

HBase运维管理

www.huawei.com





目标

- 学完本课程后，您将能够：
 - 掌握**HBase**常用维护命令/参数
 - 掌握**HBase**性能调优参数
 - 掌握**HBase**常见问题及定位



目录

1. 服务检查与日志
2. 维护命令与参数
3. **Hbase**备份与恢复
4. 性能调优命令与参数
5. 常见问题与定位

HBase目录介绍

HBase组件目录：

`{$BIGDATA_HOME}/FusionInsight_V100R002C60/FusionInsight-HBase-1.0.2/hbase`

bin	hbase 运行脚本目录
CHANGES.txt	该版本变更信息
conf	原 hbase 配置文件目录(FusionInsight 实际使用的配置目录统一在安装目录 etc 底下)
hbase-webapps	hbase web 文件目录
lib	hbase 依赖包目录
LICENSE.txt	licence
NOTICE.txt	声明文件
version.properties	版本说明

服务健康检查

手动执行HBase的健康检查：

1. 单击“服务管理”，在服务列表中单击服务指定名称。
2. 选择“更多操作 > 启动服务健康检查”启动指定服务健康检查。



健康检查指标项

- **HBase数据写入延时**

- 指标项名称：**HBase数据写入延时**
- 指标项含义：检查**HBase**数据操作是否正常。如果数据写入操作延时大于**300**毫秒，则认为不健康。
- 恢复指导：如果该指标项异常，请检查**RegionServer**的状态是否正常并处理，然后建议检查网络是否正常。

- **HBase数据删除延时**

- 指标项名称：**HBase数据删除延时**
- 指标项含义：检查**HBase**数据操作是否正常。如果数据删除操作延时大于**300**毫秒，则认为不健康。
- 恢复指导：如果该指标项异常，请检查**RegionServer**的状态是否正常并处理，然后建议检查网络是否正常。

健康检查指标项

- **HBase数据读取延时**
 - 指标项名称：**HBase数据读取延时**
 - 指标项含义：检查**HBase**数据操作是否正常。如果数据读取操作延时大于**300**毫秒，则认为不健康。
 - 恢复指导：如果该指标项异常，请检查**RegionServer**的状态是否正常并处理，然后建议检查网络是否正常。
- **服务健康状态**
 - 指标项名称：服务健康状态
 - 指标项含义：检查**HBase**服务状态是否正常。如果状态不正常，则认为不健康。
 - 恢复指导：如果该指标项异常，请检查**HMaster**和**RegionServer**的状态是否正常并先处理，然后检查**ZooKeeper**服务的状态是否为**bad**并处理。使用**HBase**客户端，确认是否可以正确读取**HBase**表中的数据，排查读数据失败的原因。最后参见告警进行处理。

健康检查指标项

- 检查告警

- 指标项名称： 告警信息
- 指标项含义： 检查服务是否存在未清除的告警。如果存在，则认为不健康。
- 恢复指导： 如果该指标项异常，建议参见告警进行处理。

服务健康检查日志

健康检查完成后生成相关日志文件

可在主**HMaster**上查看如下日志文件：

- `/var/log/Bigdata/hbase/hm/checkServiceDetail.log`
- `/var/log/Bigdata/hbase/hm/hbase.log`

HBase 日志介绍

日志存储路径:

- **HBase**相关日志的默认存储路径为`/var/log/Bigdata/hbase/`角色名
- **HMaster**:
 - `/var/log/Bigdata/hbase/hm` (运行日志)
 - `/var/log/Bigdata/audit/hbase/hm` (审计日志)
- **RegionServer**:
 - `/var/log/Bigdata/hbase/rs` (运行日志)
 - `/var/log/Bigdata/audit/hbase/rs` (审计日志)
- **ThriftServer**:
 - `/var/log/Bigdata/hbase/ts` (运行日志)
 - `/var/log/Bigdata/audit/hbase/ts` (审计日志)

HBase 日志归档

日志归档规则：

- **HBase**的日志启动了自动压缩归档功能，缺省情况下，当日志大小超过**30MB**的时候（此日志文件大小可进行配置，详情请参见“配置日志级别与文件大小”），会自动压缩，压缩后的日志文件名规则为：“<原有日志名>-<yyyy-mm-dd_hh-mm-ss>.[编号].log.zip”。最多保留最近的**20**个压缩文件，压缩文件保留个数可以在**FusionInsight Manager**界面中配置。

HBase日志说明

日志类型	日志文件名	描述
运行日志	hbase-<SSH_USER>-<process_name>-<hostname>.log	记录 HBase 启动信息的日志，主要包括启动时间，启动参数信息。
	hbase-<SSH_USER>-<process_name>-<hostname>.out	HBase 运行环境信息日志。
	<process_name>-<SSH_USER>-gc.log	HBase 服务垃圾回收日志。
	checkServiceDetail.log	HBase 服务启动是否成功的检查日志。
	hbase.log	HBase 系统日志，记录 HBase 系统运行时候所产生的大部分日志。
审计日志	hbase-audit-<process_name>.log	HBase 安全审计日志。

HBase 日志级别

- **HBase**中提供下表所示的日志级别。日志级别优先级从高到低分别是 **FATAL**、**ERROR**、**WARN**、**INFO**、**DEBUG**。程序会打印高于或等于所设置级别的日志，设置的日志等级越高，打印出来的日志就越少。

级别	描述
FATAL	FATAL 表示当前事件处理出现严重错误信息，可能导致系统崩溃。
ERROR	ERROR 表示当前事件处理出现错误信息，系统运行出错。
WARN	WARN 表示当前事件处理存在异常信息，但认为是正常范围，不会导致系统出错。
INFO	INFO 表示记录系统及各事件正常运行状态信息
DEBUG	DEBUG 表示记录系统及系统的调试信息。



目录

1. 服务检查与日志
2. 维护命令与参数
3. **Hbase**备份与恢复
4. 性能调优命令与参数
5. 常见问题与定位

HBase Shell维护命令

如何进入HBase shell

安装客户端

进入客户端执行: **source ./bigdata_env**
配置环境变量

执行**kinit** <用户名>进行认证

执行**hbase shell**进入命令窗口

HBase Shell维护命令

命令帮助类

help

- 作用：用于查询**HBase**命令帮助

集群状态查询类

version

- 作用：查询**HBase**集群版本信息

whoami

- 作用：查询当前登陆的用户

status

- 作用：用于查询**HBase**集群状态

样例1：查询基本信息，**status 'simple'**

样例2：查询详细信息，**status 'detailed'**

HBase Shell维护命令

权限类

grant

- 作用：对指定用户赋予表/列族/行的读、写、管理等权限

样例1：对用户**huawei**赋予所有表的所有权限

```
hbase(main):003:0> grant 'huawei','RWXCA'  
0 row(s) in 0.1470 seconds
```

样例2：对用户**huawei**赋予**test**表的读写

```
hbase(main):003:0> grant 'huawei','RW','test'  
0 row(s) in 0.2440 seconds
```

revoke

- 作用：收回指定用户已经被赋予的权限

HBase Shell维护命令

权限类

user_permission

- 作用：查询某表的用户权限

样例：查询test表的权限

```
hbase(main):011:0> user_permission 'test'
User                               Table,Family,Qualifier:Permission
hbase                               test,,: [Permission: actions=READ,WRITE,EXEC,CREATE,ADMIN]
1 row(s) in 0.1370 seconds
```

HBase Shell维护命令

Region操作类

assign

- 作用：手动让**region**上线

unassign

- 作用：手动让**region**下线

compact

- 作用：手动对一个表或一个**Region**进行**compact**操作

HBase Shell维护命令

Region操作类

split

- 作用：手动对一个表或一个**Region**进行分裂操作

balancer

- 作用：手动触发**HBase**负载均衡

major_compact

- 作用：手动对一个表或一个**Region**进行**Compact**操作

检测集群健康状态

Hbase hbck

- 作用：查看集群健康状态
- **hbase hbck**并不是**hbase shell**命令，在登录认证之后，直接执行**hbase hbck**即可

Metadata Repair options: (expert features, use with caution!)

```
-fix Try to fix region assignments. This is for backwards compatibility
-fixAssignments Try to fix region assignments. Replaces the old -fix
-fixMeta Try to fix meta problems. This assumes HDFS region info is good.
-fixHdfsHoles Try to fix region holes in hdfs.
-fixHdfsOrphans Try to fix region dirs with no .regioninfo file in hdfs
-fixTableOrphans Try to fix table dirs with no .tableinfo file in hdfs (online mode only)
-fixHdfsOverlaps Try to fix region overlaps in hdfs.
-fixVersionFile Try to fix missing hbase.version file in hdfs.
-maxMerge <n> When fixing region overlaps, allow at most <n> regions to merge. (n=5 by default)
-sidelineBigOverlaps When fixing region overlaps, allow to sideline big overlaps
-maxOverlapsToSideline <n> When fixing region overlaps, allow at most <n> regions to sideline per group. (n=2 by default)
-fixSplitParents Try to force offline split parents to be online.
-ignorePreCheckPermission ignore filesystem permission pre-check
```

Datafile Repair options: (expert features, use with caution!)

```
-checkCorruptHFiles Check all Hfiles by opening them to make sure they are valid
-sidelineCorruptHfiles Quarantine corrupted HFiles. implies -checkCorruptHfiles
```

Metadata Repair shortcuts

```
-repair Shortcut for -fixAssignments -fixMeta -fixHdfsHoles -fixHdfsOrphans -fixHdfsOverlaps -fixVersionFile -sidelineBigOverlaps
-repairHoles Shortcut for -fixAssignments -fixMeta -fixHdfsHoles
```

检测集群健康状态

```
[root@160-172-0-11 client]# hbase hbck
log4j:WARN No appenders could be found for logger (org.apache.hadoop.conf.Configuration).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/client/HBase/hbase-0.94.0-security/lib/phoenix-2.2.3-client.jar]
SLF4J: Found binding in [jar:file:/opt/client/HBase/hbase-0.94.0-security/lib/slf4j-log4j12-1.6.1.jar]
SLF4J: Found binding in [jar:file:/opt/client/Hadoop/hadoop-2.0.1/share/hadoop/common/lib/slf4j-log4j12-1.6.1.jar]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
Version: 0.94.0

Number of Tables: 3
Number of live region servers: 3
Number of dead region servers: 0
Master: 160-172-0-12,26001,1419583373004
Number of backup masters: 1
Number of empty REGIONINFO QUALIFIER rows in .META.: 0
Summary:
  -ROOT- is okay.
    Number of regions: 1
    Deployed on: 160-172-0-12,26003,1419583377622
  .META. is okay.
    Number of regions: 1
    Deployed on: 160-172-0-11,26003,1419583378235
  HBASE_SYSTEM is okay.
    Number of regions: 1
    Deployed on: 160-172-0-11,26003,1419583378235
  _acl_ is okay.
    Number of regions: 1
    Deployed on: 160-172-0-12,26003,1419583377622
  test is okay.
    Number of regions: 1
    Deployed on: 160-172-0-13,26003,1419583379027
0 inconsistencies detected.
Status: OK
```



目录

1. 服务检查与日志
2. 维护命令与参数
3. **HBase备份与恢复**
4. 性能调优命令与参数
5. 常见问题与定位

HBase备份

- 为了确保**HBase**日常数据安全，或者系统管理员需要对**HBase**进行重大操作（如升级或迁移等），需要对**HBase**数据进行备份，从而保证系统在出现异常或未达到预期结果时可以及时进行数据恢复，将对业务的影响降到最低。
- 备份内容为**HBase**表级别的用户数据。

HBase备份

备份前的准备和约束


- 需要准备一个用于备份数据的备集群。
- 主备集群必须已配置跨集群互信。
- 主备集群必须已配置跨集群拷贝。
- 主备集群上的时间必须一致，而且主备集群上的**NTP**服务必须使用同一个时间源。
- 根据业务需要，规划备份任务的类型、周期、备份对象、备份目录和备份任务需要使用的**YARN**队列等策略规格。
- 检查备集群**HDFS**是否有充足的空间，备份文件保存的目录建议使用用户自定义的目录。
- 使用**HDFS**客户端，以“**hdfs**”用户执行**hdfs lsSnapshottableDir**检查当前集群中已创建**HDFS**快照的目录清单，确保待备份的数据文件所在**HDFS**路径的父目录或子目录不存在**HDFS**快照，否则无法创建备份任务。

HBase备份

- 创建备份任务

1. 在FusionInsight Manager，选择“系统设置 > 备份管理”。

2. 单击“创建备份任务”。



The screenshot shows the 'FusionInsight Manager' web interface. The top navigation bar includes icons for '系统概览' (System Overview), '服务管理' (Service Management), '主机管理' (Host Management), '告警管理' (Alert Management), '审计管理' (Audit Management), '租户管理' (Tenant Management), and '系统设置' (System Settings). The '系统设置' menu is expanded, showing '备份管理' (Backup Management) and '创建备份任务' (Create Backup Task). The '创建备份任务' page contains the following fields and options:

- * 任务名称:** A text input field containing 'test' with a green checkmark icon to its right.
- 备份类型:** Two radio button options: '周期备份' (Periodic Backup) and '手动备份' (Manual Backup). '周期备份' is selected.
- * 开始时间:** A text input field containing '2016-04-27 16:17:43' with a green checkmark icon to its right.
- * 周期:** A text input field containing '1' followed by two radio button options: '按小时' (By Hour) and '按天' (By Day). '按天' is selected, with a green checkmark icon to its right.
- * 备份策略:** A warning icon followed by the text '元数据不支持增量，每次备份都为全量备份。' (Metadata does not support incremental, each backup is a full backup). Below this are three radio button options:
 - '首次全量备份，后续增量备份' (First full backup, subsequent incremental backup) - selected
 - '每次都全量备份' (Full backup every time)
 - '每n次进行一次全量备份' (Full backup every n times)

HBase备份

备份策略

- “任务名称”，填写备份任务的名称。
- “备份类型”，选择备份任务的运行类型，“周期备份”表示按周期自动执行备份，“手动备份”表示由手工执行备份。
- 创建周期备份任务，还需要填写以下参数：
 - “开始时间”，表示任务第一次启动的时间。
 - “周期”，表示任务下次启动，与上一次运行时间间隔。支持“按小时”或“按天”。
 - “备份策略”，表示任务每次启动时备份的数据量。支持“首次全量备份，后续增量备份”、“每次都全量备份”和“每n次进行一次全量备份”。选择“每n次进行一次全量备份”时，需要指定n的值。

HBase备份

[illegible]

HBase备份

- 备份配置
 - “业务数据”，勾选“**HBase**”（多实例下可勾选其他**HBase**实例）。
 - “路径类型”，选择“**RemoteHDFS(DistCp)**”。
 - “目的端**NameNode** IP地址”，填写备集群**NameNode**业务平面IP地址，支持主节点或备节点
 - “目的端路径”，填写备集群保存备份数据的**HDFS**目录。不支持填写**HDFS**的隐藏目录，例如快照或回收站目录；不支持默认的系统目录，例如“**/hbase**”或“**/user/hbase/backup**”。
 - “最大备份数”，填写备份目录中可保留的备份文件集数量。已备份的文件集数量大于“最大备份数”参数值时，默认保留最新的文件。
 - “队列名称”，填写备份任务执行时使用的**YARN**队列的名称。需和集群中已存在且状态正常的队列名称相同。
 - “备份对象”，勾选这个备份任务需要备份的用户表。支持正则表达式匹配。

HBase数据恢复

- 在用户意外修改删除、数据需要找回，系统管理员对**HBase**进行重大操作（如升级、重大数据调整等）后系统数据出现异常或未达到预期结果，模块全部故障完全无法使用，或者迁移数据到新集群的场景中，需要对**HBase**进行恢复数据操作。
- 恢复内容为**HBase**表级别的用户数据。

HBase数据恢复

- 对系统的影响
 - 恢复过程的数据还原阶段，系统会把待恢复的**HBase**表禁用，此时无法访问该表。还原阶段可能需要几分钟时间，此时**HBase**的上层应用无法正常工作。
 - 恢复过程中会停止用户认证，用户无法开始新的连接。
 - 数据恢复后，会丢失从备份时刻到恢复时刻之间的数据。
 - 数据恢复后，需要重新启动**HBase**的上层应用。
- 恢复前准备和约束
 - 主备集群必须已配置跨集群互信。
 - 主备集群必须已配置跨集群拷贝。
 - 主备集群上的时间必须一致，而且主备集群上的**NTP**服务必须使用同一个时间源。
 - 根据业务需要，规划恢复任务的类型、周期、命名空间和表等策略规格，确认备份任务需要使用的**YARN**队列。
 - 检查**HBase**备份文件保存路径。
 - 停止**HBase**的上层应用。

HBase数据恢复

恢复配置

* 任务名称:

adfs

业务数据

☒ HBase

路径配置

* 路径类型:

RemoteHDFS(DistCp)

* 源端NameNode IP地址:

10.64.35.145

* 源端路径:

/backup/hbase/abc_HBase_20160427090512

* 队列名称:

default

* 恢复点列表:

abc_HBase_HBase_20160427110508

数据配置

备份数据

defaultt1

ns1:test

当前页: 1 总页数: 1 | < < > > | 跳转到 确定

数据选择:

强制覆盖:

☒ 是 ☐ 否

HBase数据恢复

- 恢复配置
 - “业务数据”，勾选“**HBase**”（多实例下可勾选其他**HBase**实例）。
 - “路径类型”，选择“**RemoteHDFS(DistCp)**”。
 - “源端**NameNode** IP地址”，填写源集群**NameNode**业务平面IP地址，支持主节点或备节点。
 - “源端路径”，填写源集群保存备份数据的**HDFS**目录。
 - “队列名称”，填写备份任务执行时使用的**YARN**队列的名称。需和集群中已存在且状态正常的队列名称相同。
 - “恢复点列表”，拉选所需恢复的恢复点。



目录

1. 服务检查与日志
2. 维护命令与参数
3. **HBase**备份与恢复
4. 性能调优命令与参数
 - 提升实时写效率
 - 提升实时读效率
5. 常见问题与定位

提升实时写效率

- 操作场景
 - 需要把数据实时写入到**HBase**中或者对于大批量、连续**put**的场景。
- 前提条件
 - 调用**HBase**的**put**或**delete**接口，把数据保存到**HBase**中。

写数据服务端调优

配置参数	描述	默认值
hbase.regionserver.wal.durable.sync	控制HLog文件在写入到HDFS时的同步程度。如果为 true ，HDFS在把数据写入到硬盘后才返回；如果为 false ，HDFS在把数据写入OS的缓存后就返回。把该值设置为 false 比 true 在写入性能上会更优。	true
hbase.regionserver.hfile.durable.sync	控制HFile文件在写入到HDFS时的同步程度。如果为 true ，HDFS在把数据写入到硬盘后才返回；如果为 false ，HDFS在把数据写入OS的缓存后就返回。把该值设置为 false 比 true 在写入性能上会更优。	true

写数据服务端调优

配置参数	描述	默认值
GC_OPTS	<p>HBase利用内存完成读写操作。提高HBase内存可以有效提高HBase性能。GC_OPTS主要需要调整HeapSize的大小和NewSize的大小。调整HeapSize大小的时候，建议将Xms和Xmx设置成相同的值，这样可以避免JVM动态调整HeapSize大小的时候影响性能。调整NewSize大小的时候，建议把其设置为HeapSize大小的1/9。</p> <ul style="list-style-type: none">● HMaster：当HBase集群规模越大、Region数量越多时，可以适当调大HMaster的GC_OPTS参数。● RegionServer：RegionServer需要的内存一般比HMaster要大。在内存充足的情况下，HeapSize可以相对设置大一些。 <p>说明：主HMaster的HeapSize为4G的时候，HBase集群可以支持100000Region数的规模。根据经验值，单个RegionServer的HeapSize不建议超过20GB。</p>	<ul style="list-style-type: none">● HMaster： -Xms2G -Xmx2G -XX:NewSize=256M - XX:MaxNewSize=256M● RegionServer： -Xms4G -Xmx4G -XX:NewSize=512M - XX:MaxNewSize=512M

写数据服务端调优

配置参数	描述	默认值
hbase.regionserver.handler.count	表示 RegionServer 在同一时刻能够并发处理多少请求。如果设置过高会导致激烈线程竞争，如果设置过小，请求将会在 RegionServer 长时间等待，降低处理能力。根据资源情况，适当增加处理线程数。建议根据 CPU 的使用情况，可以选择设置为 100 至 300 之间的值。	100
hbase.hregion.max.filesize	表示 HBase 中 Region 的文件总大小的最大值。当 Region 中的文件大于该参数时，将会导致 Region 分裂。该参数设置过小时，可能会导致 Split 操作过于频繁。当设置过大时，可能导致 Compact 需要处理的文件大小增加，影响 Compact 执行效率。	10737418240 (单位：字节)

写数据服务端调优

配置参数	描述	默认值
hbase.hregion.memstore.flush.size	<p>在RegionServer中，当写操作内存中存在超过memstore.flush.size大小的memstore，则MemStoreFlusher就启动flush操作将该memstore以hfile的形式写入对应的store中。如果RegionServer的内存充足，而且活跃Region数量也不是很多的时候，可以适当增大该值，可以减少compaction的次数，有助于提升系统性能。</p> <p>同时，这种flush产生的时候，并不是紧急的flush，flush操作可能会有一定延迟，在延迟期间，写操作还可以进行，Memstore还会继续增大，最大值为“memstore.flush.size” * “hbase.hregion.memstore.block.multiplier”。当超过最大值时，将会阻塞操作。适当增大“hbase.hregion.memstore.block.multiplier”可以减少阻塞，减少性能波动。</p>	134217728 (单位：字节)

写数据服务端调优

配置参数	描述	默认值
hbase.regionserver.global.memstore.size	<p>RegionServer中，负责flush操作的是MemStoreFlusher线程。该线程定期检查写操作内存，当写操作占用内存总量达到阈值，MemStoreFlusher将启动flush操作，按照从大到小的顺序，flush若干相对较大的memstore，直到所占内存小于阈值。</p> <p>阈值 = "hbase.regionserver.global.memstore.size" * "hbase.regionserver.global.memstore.size.lower.limit" * "HBase_HEAPSIZE"</p>	0.4

写数据服务端调优

配置参数	描述	默认值
hbase.hstore.blockingStoreFiles	<p>在region flush前首先判断file文件个数，是否大于hbase.hstore.blockingStoreFiles。如果大于需要先compaction 并且让flush延时90s（这个值可以通过hbase.hstore.blockingWaitTime进行配置），在延时过程中，将会继续写从而使得Memstore还会继续增大超过最大值 “memstore.flush.size” * “hbase.hregion.memstore.block.multiplier”，导致写操作阻塞。当完成compaction后，可能就会产生大量写入。这样就导致性能激烈震荡。</p> <p>增加hbase.hstore.blockingStoreFiles，可以减低BLOCK几率。</p>	7

写数据服务端调优

配置参数	描述	默认值
hbase.regionserver.thread.compaction.throttle	控制一次 Minor Compaction 时，进行 compaction 的文件总大小的阈值。 Compaction 时的文件总大小会影响这一次 compaction 的执行时间，如果太大，可能会阻塞其它的 compaction 或 flush 操作。	1610612736 (单位：字节)
hbase.hstore.compaction.min	当一个 Store 中文件超过该值时，会进行 compact ，适当增大该值，可以减少文件被重复执行 compaction 。但是如果过大，会导致 Store 中文件数过多而影响读取的性能。	3

写数据服务端调优

配置参数	描述	默认值
hbase.hstore.compaction.max	控制一次 compaction 操作时的文件数量的最大值。与“ hbase.hstore.compaction.max.size ”的作用基本相同，主要是控制一次 compaction 操作的时间不要太长。	10
hbase.hstore.compaction.max.size	如果一个 HFile 文件的大小大于该值，那么在 Minor Compaction 操作中不会选择这个文件进行 compaction 操作，除非进行 Major Compaction 操作。这个值可以防止较大的 HFile 参与 compaction 操作。在禁止 Major Compaction 后，一个 Store 中可能存在几个 HFile ，而不会合并成为一个 HFile ，这样不会对数据读取造成太大的性能影响。	2684354560 (单位：字节)

写数据服务端调优

配置参数	描述	默认值
hbase.hregion.majorcompaction	<p>设置Major Compaction的执行周期。默认值为86400000毫秒，也就是1天。</p> <p>由于执行Major Compaction会占用较多的系统资源，如果正在处于系统繁忙时期，会影响系统的性能。如果业务没有较多的更新、删除、回收过期数据空间时，可以把该值设置为0，以禁止Major Compaction。</p> <p>如果必须要执行Major Compaction，以回收更多的空间，可以适当增加该值，同时配置参数“hbase.offpeak.end.hour”和“hbase.offpeak.start.hour”以控制Major Compaction发生在业务空闲的时期。</p>	86400000 (单位：毫秒)

写数据服务端调优

配置参数	描述	默认值
<ul style="list-style-type: none">• hbase.regionserver.maxlogs• hbase.regionserver.hlog.blocksize	<ul style="list-style-type: none">• 表示一个RegionServer上未进行Flush的Hlog的文件数量的阈值，如果大于该值，RegionServer会强制进行flush操作。• 表示每个HLog文件的最大大小。如果HLog文件大小大于该值，就会滚动出一个新的HLog文件，旧的将被禁用并归档。 <p>这两个参数共同决定了RegionServer中可以存在的未进行Flush的hlog数量。当这个数据量小于MemStore的总大小的时候，会出现由于HLog文件过多而触发的强制flush操作。这个时候可以适当调整这两个参数的大小，以避免出现这种强制flush的情况。</p>	<ul style="list-style-type: none">• 32• 67108864 (单位：字节)

写数据客户端调优

- 写数据时，在场景允许的情况下，最好使用**Put List**的方式，可以极大的提升写性能。每一次**Put**的**List**的长度，需要结合单条**Put**的大小，以及实际环境的一些参数进行设定。建议在选定之前先做一些基础的测试。

写数据表设计调优

配置参数	描述	默认值
COMPRESSION	<p>配置数据的压缩算法，这里的压缩是HFile中block级别的压缩。对于可以压缩的数据，配置压缩算法可以有效减少磁盘的IO，从而达到提高性能的目的。</p> <p>说明：并非所有数据都可以进行有效压缩。例如一张图片的数据，因为图片一般已经是压缩后的数据，所以压缩效果有限。常用的压缩算法是SNAPPY，因为它有较好的Encoding/Decoding速度和可以接受的压缩率。</p>	NONE

写数据表设计调优

配置参数	描述	默认值
BLOCKSIZE	<p>配置HFile中block块的大小，不同的block块大小，可以影响HBase读写数据的效率。越大的block块，配合压缩算法，压缩的效率就越好；但是由于HBase的读取数据是以block块为单位的，所以越大的block块，对于随机读的情况，性能可能会比较差。</p> <p>如果要提升写入的性能，一般扩大到128KB或者256KB，可以提升写数据的效率，也不会影响太大的随机读性能。</p>	65536 (单位：字节)
IN_MEMORY	<p>配置这个表的数据优先缓存在内存中，这样可以有效提升读取的性能。对于一些小表，而且需要频繁进行读取操作的，可以设置此配置项。</p>	false

3.2 提升实时读效率

- 操作场景
 - 需要读取**HBase**数据场景。
- 前提条件
 - 调用**HBase**的**get**或**scan**接口，从**HBase**中实时读取数据。

读数据服务端调优

配置参数	描述	默认值
GC_OPTS	<p>HBase利用内存完成读写操作。提高HBase内存可以有效提高HBase性能。GC_OPTS主要需要调整HeapSize的大小和NewSize的大小。调整HeapSize大小的时候，建议将Xms和Xmx设置成相同的值，这样可以避免JVM动态调整HeapSize大小的时候影响性能。调整NewSize大小的时候，建议把其设置为HeapSize大小的1/9。</p> <ul style="list-style-type: none">• HMaster：当HBase集群规模越大、Region数量越多时，可以适当调大HMaster的GC_OPTS参数。• RegionServer：RegionServer需要的内存一般比HMaster要大。在内存充足的情况下，HeapSize可以相对设置大一些。 <p>说明：主HMaster的HeapSize为4G的时候，HBase集群可以支持100000Region数的规模。根据经验值，单个RegionServer的HeapSize不建议超过20GB。</p>	<ul style="list-style-type: none">• HMaster： -Xms2G -Xmx2G -XX:NewSize=256M -XX:MaxNewSize=256M• RegionServer： -Xms4G -Xmx4G -XX:NewSize=512M -XX:MaxNewSize=512M

读数据服务端调优

配置参数	描述	默认值
hbase.regionserver.handler.count	该参数表示 RegionServer 在同一时刻能够并发处理多少请求。如果设置过高会导致激烈线程竞争，如果设置过小，请求将会在 RegionServer 长时间等待，降低处理能力.根据资源情况，适当增加处理线程数。建议根据 CPU 的使用情况，可以选择设置为 100 至 300 之间的值。	100
hfile.block.cache.size	HBase 缓存区大小，主要影响查询性能。根据查询模式以及查询记录分布情况来决定缓存区的大小。如果采用随机查询使得缓存区的命中率较低，可以适当降低缓存区大小。	0.25



目录

1. 服务检查与日志
2. 维护命令与参数
3. **HBase**备份与恢复
4. 性能调优命令与参数
5. 常见问题与定位

常见问题

- **HBase**组件客户端出现异常，通常分为以下几类：
 - **HBase**操作异常，如执行建表、**Put**、**Scan**时，抛出异常；
 - **HBase Master**的**Web**操作页面无法打开；
 - **HBase Shell**命令执行无法返回正确结果；
 - **HBase**集群节点进程异常退出。

可能原因

- **HBase**客户端程序异常
 - **HBase**服务或客户端环境出现异常（资源不足，网络异常，磁盘故障等）；
 - **HBase**依赖的组件无法提供服务，如**HDFS**、**ZooKeeper**出现异常；
 - **HBase**服务出现异常。

定位思路

1、排查客户端程序。

- ① 确保**HBase**客户端程序引用的**jar**包与**HBase**集群的**jar**包一致。

HBase客户端程序需要引用的**jar**包含：

HBase程序**jar**、**HDFS**基础**jar**、**ZooKeeper**基础**jar**以及与**HBase**、**ZooKeeper**、**HDFS**组件关联的第三方**jar**包。建议引用**HBase lib**库下的所有**jar**文件。

- ② 确保**HBase**客户端安装的**Java**版本与**HBase**集群安装的**Java**版本一致。
- ③ 确保**HBase**客户端程序使用的**hbase-site.xml**、**core-site.xml**、**hdfs-site.xml**配置文件和**HBase**集群一致。

定位思路

2、查看HBase集群及客户端环境是否可用。

- ① 确保HBase客户端程序环境与HBase集群服务器网络连接正常，可使用telnet命令进行验证。
- ② 确保HBase客户端程序环境的hosts文件中，正确添加了HBase集群服务器IP地址和HostName对应关系。
- ③ 在HBase集群中，omm用户下，可通过ps -ef | grep HRegionServer查看RegionServer进程，ps -ef | grep HMaster 查看HMaster进程。

3、排查HBase依赖的集群是否正常。

- ① 在FusionInsight HD Manager上，查看ZooKeeper集群是否运行正常。
- ② 在FusionInsight HD Manager上，查看HDFS集群是否运行正常。
- ③ 在FusionInsight HD Manager上，查看Yarn集群是否运行正常。

定位思路

4、HBase Shell中的常用命令是否还可用：

- ① 可以在`$HBASE_HOME/bin`路径下，执行`./hbase shell`。
- ② 执行常用的命令，如`list`,`scan 'tablename'`等操作，观察一下是否会有异常信息。

5、使用**hbck**工具进行检查，如发现问题，可反复多次执行。因为该工具的每次执行，都仅仅检查执行时间点的集群状态，该状态可能正在发生变化，如当前可能是异常的，但集群却正在尝试着恢复。

- ① 在`$HBASE_HOME/bin`路径下，执行`./hbase hbck`。
- ② 在出现的结果中，可以查看到集群是否有异常信息。
- ③ 如果有异常，可以反复多执行几次，看该异常是否始终存在。如果已经不存在，则认为集群出现了异常。

定位思路

6、使用**fsck**工具检查**HDFS**集群的健康状态：在**hdfs**的**namenode**的**bin**路径下面，运行命令：`./hadoop fsck /`

DNS无法解析

I. 现象描述

客户端日志中，类似于如下信息的日志：“**UnknownHostException**”

II. 可能原因

HBase集群中，各节点之间基本上是依靠节点的**HostName**来识别并且进行交互的，因此，如果**HostName**没有能够正确配置的话，会导致一系列异常。同理，**HBase**客户端也需要配置集群各节点的**HostName**和**IP**的对应关系才可以正常的连接到**HBase**集群。

III. 定位思路

检查“**/etc/hosts**”中是否配置了集群中各个节点的**IP**和**HostName**的映射关系

IV. 处理步骤

- ① 如果是在**Window**环境下，在文件“**C:\Windows\System32\drivers\etc\hosts**”中添加集群各节点**Ip**与**HostName**的映射关系
- ② 如果是在**Linux**环境下，在文件“**/etc/hosts**”中添加**Ip**与**HostName**的映射关系

客户端登录失败

I. 现象描述

HBase客户端程序启动并且登录时，登录失败。

II. 可能原因

- ①客户端环境中没有正确配置**krb5**文件。
- ②客户端环境的配置文件**hbase-site.xml**中没有配置访问**HBase**集群的用户名和密码文件等信息，或者所提供的密码文件已经过期。
- ③客户端环境中的**Jar**包与集群中的**Jar**包不一致。
- ④客户端环境中的时间和集群的时间不一致，且相差超出**5**分钟

客户端登录失败

III. 处理步骤

- ① 检查本地的**krb5**配置文件，是否与集群中的配置一致。

Windows环境中的配置文件路径：**C:\Windows\krb5.ini**

Linux环境中的配置文件路径：**/etc/krb5.conf**

集群中配置在安装了**KerberosServer**节点上，

/opt/huawei/Bigdata/FusionInsight/etc/KerberosClient目录下

- ② 检查本地的**hbase-site.xml**中的如下两个配置项是否正确：

username.client.kerberos.principal

username.client.keytab.file

- ③ 检查本地环境的时间是否与集群中的时间一致。
- ④ 检查本地环境的**Jar**包与集群中的**Jar**包是否一致（重点检查**HBase**，**ZooKeeper**，**Hadoop**相关的**Jar**包）。

客户端 “SASL authentication failed”

I. 现象描述

客户端进程不能正常访问**HBase**集群，出现如下类似错误打印信息：

**“SASL authentication failed. The most likely cause is missing or invalid credentials.
Consider 'kinit'. ”**

II. 可能原因

- 1、客户端进程中主动调用了多次**User#login**方法进行登录。
- 2、用户名或**keytab**文件不正确。

III. 定位思路

- ① 修改客户端代码，避免进程内多重登录。
- ② 排查修改客户端多重登录代码后，重启客户端即可恢复正常。正常消耗时间**10**分钟。

IV. 处理步骤

- ① 检查客户端日志中是否有多次如下类似日志打印：**“Login successful for user XXXXX
using keytab file XXXXXX”**。如果有多次，则说明曾经登录过多次。
- ② 对客户端代码进行排查，一定要避免进程级别内重复登录。

Region无法被打开

I. 现象描述

- 集群中各**RegionServer**的**Region**数目不均衡，并且持续较长时间。
- 通过**hbck**工具检查，发现有个别**Region**没有被打开。

II. 可能原因

目前我们发现的引起**Region**无法打开的可能的原因，包括：

- ① **HMaster**进程异常，或者无可用的**RegionServer**。
- ② 底层**HDFS**，**Zookeeper**异常导致的**HBase**进程异常导致。

Region无法被打开

III. 定位思路

可以首先查看**Master**的日志，查找出现异常的Region的相关日志，查看最后一次**assign**操作将**Region**分配到了哪个**RegionServer**上面，接下来，到相应的**RegionServer**上面去查看相应的**Region**的信息。

① **HMaster**下发打开命令的相关日志：“**DEBUG**

**org.apache.hadoop.hbase.master.AssignmentManager: Assigning region
***** to #####**”

② **HRegionServer**收到打开**Region**的相关日志：“**INFO**

**org.apache.hadoop.hbase.regionserver.HRegionServer: Received request to
open region: *******”

③ **HRegionServer**成功打开**Region**的日志：“**DEBUG**

**org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler: Opened
*******”

④ 整个**Region**打开流程结束的日志(**HMaster**日志)“**DEBUG**

**org.apache.hadoop.hbase.master.handler.OpenedRegionHandler: Opened
region *******”

Region无法被打开

IV.处理步骤

- ① 如果是因为**Region**打开流程出现了短暂的异常导致的**Region**长时间未能打开，可以直接在**hbase shell**中手动**assign**。
- ② 如果是因为**Zookeeper**或者**HDFS**进程异常的原因导致的**Region**无法打开，请首先确认**Zookeeper**或**HDFS**的异常。有一些情况下的异常是致命的，建议对**HBase**集群进行重启，例如，大部分的**Zookeeper**进程都已经异常了。

ConnectionLossException

I. 现象描述

该异常，可能会出现在HMaster,HRegionServer或者Client端。

II. 可能原因

Zookeeper问题分为两大类：

- **Recoverable errors**与**Unrecoverable errors**。**Recoverable errors**，我们可以认为是可以自动恢复的错误，而**Unrecoverable errors**则是不可恢复的。
- **HBase**依靠**Zookeeper**来进行分布式存储系统的协调服务。**HMaster**和**HRegionServer**进程在初始化的时候，均需要初始化**Zookeeper**的客户端实例。它们之间依靠**Session**来维持**Zookeeper**客户端以及**Zookeeper**服务器端的连接的有效性。**Session**会话的有效性根据客户端以及服务器端的心跳来进行维持。**HMaster**和**HRegionServer**作为**Zookeeper**的客户端，一旦超过一定的时间（该时间可配）没有能够与**Zookeeper**服务器端取得联系，**Session**就会过期，当再次取得联系以后，**Zookeeper**服务器端就会发送一个**Session Expired**事件过来。在通常情况下，**HMaster**和**HRegionServer**在收到这个事件以后，会退出。可能原因：
 - ① **JVM**在长时间的执行**GC**操作，导致线程阻断。
 - ② **RegionServer**的**Zookeeper**客户端因本地的原因，长时间无法发送心跳信息给**Zookeeper**服务器端，造成**Session Timeout**。

ConnectionLossException

III. 定位思路

因为**ConnectionLossException**，很有可能会造成**Zookeeper Session Expired**。

Zookeeper Session Expired定位思路：

- ① 如果环境中打开了**GC**日志，请从**GC**日志中确认是否存在长时间的**FullGC**。在**HBase**日志中，如果出现了如下类似的日志信息，就可能是**FullGC**问题导致的：
 - **2009-02-24 10:01:33,516 WARN org.apache.hadoop.hbase.util.Sleeper: We slept xxx ms, ten times longer than scheduled: 10000**
 - **2009-02-24 10:01:36,472 WARN org.apache.hadoop.hbase.regionserver.HRegionServer: unable to report to master for xxx milliseconds - retrying**
- ② 检查集群环境中是否为**HMaster, HRegionServer**配置足够的**HeapSize**
- ③ 检查环境中是否关闭了**Swap**, **JVM**在**Swap**的环境下，容易出问题
- ④ 检查集群环境中是否已经存在资源不足问题（**CPU**内存资源不足，硬盘空间不足）

ConnectionLossException

IV.处理步骤

- ① 如果**HMaster,HRegionServer**的**HeapSize**过小，请在**manager**页面上配置
- ② 如果集群中**Swap**未关闭，采用如下方法关闭**Swap**：
执行命令：**sysctl vm.swappiness=0**。为了让配置长期有效，需要在**/etc/sysctl.conf**中添加：**vm.swappiness = 0**
- ③ 资源不足问题，请根据具体的问题做相应的处理

思考题

1. **HMaster**服务异常，可以查看哪些日志？
2. **HBase**表的**BLOCKSIZE**属性是否越大越好？

习题

多选题

1. 以下关于**Hbase**日志保存位置的描述，正确的是（ ）？
- A. **Hmaster**的运行日志保存在：**/var/log/Bigdata/hbase/hm**
 - B. **Hmaster**的审计日志保存在：
/var/log/Bigdata/audit/hbase/hm
 - C. **RegionServer**的运行日志保存在：**/var/log/Bigdata/hbase/rs**
 - D. **RegionServer**的审计日志保存在：
/var/log/Bigdata/audit/hbase/rs



本章总结

培训与认证部

学习完本章，应该需要：

- 1、了解到**HBase**的基础目录结构，日志目录结构；
- 2、了解到**HBase**调优参数的设置；
- 3、了解到**HBase**基础问题定位的方法步骤。

Thank you

www.huawei.com