

实验三 HDFS 核心设计实验报告

江昱峰 21009200038

2023 年 11 月 16 日

1 背景介绍

在我们平常电脑应用上，我们经常会有删除文件的操作。也有时候我们会不小心将文件删除，但这些删除并非永久删除，而是保存在回收站，我们可通过恢复文件将我们删除的文件复原。

HDFS 回收机制也是如此，当我们删除文件时，会将删除文件放置在回收站，以备我们误删时恢复。HDFS 回收机制默认是关闭的，因此本节实训学习配置并开启垃圾回收机制。

2 实验目的

实践并掌握 HDFS 核心设计，具体包括以下两部分内容：

- 心跳机制配置；
- 垃圾回收机制配置。

3 实验知识

心跳机制简介：

所谓“心跳”是一种形象化描述，指的是持续的按照一定频率在运行，类似于心脏在永无休止的跳动。

心跳机制就是为了证明数据节点还活着，如果一段时间内没有向 NameNode 发送心跳包信息，就会被 Dead 状态。并且 DataNode 从心跳包回复中获取命令信息，然后进行下一步操作，所以从这里可以看出，心跳机制在整个 HDFS 系统中都有很重要的作用。

4 实验要求

完成 HDFS 核心设计，具体报考以下两部分任务：

- 配置心跳机制，模拟 DataNode 节点挂掉，是否在配置时间出现 Dead 状态。
- 配置并开启垃圾回收机制，模拟删除文件并恢复。

5 实验环境

本次实验实验环境为青椒课堂平台的 Linux（Centos 7.5）操作系统。

6 实验步骤与结果分析

6.1 心跳机制配置

任务：配置心跳机制文件

6.1.1 文件配置

1. 关闭 Hadoop 集群。
2. 进入文件 (hdfs-default.xml) 所在目录 /root/software/hadoop-2.7.7/etc/hadoop/。

```
→ ~ cd /root/software/hadoop-2.7.7/etc/hadoop  
→ hadoop ls
```

3. 添加心跳周期为 3s。

```
File Edit View Terminal Tabs Help
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed 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. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>dfs.namenode.name.dir</name>
<value>/root/hadoopData/name</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/root/hadoopData/data</value>
</property>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
  <name>dfs.heartbeat.interval</name>
  <value>3</value> //单位秒
</property>
</configuration>
```

4. 添加心跳检测时长为 15s。

```

<configuration>
<property>
<name>dfs.namenode.name.dir</name>
<value>/root/hadoopData/name</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>/root/hadoopData/data</value>
</property>
<property>
<name>dfs.replication</name>
<value>1</value>
</property>
<property>
    <name>dfs.heartbeat.interval</name>
    <value>3</value> //单位 秒
</property>
<property>
    <name>dfs.namenode.heartbeat.recheck-interval</name>

    <value>15000</value> //单位 毫秒
</property>
</configuration>

```

5. 保存配置，重启集群。

```

→ hadoop bash hadoop-env.sh
→ hadoop █

```

6.1.2 查看 DataNode 状态

1. 浏览器访问 HDFS Web UI。

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
qngjao50010 (127.0.0.1:50010)	2	In Service	68.78 GB	44 KB	6.3 GB	59.44 GB	1	44 KB (0%)	0	2.7.7

6.1.3 模拟 DataNode 宕机

1. 通过文件配置，我们计算出宕机时间为 1 分钟。公式如下：

$$Time = 2dfs.namenode.heartbeat.recheck-interval + 10dfs.heartbeat.interval$$

2. 通过 jps 命令查看进程。

```
→ hadoop jps
1046 ResourceManager
886 SecondaryNameNode
1161 NodeManager
2905 Jps
588 NameNode
718 DataNode
→ hadoop █
```

3. 使用 kill 命令删除 DataNode 节点。

```
→ hadoop kill 718
→ hadoop █
```

4. 查看 Last contact 值。当我们 kill 掉 DataNode 进程一分钟左右时间，当我们发现 Last contact 值报红，显示为 Dead 装填状态，证明我们配置成功。

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
qngjao50010 (127.0.0.1:50010)	78	In Service	68.78 GB	24 KB	6.3 GB	59.44 GB	0	24 KB (0%)	0	2.7.7

6.2 垃圾回收机制配置

6.2.1 配置垃圾回收机制文件

1. 关闭集群。
2. 进入 Hadoop 安装目录 (/root/software/hadoop-2.7.7/etc/hadoop) 设置 fs.trash.interval 值为一天。

```
Terminal 终端
File Edit View Terminal Tabs Help
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed 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. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://localhost:9000</value>
</property>
<property>
<name>hadoop.tmp.dir</name>
<value>/root/hadoopData/temp</value>
</property>
<property>
<name>fs.trash.interval</name>
<value>1440</value>
</property>
</configuration>
```

3. 设置 fs.trash.checkpoint.interval 值与 fs.trash.interval 相同。

```
Terminal 终端
File Edit View Terminal Tabs Help
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed 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. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://localhost:9000</value>
</property>
<property>
<name>hadoop.tmp.dir</name>
<value>/root/hadoopData/temp</value>
</property>
<property>
<name>fs.trash.interval</name>
<value>1440</value>
</property>
<property>
<name>fs.trash.interval</name>
<value>1440</value>
</property>
</configuration>
```

4. 重启集群。

```
→ hadoop bash hadoop-env.sh
→ hadoop
```

6.2.2 模拟文件删除并恢复

1. 在/root 下创建并编辑文件 test.txt。编辑内容如下：
core-site.xml
fs.trash.interval
fs.trash.checkpoint.interval


```
File Edit View Terminal Tabs Help
core-site.xml
fs.trash.interval
fs.trash.checkpoint.interval
```

2. 将文件上传到 HDFS 根目录。

```
→ root hadoop fs -put test.txt /
→ root
```

3. 删除 HDFS 文件/test.txt。

```
→ root hadoop fs -rm /test.txt
23/11/12 12:24:17 INFO fs.TrashPolicyDefault: Namenode trash configuration: Deletion interval = 1
440 minutes, Emptier interval = 0 minutes.
23/11/12 12:24:17 INFO fs.TrashPolicyDefault: Moved: 'hdfs://localhost:9000/test.txt' to trash at
: hdfs://localhost:9000/user/root/.Trash/Current/test.txt
Moved: 'hdfs://localhost:9000/test.txt' to trash at: hdfs://localhost:9000/user/root/.Trash/Curre
nt
→ root
```

4. 通过执行命令返回信息查看被删除文件所在路径并进入路径查看文件是否存在。默认的地址为“/user/搭建集群的用户名/.Trash/”。

```
→ root hdfs dfs -ls /user/root
Found 1 items
drwx----- - root supergroup          0 2023-11-12 12:24 /user/root/.Trash
→ root
```

```
→ root hadoop fs -ls /user/root/.Trash/Current
Found 1 items
-rw-r--r--  1 root supergroup          62 2023-11-12 12:23 /user/root/.Trash/Current/test.txt
→ root
```

5. 将文件移动到 HDFS 根目录。

```
→ root hadoop fs -mv /user/root/.Trash/Current/test.txt /
→ root
```

6. 查看文件内容。

```
→ root hadoop fs -cat /test.txt  
core-site.xml  
fs.trash.interval  
fs.trash.checkpoint.interval  
→ root
```

7 结果分析

- 超过宕机时间后，心跳状态变为 Dead。
- 类似于 Windows 操作系统，Hadoop 集群中也有指定的回收站用于回收垃圾。

8 困难解决

3.2.3(模拟文件删除并恢复)部分中,执行命令返回的被删除文件所在路径信息与实验平台文档内容提供的路径不完全相同,前者多了一个/Current的子目录。我根据实际情况,判定为文档错误。

9 心得体会

做完本次实验,除了掌握了实验目的部分中所有内容的收获之外,我还有以下几点心得体会:

- 对比分析了心跳机制与 TCP 协议中三次握手机制等的异同点。
- 对比分析了 HDFS 中的垃圾回收机制与 Windows、Linux 等操作系统中回收站的异同点。