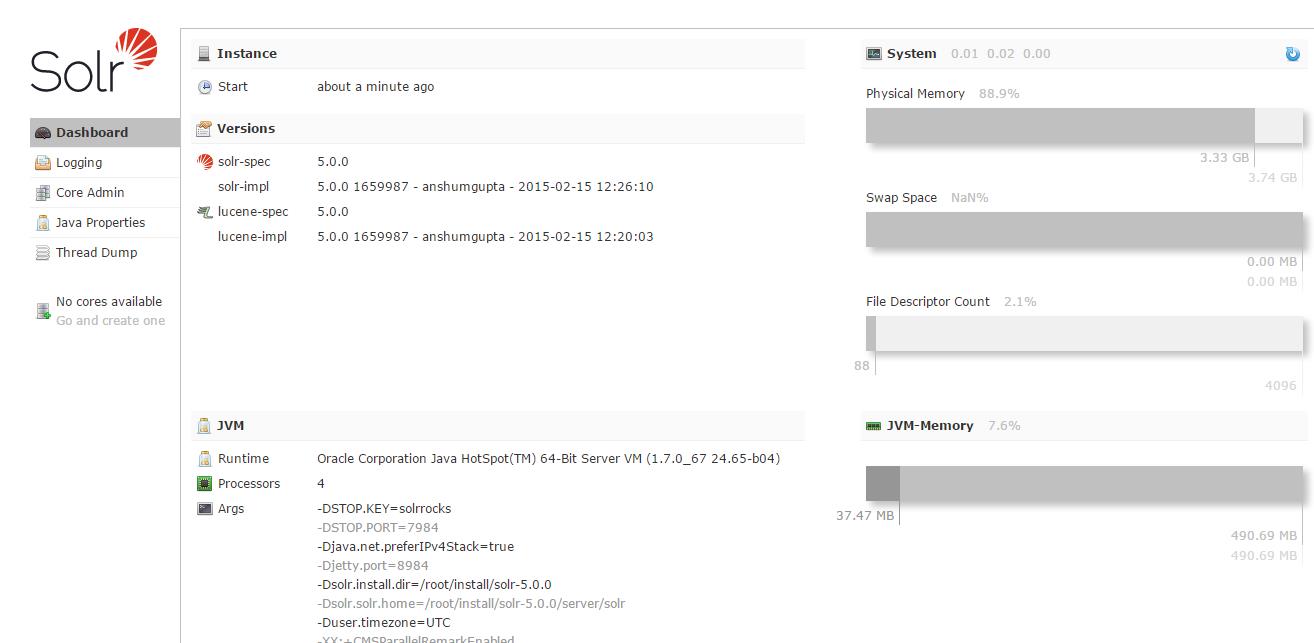
1. 启动

端口为8984

$bin/solr start -p 8984

5）Web admin UI

登录URL为http://bdi33.cmss.com:8984/solr/#/，如下图所示：



6）创建Core

# bin/solr create -c hellosolr

输出如下：

Setup new core instance directory:

/root/install/solr-5.0.0/server/solr/hellosolr

Creating new core 'hellosolr' using command:

http://localhost:8984/solr/admin/cores?action=CREATE&name=hellosolr&instanceDir=hellosolr

{

"responseHeader":{

"status":0,

"QTime":2013},

"core":"hellosolr"}

命令执行后，在目录solr-5.0.0/server下生成hellosolr所对应的目录，有以下几个文件

|  |
| --- |
| |solr.xml //定义solr server实例的配置项  |zoo.cfg //solrCloud |
| hellosolr/ |
| |-- conf  |--.currency.xml  ..... |---elevate.xml  |---managed-schema //索引document信息，可以定义docement的fields集，包括field type等，影响solr索引的策略  |---params.json  |---protwords.txt  |---solrconfig.xml //数据目录的位置  |----synonyms.txt  |---stopwords.txt  |---lang |
| |-- core.properties //core属性如索引目录|配置文件|core名称及其他选项 |
| |-- data  |--index  --- |---segments\_1  --- |---write.lock  ...... |--tlog |

7）添加documents

Solr可以查询到匹配查询条件的文档， scheme定义了内容如何组织。在solr/examples目录下有多种类型的文件。bin目录下的post命令用于上传不同类型的数据，包括原生XML，JSON，CSV甚至web crawl数据。

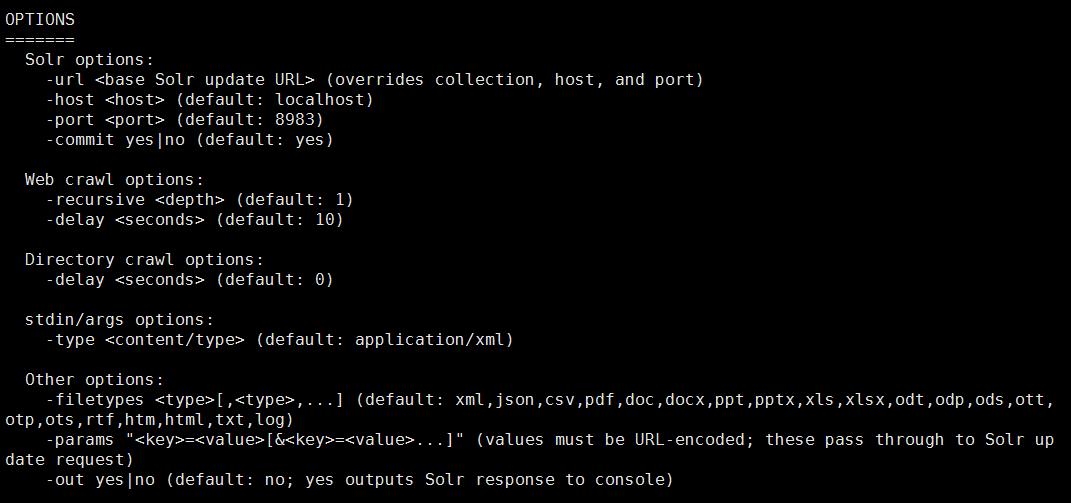
命令如下：

$bin/post -help

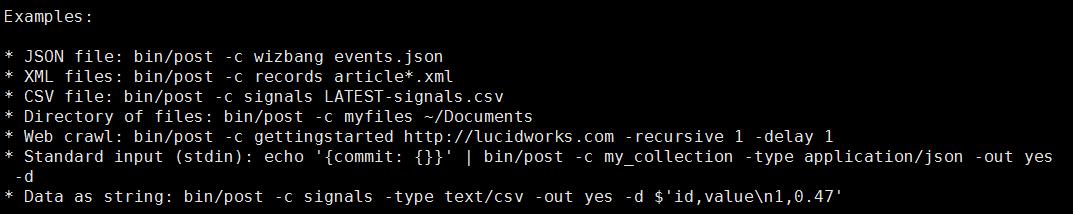
Usage: post -c <collection> [OPTIONS] <files|directories|urls|-d ["...",...]>

or post -help

collection name defaults to DEFAULT\_SOLR\_COLLECTION if not specified



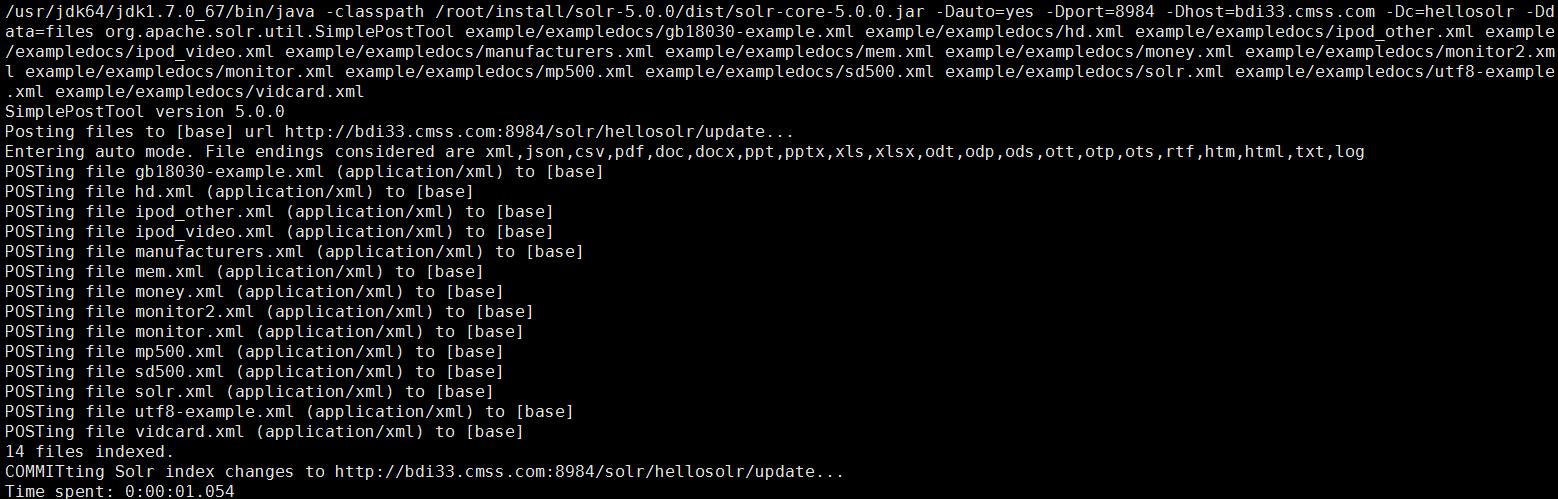
下面是几个上传的例子：



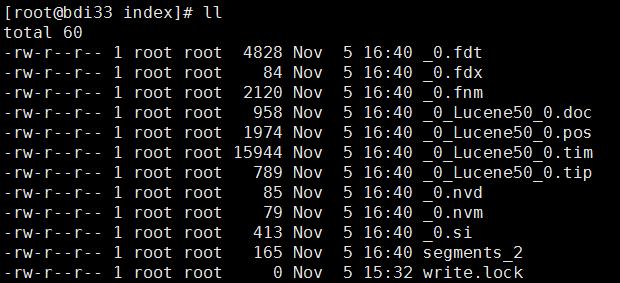
实例：

$bin/post -c hellosolr -port 8984 -host bdi33.cmss.com example/exampledocs/\*.xm

运行结果如下：



Solr为这些文件建立了索引，在目录:${solr}/server/solr/hellosolr/data/index下生成了索引文件，如下所示：



8）查询

通过前面几个步骤，已经建立了索引后的文档，可以进行查询，最简单的方式是通过创建URL，例如，查询filed中含有"video"的所有文档

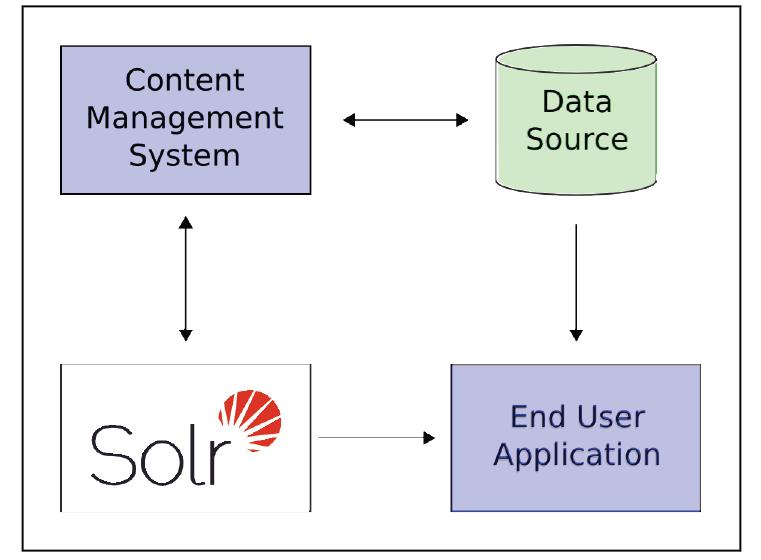
wget http://bdi33.cmss.com:8984/solr/hellosolr/select?q=video

结果如下：



# 2.Solr简介

下图是Solr 在应用中的使用



在图中，Solr在Server Application之外运行，例如网上商店提供用户接口，Shopping Cart，为用户购物提供便利。Inventory管理应用运行store职员修改产品信息。产品信息存储在数据库中（同时也在Solr存储信息）。使用Solr可以简化产品的搜索：

1）Schema，Solr从schema中获取如何对文档进行索引，以在线商店为例，schema定义了产品名字，描述，price，制造商等fields。

2） 在应用Server上部署Solr

3） 将应用要搜索的文档，feed给Solr

4）.为应用提供搜索功能

Solr的查询操作是RESTful，使用URL进行HTTP访问，响应结果是结构化文档，一般是XML,也可以是JSON,CSV及其他格式。

下面是Solr的基本概念

1. Collection

SolrCloud集群中逻辑意义上的完整的索引，常被划分为一个或者多个Shard，使用相同的Config Set，如果Shard数超过一个，就是分布式索引

1. Shard

Collection的逻辑分片，每个Shard被化为一个或者多个replicas，通过选举确定哪个是Leader

1. Core

在standalone模式下，Core是指代一个单独的索引、关联事务日志以及一些配置文件（包括schema.xml和solrconfig.xml等）的单元。在安装Solr时，可以创建多个core，多core模式允许你在同一台机器上使用不同的结构来索引数据，以及在如何把数据展现给不同的用户方面提供更多的控制。

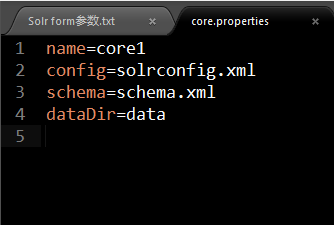
core.properites属性文件中指定一些core属性，如索引目录,配置文件,core名称以及其他选项。Solr会在安装目录下发现core.properties文件

Core拥有自己独立的索引以及事务日志，每个Core拥有自己独立的Core目录，拥有自己的独立Core访问URL(http://ip:port/solr/corename)，这种设置为solr distributed searching 提供便利。不同的Core可以在同一台Server上，当然Core也为索引数据分类提供便利，避免单个索引目录下的数据体积过大。

下面是core.properties的配置：

|  |  |
| --- | --- |
| 配置项 | 含义 |
| name | Core的名称 |
| config | 用于指定solrconfig.xml配置文件的文件名，默认是solrconfig.xml |
| dataDir | 用于存放lucene索引和log日志文件的目录路径，该路径相对于core根目录，默认是当前core目录下的data |
| ulogDir | 配置事务日志文件路径，默认值是当前core目录下的data目录下的tlog目录 |
| schema | 配置schema.xml配置文件的文件名，必须存放在当前core目录的conf下 |
| shard | 用来设置当前core的分片id，可选参数 |
| Collection | 配置当前core属于哪个solrCloud Collection |
| LoadOnStartup | 是否在solr启动部署时加载core并且为core创建一个新的IndexSearcher实例 |
| transient | 设置当solr的transient-cacheSize的阈值达到限制值的时候，是否自动卸载当前core |

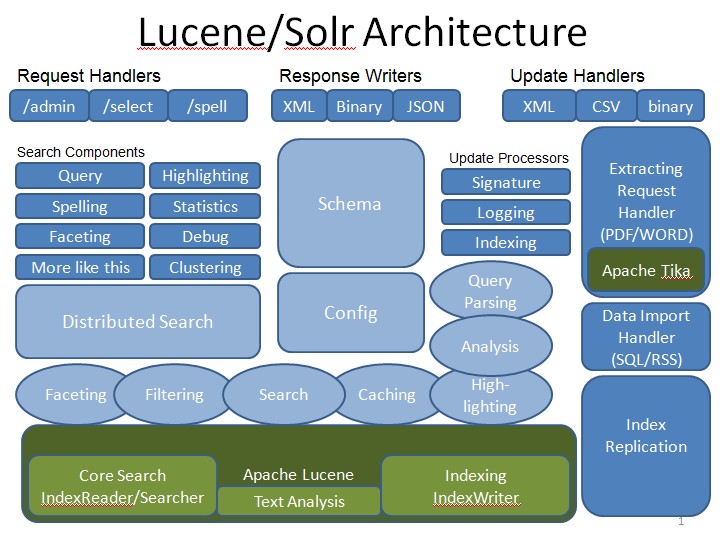
下图是一个简单的core.properties配置示例：



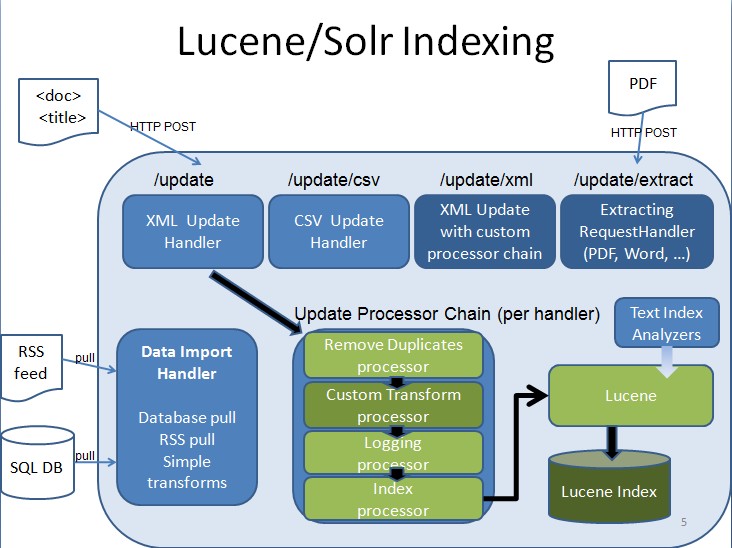
solr在启动部署的时候，首先会加载core.properties配置文件，通过core.properties配置来自动发现core，然后根据里面配置的config和shema来加载solrconfig.xml和schema.xml这两个核心配置文件。

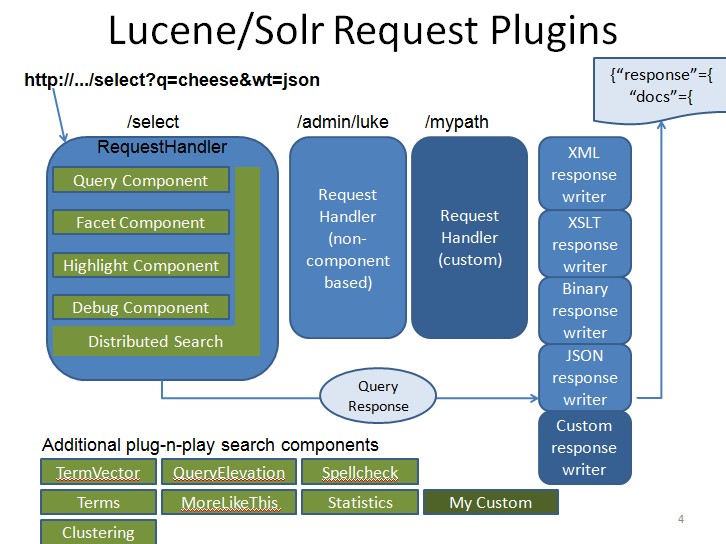
Solr的基本处理单元是documents（描述实体的数据集合）,例如描述一个人，需要包括姓名，生平，喜欢的颜色及鞋码等。描述书，需要包括标题，坐在，出版时间，页码等。Document又由fields组成，fields描述具体的某项信息，例如鞋码可以是一个filed，Fields可以包含多种类型的数据，例如Name Filed是字符串，Shoe size是浮点数，可以通过filed type来告诉Solr如何解读该field，及如何查询。当载入一个Document时，Solr从fields中获取信息，并将信息添加到index中，查询时从索引中获取匹配的文档。

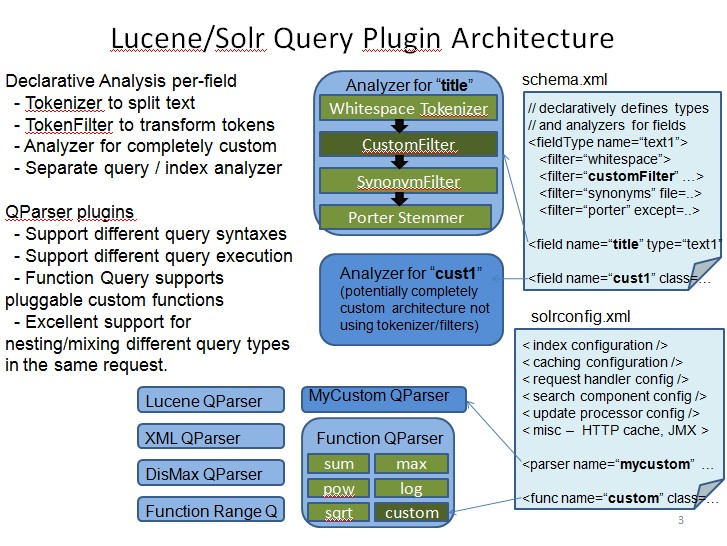
Lucene/Solr的架构如下所示：



Solr建立索引过程：







参考文献：http://www.solr.cc/blog/?p=167

# 3.Solr 与Hadoop

## 3.1 Solr索引存储在HDFS上

Solr支持从HDFS文件系统中读取index文件和transation log 文件。但是不用Hadoop MR程序来处理Solr数据，仅将Hadoop作为存储使用。

hadoop的版本要使用2.x的版本，通过配置solrconfig.xml来指定使用的文件系统，使用HdfsDirectoryFactory，配置工厂类用于从某文件系统中读写数据。data dir配置为:hdfs://host:port/path，log的目录为hdfs://host:port/pathc，facotry的类型配置为hdfs

|  |  |
| --- | --- |
| 配置项 | 配置 |
| solr.directoryFactory | HdfsDirectoryFactory |
| solr.lock.type | hdfs |
| solr.data.dir | hdfs://host:port/path |
| solr.updatelog | hdfs://host:port/path |

关于HdfsDirectoryFactory的配置项，如下所示：

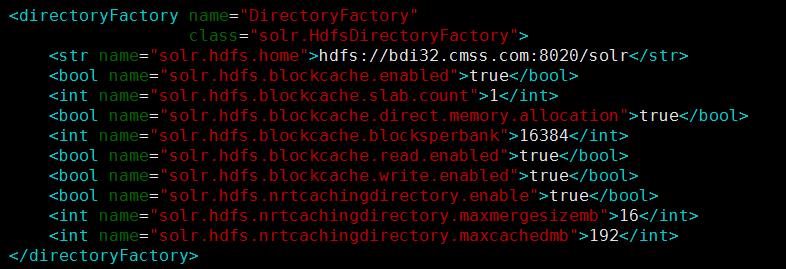
|  |  |  |
| --- | --- | --- |
| 参数 | 描述 | 配置值 |
| solr.hdfs.home | HDFS目录，用于Solr写入数据，包括数据和日志 | hdfs://host:port/path/solr |
| solr.hdfs.blockcache.enabled | 启用blockcache | true |
| solr.hdfs.blockcache.read.enabled | 启用read cache | true |
| solr.hdfs.blockcache.write.enabled | 启用write cache | true |
| solr.hdfs.blockcache.direct.memory.allocation | Memory分配的启动，false，使用堆 | true |
| solr.hdfs.blockcache.slab.count | 1 | Slab的数目，每个slab大小为128M |
| solr.hdfs.blockcache.global | false | 使用全局cache |
| solr.hdfs.nrtcachingdirectory.enable | true | 是否使用NRTCachingdirecotry |
| solr.hdfs.nrtcaching.maxmergesizemb | 16 | Merge的最大segement大小 |
| solr.hdfs.nrtcachingdirectory.maxcachemb | 192 | NRTCachingDirectory max cache大小 |
| solr.hdfs.confdir |  | HDFS Client 配置文件的目录 |

实践如下：

1）创建core

# bin/solr create -c hello

修改配置文件solrconfig.xml，在目录../hello/conf中



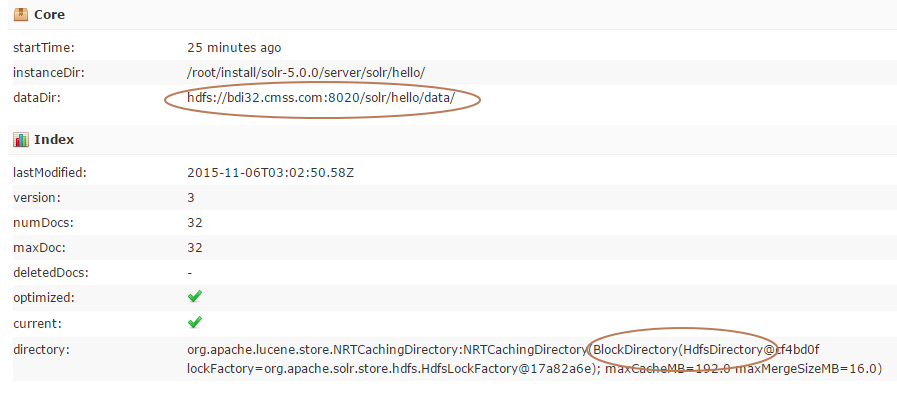
配置lock.type为hdfs

C:\Users\yshe\Documents\Fetion\1345511229\temp\f157d84b78c9e2075ca71edb5a8471a2.jpg

2）重启solr

$bin/solr restart

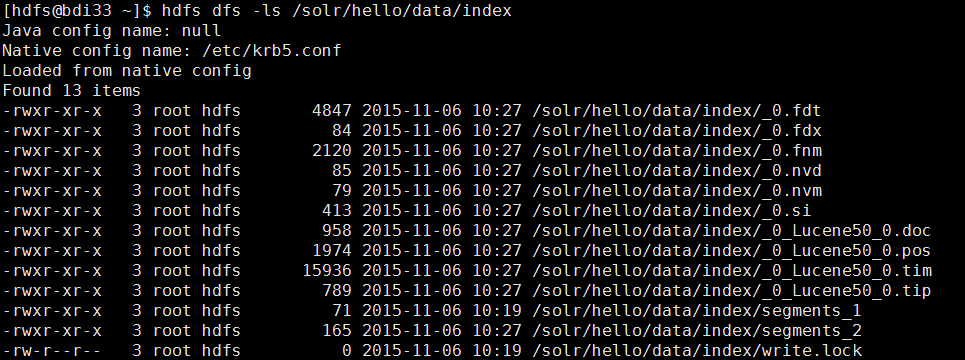
3）查看hello的配置



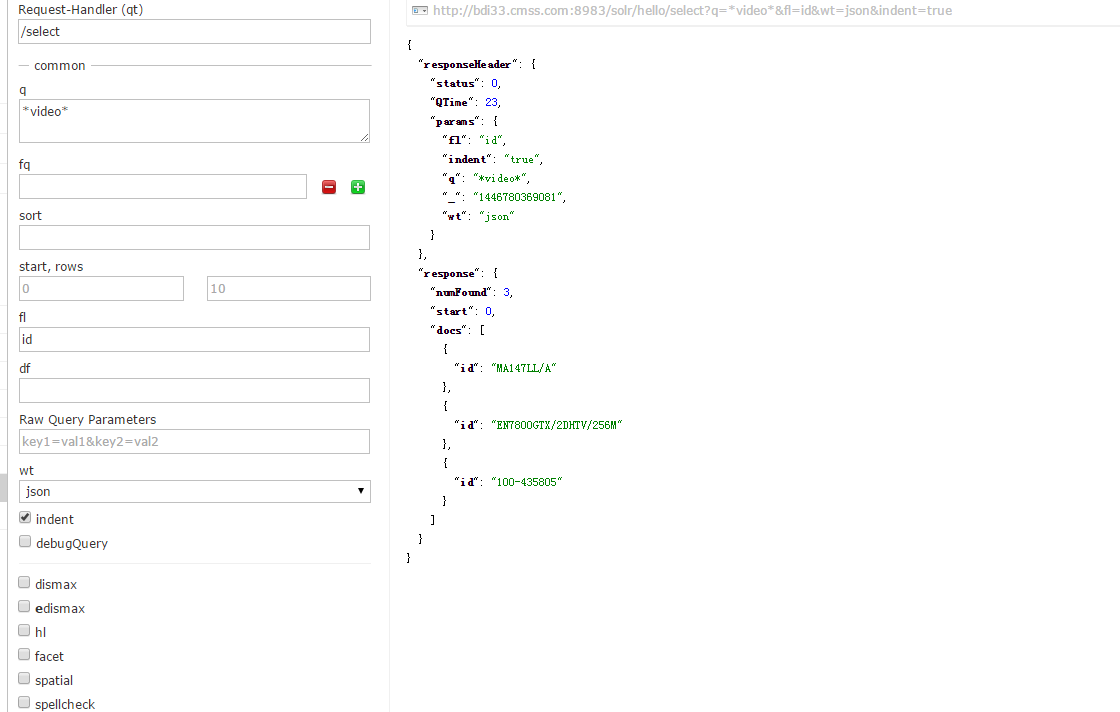
4）添加documents

$bin/post -c hello example/exampledocs/\*.xml

执行结果，在hdfs的配置目录中生成索引

****

5）执行查询



查询包含字符串video的文档，共有三个，图中列举了文档的id field。

注:测试solr版本为5.0.0，hadoop版本为bc1.2.0

相同步骤，测试5.3.0,hadoop版本bc1.2.0

参考文献：

https://cwiki.apache.org/confluence/display/solr/Running+Solr+on+HDFS

## 3.2 Solr从HDFS中读取Document

## 3.3 Solr使用MR创建索引

相关类：

MapReduceIndexerTool

参考文献：

http://blog.csdn.net/itbasketplayer/article/details/41040805

MapReduce Batch Indexing Reference

http://www.cloudera.com/content/www/en-us/documentation/archive/search/1-3-0/Cloudera-Search-User-Guide/csug\_mapreduceindexertool.html

Solr Index use MapReduce

[[SOLR-1045] Build Solr index using Hadoop MapReduce](http://www.baidu.com/link?url=HnZ0Qm_cTEJ_XqIhCqprMXrd8CLH3TBwk2QHUJj_l9i3L7vDt-gPEN_O2wkjMswbY8AU7-FFtpioxOTLjuDo9K&wd=&eqid=a8a09a3a0001ea6f00000004563c4a5e)

https://cwiki.apache.org/confluence/display/solr/Running+Solr+on+HDFS

http://zh.hortonworks.com/hadoop-tutorial/searching-data-solr/

# 4.SolrCloud

# 5.问题

问题1：

文档数据放在哪个目录，是否可以放到HDFS中，索引数据放在哪个目录

目标文件位于Hdfs上，索引位于本地目录

目标文件位于hdfs上，索引位于hdfs

目标文件本地目录，索引hdfs

问题2：



java -Dsolr.directoryFactory=HdfsDirectoryFactory -Dsolr.lock.type=hdfs -Dsolr.data.dir=hdfs://bdi32.cmss.com:8020/solr -Dsolr.upldatelog=hdfs://bdi32.cmss.com:8020/solr -jar start.jar

Error creating core [testhdfs]: Lock obtain timed out: org.apache.solr.store.hdfs.HdfsLockFactory$HdfsLock@5a38d2d8

org.apache.solr.common.SolrException: Lock obtain timed out: org.apache.solr.store.hdfs.HdfsLockFactory$HdfsLock@5a38d2d8

# 5.参考文献

Solr架构：http://www.solr.cc/blog/?p=167

Solr Core: http://iamyida.iteye.com/blog/2211222