hadoop + ffmpeg 分布式转码系统实践
hadoop 安装
ffmpeg 安装
mkvtoolnix 安装

一、分割视频：
```
mkvmerge --split size:32m ./heihu01.mp4 -o ./heihu01.%05d.mp4
```

二、hdfs中创建存放分割后视频的目录
```
hadoop fs -mkdir movies
```

三、上传分割后的视频
```
for i in `ls heihu01.*.mp4`; do hadoop fs -put $i movies/; done
```

四、创建mapper输入数据文件路径
```
hadoop fs -mkdir movies_input
```

五、生成mapper数据文件，并上传
```
cat > mapper_input.sh<<EOF
pwd=\`pwd\`
tmp_file='movies_tmp.txt'
num=2   #TaskTracker数量
true > \${tmp_file}
hadoop fs -rm movies_input/movies_*
for i in \`ls *.[0-9][0-9][0-9][0-9][0-9].*\`;do  echo movies/\$i >> \${tmp_file};done
rows="\$((\$(wc -l \${tmp_file}|cut -d' ' -f1)/\$num))"
split -l \$rows \${tmp_file} movies_
hadoop fs -put movies_[a-z0-9][a-z0-9] movies_input
EOF


chmod +x mapper_input.sh
./mapper_input.sh
```

六、创建转换后视频的上传路径
```
hadoop fs -mkdir movies_put
```

七、检查Hadoop Streaming的执行身份与工作目录
1、编写脚本
```
cat > test_mapper.sh << EOF
#!/bin/bash
set -x
id="`whoami`"
mkdir -p /tmp/\$id
host=\`hostname\`
pwd=\`pwd\`
uid=\`whoami\`
put_dir='movies_put'

while read line; do
  input=\$line
  filename=\`basename \$input\`
  echo "\$uid@\$host:\$pwd> hadoop fs -get \$input /tmp/\$id/\$filename"
  echo "\$uid@\$host:\$pwd> ffmpeg -y -i /tmp/\$id/\$filename -s qcif -r 20 -b 200k -vcodec
```

```
mpeg4 -ab 64k -ac 2 -ar 22050 -acodec libfaac output-\$filename.3gp"
  echo "\$uid@\$host:\$pwd> hadoop fs -put output-\$filename \${put_dir}"
done
rm -rf /tmp/\$id
EOF

chmod a+x test_mapper.sh
```

2、本地执行测试

```
cat movies_aa |./test_mapper.sh
```

3、hadoop streaming执行测试

```
hadoop jar /usr/local/hadoop/contrib/streaming/hadoop-streaming-1.0.2.jar -input
movies_input -output movies_output -mapper test_mapper.sh -file test_mapper.sh
```

4、查看hadoop streaming执行结果

```
hadoop fs -cat /user/$(whoami)/movies_output/part-00000 | head
```

5、删除测试输出

```
hadoop fs -rmr movies_output    #删除测试hadoop streaming的输出
```

八、使用hadoop streaming执行转码

1、编写脚本

```
cat > mapper.sh << EOF
#!/bin/bash
id="hduser"
mkdir -p /tmp/\$id
host=\`hostname\`
pwd=\`pwd\`
uid=\`whoami\`
put_dir='movies_put'
cd "/tmp/\$id"
true > a

while read line; do
  input=\$line
  filename=\`basename \$input\`
  echo "\$uid@\$host> hadoop fs -get \$input /tmp/\$id/\$filename"
  /usr/local/hadoop/bin/hadoop fs -get \$input /tmp/\$id/\$filename 2>&1
  echo "\$uid@\$host> ffmpeg -y -i /tmp/\$id/\$filename -s qcif -r 20 -b 200k -vcodec mpeg4 -
ab 64k -ac 2 -ar 22050 -acodec libfaac output-\$filename.3gp"
  ffmpeg -y -i /tmp/\$id/\$filename -s 320*240 -r 20 -b 200k -vcodec mpeg4 -ab 64k -ac 2 -ar
22050 -qscale 5 -acodec libfaac  output-\$filename.3gp < a 2>&1
  /usr/local/hadoop/bin/hadoop fs -put output-\$filename.3gp \${put_dir} 2>&1
  echo "\$uid@\$host> hadoop fs -chown \$id \${put_dir}/output-\$filename.3gp"
  /usr/local/hadoop/bin/hadoop fs -chown \$id \${put_dir}/output-\$filename.3gp 2>&1

done
rm -f a
rm -rf /tmp/\$id
EOF

chmod a+x mapper.sh
```

2、本地执行测试

```
cat movies_aa | ./mapper.sh
hadoop fs -rm movies_put/*      #删除本地执行的遗留文件
```

3、使用hadoop执行脚本

```
hadoop jar /usr/local/hadoop/contrib/streaming/hadoop-streaming-1.0.2.jar -input
movies_input -output movies_output -mapper mapper.sh  -file mapper.sh
```

4、验证结果

```
hadoop fs -cat movies_output/part-00000 |head
hadoop fs -ls movies_put
Found 6 items
-rw-r--r--   3 hduser supergroup   19584280 2012-05-28 13:53
/user/hduser/movies_put/output-heihu01.00001.mp4.3gp
-rw-r--r--   3 hduser supergroup   14872878 2012-05-28 13:54
/user/hduser/movies_put/output-heihu01.00002.mp4.3gp
-rw-r--r--   3 hduser supergroup   12052800 2012-05-28 13:55
/user/hduser/movies_put/output-heihu01.00003.mp4.3gp
-rw-r--r--   3 hduser supergroup   11174014 2012-05-28 13:53
/user/hduser/movies_put/output-heihu01.00004.mp4.3gp
-rw-r--r--   3 hduser supergroup   15713836 2012-05-28 13:55
/user/hduser/movies_put/output-heihu01.00005.mp4.3gp
-rw-r--r--   3 hduser supergroup   13084511 2012-05-28 13:56
/user/hduser/movies_put/output-heihu01.00006.mp4.3gp
```

5、reduce合并视频

```
cat >reduce.sh <<EOF
#!/bin/bash
tmp_file="movies_tmp.txt"
id="hduser"
pwd=\`pwd\`
dir="/tmp/\${id}_merger"
mkdir \$dir
cd $dir
true > \${tmp_file}
hadoop fs -ls movies_put|awk '{print \$8}'|sed '/^$/d' >> \${tmp_file}

unset m
for i in \`cat \${tmp_file}\`
do
    hadoop fs -get \$i \$dir
    filename=\`basename \$i\`
    if [ ! -z \$m ];then
        filename="+\$filename"
    fi
    echo \$filename >> \$dir/files.txt
    m=\$((m+1))
done

mkvmerge -o \$dir/output.3gp \`cat \$dir/files.txt\`
hadoop fs -put \$dir/output.3gp movies_put/
rm -rf \$dir
EOF
```

chmod +x reduce.sh
6、本地执行测试
./reduce.sh

7、使用hadoop执行脚本
hadoop jar /usr/local/hadoop/contrib/streaming/hadoop-streaming-1.0.2.jar -input movies_input -output movies_output -mapper mapper.sh -reducer reduce.sh -file reduce.sh -file mapper.sh

8、验证结果
hadoop fs -ls movies_put
Found 7 items
-rw-r--r-- 3 hduser supergroup 19584280 2012-05-29 14:15 /user/hduser/movies_put/output-heihu01.00001.mp4.3gp
-rw-r--r-- 3 hduser supergroup 14872878 2012-05-29 14:16 /user/hduser/movies_put/output-heihu01.00002.mp4.3gp
-rw-r--r-- 3 hduser supergroup 12052800 2012-05-29 14:17 /user/hduser/movies_put/output-heihu01.00003.mp4.3gp
-rw-r--r-- 3 hduser supergroup 11174014 2012-05-29 14:15 /user/hduser/movies_put/output-heihu01.00004.mp4.3gp
-rw-r--r-- 3 hduser supergroup 15713836 2012-05-29 14:16 /user/hduser/movies_put/output-heihu01.00005.mp4.3gp
-rw-r--r-- 3 hduser supergroup 13084511 2012-05-29 14:17 /user/hduser/movies_put/output-heihu01.00006.mp4.3gp
-rw-r--r-- 3 hduser supergroup 86175913 2012-05-29 14:17 /user/hduser/movies_put/output.3gp

hadoop fs -cat movies_output/part-00000

附：
hadoop streaming 调试
hadoop 的output 中只记录正确输出，因此调试错误需要将命令的输出重定向到正确输出
即在命令后加"2>&1"，如：
mkvmerge -o $dir/output.3gp `cat $dir/files.txt` 2>&1