|  |  |
| --- | --- |
| **文档版本** | V1.0 |
| **更新日期** | 2025-09-28 |
| 作者 | 罗虎 |

# HBase同步优化

参考 Cloudera Runtime 7.2.16 «HBase Replication» 官方文档，总结如下优化要点，帮助在跨集群或跨园区场景中提升增量同步的稳定性与吞吐。内容按规划、配置、运维三个维度展开。

## 1. 场景与目标

- 满足跨数据中心、同城双活、容灾备份等复制需求，保障 RPO≈0、RTO 短。

- 在不影响主集群前台读写的前提下，提高 WAL 复制效率，降低积压。

- 明确容量上限、监控指标与回退机制，确保持续可运维。

## 2. 常见复制拓扑（官方示例）

根据官方文档总结，以下拓扑在实际部署中最常见：

- 单中心多目标：一个中心源集群同时向多个目标集群推送变更，以满足异地多活或容灾需求。

- 双向互备：源集群向目标集群复制的同时，目标集群也会把自身写入回推到源端，须做好冲突处理与访问隔离。

- 星型汇聚：多个低延迟业务集群将数据上送至中心集群，用于备份或离线分析；分析结果再同步回业务集群。

- 多级链式：复制链路可按需要多级串联，形成主 → 中间站 → 终端的阶梯式容灾体系。

## 3. 复制前置条件

- 版本互通：支持从 CDH 5 复制至 Cloudera Private Cloud Base（CP Base），以及从 CP Base 回传至 CDH 5，前提是 CP Base 版本 ≥ 7.1。

- 全集群互通：两个集群的所有主机（含 ZooKeeper 节点）需互相可达，确保 WAL 推送与元数据同步无阻。

- 表结构一致：所有已设置复制作用域的列族所在表必须在各集群中预先存在且命名一致；若目标端未建表，需先按源端 schema 创建空表。

- HDP 兼容：若源端为 HDP 3.1.5，可在目标端 CP Base 版本 ≥ 7.1.6 的情况下建立复制链路。

## 4. 关键注意事项

- 时间戳保持不变：复制后的 HLog 条目保留原始时间戳，若出现相同行键、列族、列限定符与时间戳的冲突，以后到达的写入可见，需评估冲突概率。

- 增量计数器：Increment Column Values 在复制链路中按普通 Put 处理，若双向活跃写入同一字段，将互相覆盖；建议不同集群不要对同一坐标执行递增操作，避免 HBASE-2804 所描述的计数漂移。

- 时钟同步：源与目标集群的时钟必须同步，推荐统一使用 NTP，避免因时间漂移导致版本顺序异常。

- 非复制变更：以下变更不会自动复制，需借助 Snapshot、CopyTable 或 DistCp 等方式补齐：

- 复制开启前已存在的数据。

- 绕过 WAL 的写入（BulkLoad 或 writeToWal(false) 调用）。

- 表结构修改（DDL 变更）。

- 复制控制变量：hbase.replication 与 ZooKeeper 上的复制 znode 共同决定复制状态；当执行 disable\_table\_replication 或长时间停用 peer 时，新加入的 RegionServer 或滚动产生的新 WAL 可能因 znode=FALSE 而未入复制队列，恢复后也不会自动补齐，须谨慎评估停机窗口并视情况手动滚动 WAL。

- ACL 同步差异：名称空间、表、列族或单元上的 ACL 不会随数据复制，目标端需手动重建，对多租户环境要做好访问控制比对。

- WAL 滚动影响：新加入的 peer 默认仅收到 WAL 滚动之后的写入；若在引入 peer 前发生滚动，先前写入不会自动回补，可在切换前手动滚动 WAL 并验证数据完整性。

## 5. 安全复制配置

1. Kerberos 基础：复制双方均需启用 Kerberos；不支持从非 Kerberos 集群向 Kerberos 集群复制。若 ZooKeeper 使用自定义 principal，安全复制不可用。

2. 跨域互信：若存在跨 Realm 复制需求，通过 list\_principals 等命令确认各 Realm 的 kdc、admin\_server、default\_domain，并在 Cloudera Manager 的 HDFS 配置中填写可信 Kerberos Realm，重启 HDFS 生效。

3. ZooKeeper 配置：确保 ZooKeeper 节点按照安全指南完成 Kerberos 与 ACL 设置。

4. HDFS 加密链路：在双方集群启用数据传输加密与鉴权（如 dfs.encrypt.data.transfer=true、dfs.data.transfer.protection=privacy、hadoop.security.authorization=true、hadoop.rpc.protection=privacy 等）。使用 CM 管理时需在界面上逐项启用对应配置。

5. HBase 安全参数：统一设置 hbase.security.authentication=kerberos、hbase.security.authorization=true、hbase.rpc.protection=privacy、hbase.ssl.enabled=true 等参数，并开启 Thrift/REST 的 SSL 与 auth-conf。

6. 安全 Peer 定义：通过 add\_peer 注入 peer 时可直接指定 CLUSTER\_KEY。若某个 Peer 需自定义 principal/keytab，使用扩展语法在 CONFIG 中传入 hbase.master.kerberos.principal、hbase.regionserver.keytab.file 等字段，实现按 peer 定制的跨域可信规则。

## 6. Bulk Load 复制注意

- Bulk Load 通过直接生成 HFile 并导入 RegionServer，绕过常规写链路与 WAL；若不额外配置，数据不会在复制链路中同步。

- 对于频繁执行 Bulk Load 的场景，需启用官方提供的 Bulk Load Replication 能力，确保 HFile 变更写入复制队列；同时评估额外的网络与存储压力。

- 若暂无法启用 Bulk Load 复制，可在装载后使用 DistCp、Snapshot 恢复等方式补齐备份集群的数据，避免副本不一致。

## 7. 目标端表结构准备

- 当目标集群尚未创建待复制表时，可在源集群 HBase Shell 执行 describe <table> 获取列族定义。

- 按官方建议将 TTL => FOREVER 替换为 TTL => org.apache.hadoop.hbase.HConstants::FOREVER，在表名前添加 CREATE 关键字，并保留列族属性段落。

- 将整理后的 CREATE '<table>', {NAME => 'cf', ...} 语句粘贴到目标集群 HBase Shell 执行，即可生成与源端一致的空表结构。

## 8. 部署流程指引

1. 使用 HBase 超级用户（默认 hbase）执行所有与复制相关的命令；若 ZooKeeper 启用了 Kerberos，需先在 HBase Shell 中完成认证。

2. 启动并配置好源、目标集群，确保双方所有主机（含 ZooKeeper）互相可达；目标集群提前创建与源端同名、同列族的表。

3. 在源集群启用复制，可通过 Cloudera Manager 或手动在 hbase-site.xml 设置 hbase.replication=true。

4. 以 HBase Principal 获取 Kerberos 凭据，例如：kinit -k -t /etc/hbase/conf/hbase.keytab hbase/FQDN@REALM。

5. 在源集群 HBase Shell 中执行 add\_peer 'peerId', 'CLUSTER\_KEY'，其中 CLUSTER\_KEY 形如 zk1,zk2,zk3:2181:/hbase。

6. 为需要同步的列族设置 REPLICATION\_SCOPE=1，操作顺序为 disable 表 → alter 列族 → enable 表。

7. 通过源端日志确认复制是否开始（示例：Choosing peer ...），并可运行 hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication <peerId> <table> 校验数据一致性，结合 --starttime、--families 等参数精准验证。

8. 若需对已有存量数据构建同步链路：

- 在 add\_peer 后立刻执行 disable\_peer('peerId')，让新增变更暂存于源端复制队列。

- 对每张表执行 snapshot '<table>', '<snapshot-name>'，确保内存数据刷新并生成快照。

- 使用 ExportSnapshot 将快照从源端导出并传输到目标集群，例如 hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot -snapshot MySnapshot -copy-to hdfs://dest:8020/hbase\_root\_dir -mappers 16。

- 在目标集群导入并 restore\_snapshot，恢复快照生成的 HFile；安全集群请参照“安全复制配置”章节同步 Kerberos 配置。

- 完成快照恢复后执行 enable\_peer('peerId')，源端会自动回放暂存的增量日志，随后进入常规增量复制。

## 9. 架构规划

- 复制拓扑：优先选择主被动（一主多备）拓扑，避免环形复制放大网络流量；需要双活时，务必拆分写入热点并启用冲突解决策略。

- Region 分布：保持 Region 严格均衡，减少单台 RegionServer 的复制压力；热点表可采用预分区或局部索引优化写入分布。

- 命名一致性：保证源端与目标端的表结构、命名空间一致，避免复制失败后回放困难。

## 10. 关键配置调优

- WAL 管理：主集群适当提升 hbase.wal.provider 为异步 FSHLog，并确保 WAL 目录使用独立 SSD；将 WAL 滚动阈值设置为合适大小（如 512MB）以减少小文件。

- 复制线程：依据 RegionServer CPU/带宽，适度提升复制源线程池（Cloudera 文档建议保持线程数与可复制队列平衡，默认 10~20 可逐步递增，避免盲目翻倍）。

- 批次与压缩：启用 WAL 压缩（hbase.regionserver.wal.enablecompression=true），并在目标端开启 hbase.rpc.compression，降低网络传输量；适度调大单批次条目数以减少 RPC 次数。

- 限流与重试：利用 replication throttling（在 CM > HBase > Configuration 搜索 Replication Throttle Bandwidth）限制带宽峰值；合理设置失败重试间隔，防止频繁重连造成雪崩。

- 序列化方式：按官方推荐启用 protobuf codec，并保持源/目标端版本一致，降低兼容性风险。

- 目的端并发：针对写入密集场景，可提升 hbase.regionserver.replication.handler.count、replication.sink.client.retries.number 与 replication.sink.client.ops.timeout 等参数，增强目的端复制线程数、重试次数与写入超时容忍度。

## 11. 运维与监控

- CM 监控：重点关注以下指标：

- Replication Lag（源与目的的最新 SequenceId 差值）。

- WAL Files Queued（排队 WAL 数量）。

- Replication RS Ops/sec（每秒复制操作数）。

- 报警策略：针对延迟、队列长度、失败率设阈值，确保复制延迟超 5 分钟即报警；结合 SMM/SMM-UI 分析 RPC 错误码，快速定位网络或权限问题。

- 压测验证：在生产变更前通过 Tooling（如 YCSB、hbck2）进行容量测试，确认升级后延迟与吞吐改善效果。

- VerifyReplication 校验：建议在源或目的端部署 YARN（无 YARN 可用时可启用本地模式但性能受限），并确保 hbase 用户具备 YARN 白名单及 /user/hbase HDFS 目录权限。通过 sudo -u hbase hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication peerId table 执行校验，可结合 --starttime/--endtime/--versions/--families 精准限定时间窗或列族；关注计数器含义如下：

- GOODROWS：源、目标一致的行数。

- CONTENT\_DIFFERENT\_ROWS：键一致但值不同的行。

- ONLY\_IN\_SOURCE\_TABLE\_ROWS：仅存在于源端的行。

- ONLY\_IN\_PEER\_TABLE\_ROWS：仅存在于目标端的行。

- BADROWS：差异行总数，等于前三项之和。

## 12. 故障与回退

- 积压处理：当 WAL 积压严重时，可临时提高复制线程或清理无关队列；必要时停用低优先级的复制对等端，先恢复核心业务。

- 黑名单机制：通过 Cloudera Manager 下发 Exclude Tables 黑名单，避免开发测试表拖累整体复制。

- 回放验证：在目标集群执行 verifyrep 与 count 校验；对重要表启用 Snapshots + DistCp 双重兜底。

- 按 Peer 暂停：使用 disable\_peer("peerId") 暂停指定对等端，复制日志仍会累积以备补回；待恢复时执行 enable\_peer("peerId") 即可重新同步。

- 应急停复：遇到严重故障时，可在源集群逐个执行 disable\_peer("peerId") 并结合 disable\_table\_replication '<table>'，阻止新写入进入复制链路；队列中已排队的日志仍会回放。恢复时按需执行 enable\_peer("peerId")。

## 13. 操作建议

1. 定期复查复制 Peer 配置（认证、ACL、序列化方式），配合 CM 模板统一管理。

2. 结合 hbase shell 的 list\_replicated\_tables/status 'replication' 定期巡检。

3. 建立 SOP：扩容 RegionServer → 调整复制参数 → 验收监控指标 → 备案。

## 14. 结语

通过以上实践，可以将 HBase 主备复制的延迟稳定在秒级，并显著降低 WAL 队列积压风险。后续若进行版本升级或跨公有云部署，应再次对网络抖动、Kerberos 双向认证以及安全组策略做专项评估，以保持复制链路弹性与安全。