**用户信息搜索集群配置指导**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 版本发布日期 | 修订人 | 修订内容描述 |
|  |  |  |  |

目录

一、搜索集群架构简介 2

1.1整体功能描述 2

1.2拓扑结构 2

二、zookeer集群配置 3

2.1下载并安装zookeeper 3

2.2配置zookeeper 3

2.2.1配置zookeeper集群服务器列表 3

2.2.2配置zookeeper服务器日志 3

2.2.3配置zookeeper服务器的标识 4

2.3启动zookeeper服务 4

三、Solrcloud集群配置 4

3.1下载并安装solr 4

3.2配置solr日志文件 4

3.3配置搜索服务 5

3.3.1配置solr服务器地址和监听端口 5

3.3.2配置搜索文档集合usercore 5

3.3.3配置core集合 5

3.3.4配置搜索模式 5

3.3.4配置搜索运行参数 6

3.4配置resin 6

3.4.1配置resin属性文件 6

3.4.1配置resin启动文件 7

3.4.3配置第一个solr服务器的resin启动脚本 8

3.4.4配置第二个及后续solr服务器的resin启动脚本 8

3.5索引管理 8

3.6管理solr控制台 8

四、Search服务器配置 9

# 一、搜索集群架构简介

## 1.1整体功能描述

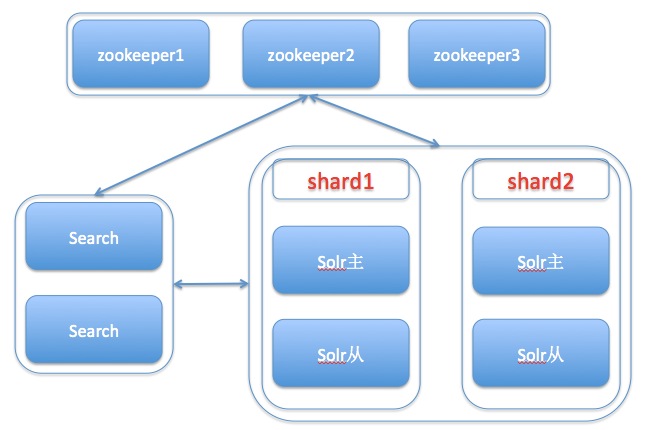
通过独立的zookeeper集群管理solrcloud集群。

Solrcloud集群负责索引的创建、查询、更新和删除操作。Solrcloud集群做sharding，每个sharding内做主从备份。

Search服务通过zookeeper集群获取solrcloud信息，向solrcloud提交用户信息索引请求，并向solrcloud提交用户信息搜索请求以完成用户信息的搜索。

Service服务对客户端提供搜索接口，给Search服务提供用户信息获取接口，同时从Search服务中查询获取用户信息。

## 1.2拓扑结构



# 二、zookeer集群配置

## 2.1下载并安装zookeeper

下载最新发布版本zookeeper-3.4.6，并解压。

## 2.2配置zookeeper

需要配置zookeeper集群的服务器列表，同时指定本zookeeper服务器的ID。

### 2.2.1配置zookeeper集群服务器列表

在conf/zoo.cfg文件中配置zookeeper集群的服务器列表。

在文件最后增加zookeeper服务器列表

server.1=10.10.10.10:2001:3001

server.2=10.10.10.10:2002:3002

server.3=10.10.10.10:2003:3003

### 2.2.2配置zookeeper服务器日志

在conf/log4j.properties文件中配置日志文信息，指定日志文件路径。

log4j.rootLogger=info,console,file

log4j.appender.console=org.apache.log4j.ConsoleAppender

log4j.appender.console.layout=org.apache.log4j.PatternLayout

log4j.appender.console.layout.ConversionPattern=%-d{yyyy-MM-dd HH:mm:ss} [%c]-[%p] %m%n

log4j.appender.file=org.apache.log4j.DailyRollingFileAppender

log4j.appender.file.File=/data/logs/zookeeper/stdout.log

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%-d{yyyy-MM-dd HH:mm:ss} [%c]-[%p] %m%n

log4j.appender.file.Threshold=DEBUG

log4j.appender.file.DatePattern='.'yyyyMMdd

log4j.appender.file.encoding=UTF-8

### 2.2.3配置zookeeper服务器的标识

zookeeper服务器标识需要和zoo.cfg文件中定义的一致。

在data目录下创建myid文件，myid文件内容为服务器id，例如1。三个zookeeper服务器的id分别对应 为1，2和3。

如果在zookeeper安装目录下没有data目录，则手动创建。

## 2.3启动zookeeper服务

使用bin目录下的脚本来管理zookeeper服务。

启动zookeeper服务命令：./zkServer.sh start

# 三、Solrcloud集群配置

## 3.1下载并安装solr

下载最新发布版本solr-4.10.3，并解压。

从dist/solr-4.10.3.war拷贝到web主目录下并解压。假定web主目录为/data/work/solr/。

将example/lib/ext/\*.jar拷贝到/data/work/solr1/WEB-INF/lib目录下。

## 3.2配置solr日志文件

在WEB-INF目录下创建classes目录，在classes目中中新建log4j.properties文件。

log4j.properties日志文件内容：

log4j.rootLogger=info,console,file

log4j.appender.console=org.apache.log4j.ConsoleAppender

log4j.appender.console.layout=org.apache.log4j.PatternLayout

log4j.appender.console.layout.ConversionPattern=%-d{yyyy-MM-dd HH:mm:ss} [%c]-[%p] %m%n

log4j.appender.file=org.apache.log4j.DailyRollingFileAppender

log4j.appender.file.File=/data/logs/solr/stdout.log

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%-d{yyyy-MM-dd HH:mm:ss} [%c]-[%p] %m%n

log4j.appender.file.Threshold=DEBUG

log4j.appender.file.DatePattern='.'yyyyMMdd

log4j.appender.file.encoding=UTF-8

log4j.logger.org.apache.zookeeper=WARN

log4j.logger.org.apache.hadoop=WARN

# set to INFO to enable infostream log messages

log4j.logger.org.apache.solr.update.LoggingInfoStream=OFF

## 3.3配置搜索服务

在web主目录下创建home目录/data/work/solr/home，用于放置跟solr有关的配置文件。

### 3.3.1配置solr服务器地址和监听端口

配置solr服务主机名和监听端口，需要和resin启动的监听端口保持一致。

<str name="host">${host:solr}</str>

<int name="hostPort">${jetty.port:9096}</int>

<str name="hostContext">${hostContext:}</str>

### 3.3.2配置搜索文档集合usercore

在home目录下创建usercore目录，usercore的名字将作为文档集的名字，需要和core.properties文件指定的保持一致。

### 3.3.3配置core集合

在usercore/core.properties文件中指定core的collection名字为usercore。

name=usercore

### 3.3.4配置搜索模式

在usercore/conf/schema.xml文件中配置用户信息搜索用到的模式信息。

<field name="uid" type="string" indexed="true" stored="true" required="true" />

<field name="username" type="string" indexed="true" stored="true" omitNorms="true" />

<field name="avatar" type="string" indexed="false" stored="true" omitNorms="true" />

<field name="type" type="tint" indexed="true" stored="true" omitNorms="true" />

<field name="level" type="tint" indexed="true" stored="true" omitNorms="true" />

<field name="gender" type="int" indexed="true" stored="true" omitNorms="true" />

<field name="birthday" type="tint" indexed="true" stored="true" omitNorms="true" />

<field name="label" type="string" indexed="true" stored="true" multiValued="true" />

<field name="registertime" type="tint" indexed="true" stored="true" omitNorms="true" />

<field name="lasttime" type="tint" indexed="true" stored="true" omitNorms="true" />

<field name="geospatial" type="location" indexed="true" stored="true" omitNorms="true" />

<field name="location" type="text\_general" indexed="true" stored="true" omitNorms="true" />

<field name="multilocation" type="string" indexed="true" stored="false" multiValued="true" />

<dynamicField name="\*\_coordinate" type="tdouble" indexed="true" stored="false"/>

<uniqueKey>uid</uniqueKey>

### 3.3.4配置搜索运行参数

在usercore/conf/solrconfig.xml文件中配置solr运行参数。

其中，

1）配置了默认搜索字段为uid。

<str name="df">uid</str>

2）配置硬提交15秒/次（硬提交将索引持久化硬盘）

<autoCommit>

<maxTime>${solr.autoCommit.maxTime:15000}</maxTime>

<openSearcher>false</openSearcher>

</autoCommit>

3）禁用软提交（软提交仅仅使索引在内存可见）

<autoSoftCommit>

<maxTime>${solr.autoSoftCommit.maxTime:-1}</maxTime>

</autoSoftCommit>

## 3.4配置resin

resin中需要指定web主目录，并且指定日志文件路径。

### 3.4.1配置resin属性文件

在resin/conf/resin.properties文件中配置resin集群属性。

home\_cluster : solr

### 3.4.1配置resin启动文件

在resin/conf/resin.properties文件中配置resin启动参数。

<cluster id="solr">

<character-encoding>UTF-8</character-encoding>

<root-directory>.</root-directory>

<server-default>

<http address="\*" port="9096"/>

<jvm-arg>-Xms256m</jvm-arg>

<jvm-arg>-Xmx512m</jvm-arg>

<jvm-arg>-Xmn128m</jvm-arg>

<jvm-arg>-XX:MaxPermSize=128m</jvm-arg>

<jvm-arg>-XX:-UseParallelGC</jvm-arg>

<jvm-arg>-XX:ParallelGCThreads=4</jvm-arg>

<watchdog-port>9706</watchdog-port>

</server-default>

<server id="solr1" address="127.0.0.1" port="9806"></server>

<host id="" root-directory=".">

<web-app id="/" root-directory="/data/work/solr">

<character-encoding>UTF-8</character-encoding>

<servlet-mapping servlet-class='com.caucho.servlets.ResinStatusServlet'>

<url-pattern>/resin-status.servlet</url-pattern>

<init enable="read"/>

</servlet-mapping>

<env-entry>

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

<env-entry-value>/data/work/solr/home</env-entry-value>

</env-entry>

</web-app>

<stdout-log path="/data/logs/solr/stdout.log" timestamp="" rollover-period="1D"/>

<stderr-log path='/data/logs/solr/stderr.log' timestamp="" rollover-period='1D'/>

其中特别要注意到有一段solr特有的配置，用来指定solr配置文件主目录。这里指定的solr配置文件主目录必须与之前配置的solr目录保持一致，否则会因为找不到solr配置导致solr服务启动失败。

<env-entry>

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

<env-entry-value>/data/work/solr/home</env-entry-value>

</env-entry>

### 3.4.3配置第一个solr服务器的resin启动脚本

在resin/conf/resin.properties文件中配置resin启动脚本，增加启动参数。

JAVA\_OPTS="-Dbootstrap\_confdir=/data/work/solr1/home/usercore/conf -Dcollection.configName=usercoreConf -DzkHost=10.221.144.87:2181,10.221.144.87:2182,10.221.144.87:2183 -DnumShards=2"

这个启动参数中，需要指定zookeeper集群的地址，solr集群分片数量，solr配置文件主目录，以及在zookeeper集群中标记的配置名称。

修改后的启动脚本：

exec $JAVA\_EXE $JAVA\_OPTS -jar ${RESIN\_HOME}/lib/resin.jar $\*

注意必须保证solr集群中的第一个solr服务器首先启动，然后其他solr集群中的其他服务器才能启动。否则整个solr集群可能启动失败。

### 3.4.4配置第二个及后续solr服务器的resin启动脚本

在resin/conf/resin.properties文件中配置resin启动脚本，增加启动参数。

JAVA\_OPTS="-DzkHost=10.221.144.87:2181,10.221.144.87:2182,10.221.144.87:2183"

这个启动参数中，只需要指定zookeeper集群的地址，其它信息将在solr服务器连接到zookeeper集群后自动获取。

修改后的启动脚本：

exec $JAVA\_EXE $JAVA\_OPTS -jar ${RESIN\_HOME}/lib/resin.jar $\*

## 3.5索引管理

solr服务器创建的索引默认保存在文档集合目录下的data目录/data/work/solr1/home/usercore/data中。

复制或者删除索引仅需要删除这个data目录即可。

如果索引模式文件改变，例如增加或者删除字段后需要重建索引，重建之前必须删除data目录。

## 3.6管理solr控制台

登录solr服务器集群管理控制台http://usercoresolr.mele.tv:9096/solr。通过这个管理控制台可以对solr集群进行管理。

solr管理控制台建议不要对外网开放。

# 四、Search服务器配置

Search服务器配置与大部分service服务器大致一样。主要区别是在specialconfig.properties文件中需要配置zookeeper集群的地址。

solrcloud.zookeeper.zkHosts=10.221.144.87:2181,10.221.144.87:2182,10.221.144.87:2183