**Chap9 Hive基本配置实验步骤**

##### 1  安装配置JDK 1.8

**步骤1.**  **创建工作路径**

在/usr目录下建立cx工作路径。

**步骤2.**  **解压安装包**

我们可以在/usr/software/目录下找到jdk-8u60-linux-x64.tar.gz安装包，执行命令将其解压到/usr/cx/目录下。

**步骤3.**  **配置环境变量**

使用vi编辑器打开 ~/.bashrc文件，配置JDK环境变量并更新环境变量。

**步骤4.**  **验证JDK配置**

执行命令，验证JDK是否配置成功。

2  **主机名配置**

**步骤1.**  **编辑主机名**

使用vi编辑器编辑主机名配置文件/etc/sysconfig/network。

**步骤2.**  **IP地址与主机名映射文件配置**

使用vi编辑器编辑hosts文件/etc/hosts，并检测主机名与IP映射是否配置成功：

3  **SElinux安全配置**

**步骤1.**  **关闭SElinux**

CentOS默认启用了SElinux，在网络服务方面权限要求比较严格，使用vi编辑器打开SElinux配置文件/etc/selinux/config将SElinux关闭。

**步骤2.**  **SElinux配置即时生效**

更改后的内容会在下次系统重启的时候生效，使用命令使更改即时生效。

4  **配置SSH免密码登录**

**步骤1.**  **生成秘钥**

输入命令，生成本机密钥文件。

**步骤2.**  **秘钥分发**

把当前节点的公钥文件id\_dsa.pub内容输出追加到~/.ssh/authorized\_keys文件的末尾。

**步骤3.**  **验证免密码登录是否配置成功**

通过命令验证免密码登录是否配置成功；连接成功后，通过命令退出连接。

5  **安装配置Hadoop**

**步骤1.**  **解压安装文件**

通过命令解压Hadoop安装文件，将文件解压到/usr/cx目录下。

**步骤2.**  **配置Hadoop环境变量**

使用vi编辑器编辑~/.bashrc文件，配置并更新环境变量。

**步骤3.**  **验证环境变量配置**

通过命令验证Hadoop环境变量是否配置成功。

**步骤4.**  **编辑Hadoop配置文件：**

1.   使用vi命令打开hadoop-env.sh配置文件进行编辑。

2.   使用vi命令打开core-site.xml配置文件进行编辑。

3.   使用命令复制mapred-site.xml.template文件并重命名为mapred-site.xml。

4.   使用vi命令打开mapred-site.xml文件进行配置。

5.   执行命令创建Hadoop的数据存储目录namenode和datanode。

6.   使用vi命令打开hdfs-site.xml文件进行配置。

7.   使用vi命令打开slaves文件进行配置。

**步骤5.**  **格式化HDFS**

使用Hadoop命令格式化HDFS文件系统。

6  **Hadoop运行及测试**

**步骤1.**  **启动Hadoop**

1.   通过命令行启动Hadoop。

2.   查看相应的JVM进程确定Hadoop是否配置及启动成功。

**步骤2.**  **Web页面测试**

1.   测试任务调度的Web管理页面是否可以成功访问。

2.   测试HDFS的Web管理页面是否可以成功访问。

7  **安装配置Hive**

**在模板中，我们已经将相应的Hive安装包apache-hive-1.2.1-bin.tar.gz放到/usr/software/目录下，同学们不需要再次下载，可以直接使用。**

我们可以在apache的官网下载相应版本的Hive，官网地址为：http://mirror.bit.edu.cn/apache/hive/（如果有需要，同学们需要在新的标签页打开）。

**步骤1.**  **解压安装文件**

通过下列命令解压Hive安装文件，将文件解压到/usr/cx目录下：

tar -zxvf /usr/software/apache-hive-1.2.1-bin.tar.gz -C /usr/cx/

**步骤2.**  **配置Hive环境变量**

1.   通过下列命令使用vi编辑器编辑~/.bashrc文件：

vi  ~/.bashrc

打开后的文件内容如下所示：

# .bashrc

# User specific aliases and functions

alias rm='rm -i'

alias cp='cp -i'

alias mv='mv -i'

# Source global definitions

if [ -f /etc/bashrc ]; then

        . /etc/bashrc

fi

export JAVA\_HOME=/usr/cx/jdk1.8.0\_60

export PATH=$JAVA\_HOME/bin:$PATH

export CLASSPATH=.:$JAVA\_HOME/jre/lib/rt.jar:$JAVA\_HOME/jre/lib/tools.jar

export HADOOP\_HOME=/usr/cx/hadoop-2.7.1

export PATH=$PATH:$HADOOP\_HOME/bin:$PATH

export PATH=$PATH:$HADOOP\_HOME/sbin:$PATH

(----------------在此处增加内容-------------------)

2.   在~/.bashrc文件中增加以下内容：

export HIVE\_HOME=/usr/cx/apache-hive-1.2.1-bin

export PATH=$HIVE\_HOME/bin:$PATH

编辑完成后保存文件并退出vi编辑器。

3.   执行如下命令，更新环境变量：

source  ~/.bashrc

4.   通过下列命令验证Hive环境变量是否配置成功：

hive -H

如果出现如下提示信息，则说明Hive安装配置成功：

[root@master cx]# hive -H         （注：输入Hive指令，会打印出Hive默认的一些语法操作）

usage: hive

 -d,--define <key=value>          Variable subsitution to apply to hive

                                  commands. e.g. -d A=B or --define A=B

    --database <databasename>     Specify the database to use

 -e <quoted-query-string>         SQL from command line

 -f <filename>                    SQL from files

 -H,--help                        Print help information

    --hiveconf <property=value>   Use value for given property

    --hivevar <key=value>         Variable subsitution to apply to hive

                                  commands. e.g. --hivevar A=B

 -i <filename>                    Initialization SQL file

 -S,--silent                      Silent mode in interactive shell

 -v,--verbose                     Verbose mode (echo executed SQL to the

                                  console)

[root@master cx]#

***如果没有正确输出相关信息，请检查~/.bashrc文件中的Hadoop环境变量是否配置正确，同时请确定是否使用source  ~/.bashrc命令更新环境变量配置。***

**步骤3.**  **编辑Hive配置文件**

1.   使用vi命令打开hive-config.sh配置文件进行编辑：

vi /usr/cx/apache-hive-1.2.1-bin/bin/hive-config.sh

打开后的文件内容如下所示：

# Licensed to the Apache Software Foundation (ASF) under one or more

# contributor license agreements.  See the NOTICE file distributed with

# this work for additional information regarding copyright ownership.

# The ASF licenses this file to You under the Apache License, Version 2.0

# (the "License"); you may not use this file except in compliance with

# the License.  You may obtain a copy of the License at

#

#     http://www.apache.org/licenses/LICENSE-2.0

#

# Unless required by applicable law or agreed to in writing, software

# distributed under the License is distributed on an "AS IS" BASIS,

# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

# See the License for the specific language governing permissions and

# limitations under the License.

#

# processes --config option from command line

#

this="$0"

while [ -h "$this" ]; do

  ls=`ls -ld "$this"`

  link=`expr "$ls" : '.\*-> \(.\*\)$'`

(-------------------省略------------------------)

2.   在文件末尾添加下列内容：

export JAVA\_HOME=/usr/cx/jdk1.8.0\_60

export HADOOP\_HOME=/usr/cx/hadoop-2.7.1

export HIVE\_HOME=/usr/cx/apache-hive-1.2.1-bin

编辑完成后保存文件并退出vi编辑器。

8  **Hive测试**

**注意：由于Derby数据库会将当前的访问目录作为元数据的存储目录，因此我们约定后文中的操作，需要先进入/root目录，然后再进行操作。**

1.   在终端模拟器中输入下列命令，进入/root目录：

cd /root/

2.   通过下面了的命令进入Hive CLI命令行操作模式：

hive

命令运行后的返回结果如下所示：

[root@master ~]# hive

Logging initialized using configuration in jar:file:/usr/cx/apache-hive-1.2.1-bin/lib/hive-common-1.2.1.jar!/hive-log4j.properties

hive>

3.   如果成功进入Hive CLI，我们便可使用HiveQL语句进行数据的管理操作，通过下面的命令创建测试数据表：

create table test(id int);

命令运行后的返回结果如下所示：

hive> create table test(id int);

OK

Time taken: 1.889 seconds

4.   通过下面的命令对表进行查询：

select count(\*) from test;

命令运行后的返回结果如下所示：

hive> select count(\*) from test;

Query ID = root\_20181120091319\_73af20bb-23a5-49ec-9c01-e249c9b894a5

Total jobs = 1

Launching Job 1 out of 1

Number of reduce tasks determined at compile time: 1

In order to change the average load for a reducer (in bytes):

  set hive.exec.reducers.bytes.per.reducer=<number>

In order to limit the maximum number of reducers:

  set hive.exec.reducers.max=<number>

In order to set a constant number of reducers:

  set mapreduce.job.reduces=<number>

Starting Job = job\_1542617837245\_0001, Tracking URL = http://master:8088/proxy/application\_1542617837245\_0001/

Kill Command = /usr/cx/hadoop-2.7.1/bin/hadoop job  -kill job\_1542617837245\_0001

Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1

2018-11-20 09:13:39,571 Stage-1 map = 0%,  reduce = 0%

2018-11-20 09:13:49,562 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.34 sec

2018-11-20 09:14:00,459 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 2.96 sec

MapReduce Total cumulative CPU time: 2 seconds 960 msec

Ended Job = job\_1542617837245\_0001

MapReduce Jobs Launched:

Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 2.96 sec   HDFS Read: 6361 HDFS Write: 2 SUCCESS

Total MapReduce CPU Time Spent: 2 seconds 960 msec

OK

0

Time taken: 41.902 seconds, Fetched: 1 row(s)

由返回结果可知，解析器已经成功将HiveQL语句解析成MapReduce任务运行，因此可以确定Hive安装配置成功。

5.   通过下列命令退出Hive CLI：

exit;

命令运行后将会退出Hive CLI，返回结果如下所示：

hive> exit;

[root@master ~]#

由于我们是在/root目录下进入的HiveCLI操作环境，因此退出HiveCLI后，将会在/root目录下查看到生成的元数据文件metastore\_db。当下次需要进行HiveCLI操作，我们依然需要在/root目录下进入，才能保证访问到元数据文件metastore\_db，从而正确读取已经创建的数据表信息。

由于在安装hive的时候，我们配置了环境变量，所以无论在什么目录下，我们都可以直接在shell命令行中输入hive进入hive的CLI。需要注意的是，在hive运行之前，必须保证你的hadoop已经运行起来了。

9  **数据库操作**

Hive中数据库的概念本质上仅仅是表的一个目录或者命名空间。然而，对于具有很多组和用户的大集群来说，这是非常有用的，因为这样可以避免表命名冲突。通常会使用数据库来将生产表组织成逻辑组。

**步骤1.**  **创建数据库**

1.   在HiveCLI命令行界面中通过下列命令创建数据库financials：

CREATE DATABASE financials;

命令运行后的返回结果如下所示：

hive> CREATE DATABASE financials;

OK

Time taken: 1.131 seconds

hive>

如果我们再次运行命令创建数据库financials，由于数据库financials已经存在，因此会出现错误提示，错误信息如下所示：

hive> CREATE DATABASE financials;

FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask. Database financials already exists

hive>

2.   通过下列命令指定当数据库financials不存在时进行创建，如果已经存在则忽略：

CREATE DATABASE IF NOT EXISTS financials;

命令运行后的返回结果如下所示：

hive> CREATE DATABASE IF NOT EXISTS financials;

OK

Time taken: 0.025 seconds

hive>

3.   通过如下命令可以查看Hive中所包含的数据库信息：

SHOW DATABASES;

命令运行后的返回结果如下所示（可见上文中创建的financials数据库已经成功显示）：

hive> SHOW DATABASES;

OK

default

financials

Time taken: 0.344 seconds, Fetched: 2 row(s)

hive>

4.   可以使用正则表达式匹配来筛选出需要的数据库名。通过下面的命令列举出所有以字母fi开头，以其他字符结尾（即.\*部分含义）的数据库名：

SHOW DATABASES LIKE 'fi.\*';

命令运行后的返回结果如下所示：

hive> SHOW DATABASES LIKE 'fi.\*';

OK

financials

Time taken: 0.055 seconds, Fetched: 1 row(s)

hive>

默认情况下，当我们创建数据库financials时，Hive将会在HDFS的/user/hive/warehouse/目录下创建financials.db子目录，我们可以访问HDFS进行查看。

5.   打开一个新的终端模拟器，通过下面的命令查看/user/hive/warehouse/目录下的内容：

hadoop fs -ls /user/hive/warehouse/

命令运行后的返回结果如下所示（由返回结果可以看出，financials.db子目录已经被成功创建）：

[root@master ~]# hadoop fs -ls /user/hive/warehouse/

18/11/20 15:56:39 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

Found 1 items

drwxr-xr-x   - root supergroup          0 2018-11-20 15:41 /user/hive/warehouse/financials.db

[root@master ~]#

**步骤2.**  **为数据库增加描述信息**

1.   在HiveCLI命令行界面中通过下面的命令可以在创建数据库的同时为数据库增加描述信息：

CREATE DATABASE mydatabase02 COMMENT 'This is my second database.';

命令运行后的返回结果如下所示：

hive> CREATE DATABASE mydatabase02 COMMENT 'This is my second database.';

OK

Time taken: 0.066 seconds

hive>

2.   通过下面的命令可以查看数据库的相关信息：

DESCRIBE DATABASE mydatabase02;

命令运行后的返回结果如下所示：

hive> DESCRIBE DATABASE mydatabase02;

OK

mydatabase02    This is my second database.    hdfs://master:9000/user/hive/warehouse/mydatabase02.db   root USER

Time taken: 0.074 seconds, Fetched: 1 row(s)

hive>

DESCRIEB DATABASE语句可以显示出这个数据库描述信息以及所在的文件目录位置路径等信息。

3.   在HiveCLI命令行界面中通过下面的命令可以在创建数据库的同时为数据库增加相关的键-值对属性信息：

CREATE DATABASE mydatabase03 WITH DBPROPERTIES('creator'='root','date'='2015-8-11');

命令运行后的返回结果如下所示：

hive> CREATE DATABASE mydatabase03 WITH DBPROPERTIES('creator'='root','date'='2015-8-11');

OK

Time taken: 0.121 seconds

hive>

4.   通过下面的命令可以查看数据库的相关信息：

DESCRIBE DATABASE mydatabase03;

命令运行后的返回结果如下所示：

hive> DESCRIBE DATABASE mydatabase03;

OK

mydatabase03         hdfs://master:9000/user/hive/warehouse/mydatabase03.db   root USER

Time taken: 0.071 seconds, Fetched: 1 row(s)

hive>

5.   通过EXTENDED关键词可以更加清晰的查看数据库的键-值对信息：

DESCRIBE DATABASE EXTENDED mydatabase03;

命令运行后的返回结果如下所示：

hive> DESCRIBE DATABASE EXTENDED mydatabase03;

OK

mydatabase03         hdfs://master:9000/user/hive/warehouse/mydatabase03.db   root USER   {date=2015-8-11, creator=root}

Time taken: 0.051 seconds, Fetched: 1 row(s)

hive>

**步骤3.**  **显示当前数据库**

1.   可以通过下面的命令设置一个属性值来在提示符里面显示当前所在的数据库信息：

set hive.cli.print.current.db=true;

命令运行后的返回结果如下所示：

hive> set hive.cli.print.current.db=true;

hive (default)>

2.   通过下面的命令切换到数据库mydatabase03，提示符中的数据库信息也会随之改变：

use mydatabase03;

命令运行后的返回结果如下所示：

hive (default)> use mydatabase03;

OK

Time taken: 0.036 seconds

hive (mydatabase03)>

**步骤4.**  **删除数据库**

默认情况下，Hive是不允许用户删除一个包含有表的数据库的。因此当数据库中存在数据表，可以先删除数据库中的表，然后再删除数据库，或者在删除命令的后面加上关键字CASCADE，这样可以使Hive自行先删除数据库中的表。

1.   通过下面的命令可以删除一个空数据库：

DROP DATABASE IF EXISTS mydatabase02;

命令运行后的返回结果如下所示：

hive (mydatabase03)> DROP DATABASE IF EXISTS mydatabase02;

OK

Time taken: 0.052 seconds

hive (mydatabase03)>

2.   通过下面的命令在数据库mydatabase03中创建表test：

CREATE TABLE mydatabase03.test(name STRING);

命令运行后的返回结果如下所示：

hive (mydatabase03)> CREATE TABLE mydatabase03.test(name STRING);

OK

Time taken: 0.404 seconds

hive (mydatabase03)>

3.   通过下面的命令可以查看mydatabase03数据库中的表信息：

show tables;

命令运行后的返回结果如下所示：

hive (mydatabase03)> show tables;

OK

test

Time taken: 0.069 seconds, Fetched: 1 row(s)

hive (mydatabase03)>

4.   通过下面的命令删除数据库mydatabase03，将会出现错误提示：

DROP DATABASE IF EXISTS mydatabase03;

命令运行后的返回结果如下所示：

hive (mydatabase03)> DROP DATABASE IF EXISTS mydatabase03;

FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask. InvalidOperationException(message:Database mydatabase03 is not empty. One or more tables exist.)

hive (mydatabase03)>

5.   通过下面的命令可以强制删除非空数据库mydatabase03：

DROP DATABASE IF EXISTS mydatabase03 CASCADE;

命令运行后的返回结果如下所示：

hive (mydatabase03)> DROP DATABASE IF EXISTS mydatabase03 CASCADE;

OK

Time taken: 1.03 seconds

hive (mydatabase03)>

6.   通过下面的命令查看数据库：

show databases;

命令运行后的返回结果如下所示，可见数据库mydatabase03已经成功被删除：

hive (mydatabase03)> show databases;

OK

default

financials

Time taken: 0.027 seconds, Fetched: 2 row(s)

hive (mydatabase03)>

7.   通过下面的命令可以切换回default数据库：

use default;

命令运行后的返回结果如下所示：

hive (mydatabase03)> use default;

OK

Time taken: 0.039 seconds

hive (default)>

**步骤5.**  **修改数据库**

在Hive中数据库的元数据信息都是不可更改的，包括数据库名和数据库所在的目录位置。可以使用ALTER DATABASE <database>命令更改某个数据库的DBPROPERTIES键-值对属性信息，但是没有办法可以删除或者“重置”数据库属性。

1.   在HiveCLI命令行界面通过下面的命令新建一个带有键-值对属性信息的数据库mydatabase04：

CREATE DATABASE mydatabase04 WITH DBPROPERTIES('data'='2012-01-01');

命令运行后的返回结果如下所示：

hive (default)> CREATE DATABASE mydatabase04 WITH DBPROPERTIES('data'='2012-01-01');

OK

Time taken: 0.055 seconds

hive (default)>

2.   通过下面的命令查看数据库的键-值对属性信息：

DESCRIBE DATABASE EXTENDED mydatabase04;

命令运行后的返回结果如下所示：

hive (default)> DESCRIBE DATABASE EXTENDED mydatabase04;

OK

mydatabase04         hdfs://master:9000/user/hive/warehouse/mydatabase04.db   root USER   {data=2012-01-01}

Time taken: 0.053 seconds, Fetched: 1 row(s)

hive (default)>

3.   通过下面的命令可以修改键-值对属性信息：

ALTER DATABASE mydatabase04 SET DBPROPERTIES('data'='2015-08-12');

命令运行后的返回结果如下所示：

hive (default)> ALTER DATABASE mydatabase04 SET DBPROPERTIES('data'='2015-08-12');

OK

Time taken: 0.068 seconds

hive (default)>

4.   通过下面的命令查看数据库的键-值对属性信息：

DESCRIBE DATABASE EXTENDED mydatabase04;

命令运行后的返回结果如下所示，可见属性信息已经成功被修改：

hive (default)> DESCRIBE DATABASE EXTENDED mydatabase04;

OK

mydatabase04         hdfs://master:9000/user/hive/warehouse/mydatabase04.db   root USER   {data=2015-08-12}

Time taken: 0.039 seconds, Fetched: 1 row(s)

hive (default)>