ApexDB
部署和运维手册（企业级）

2025/01

目录

软件使用和配置白皮书

说明

本文档主要针对RPM/DEB/TGZ等标准部署方式；若为其它方式请根据实际情况酌情处理

安装包获取

请在从同盾处获取最新安装包

注意，不同的CPU架构安装包是不一样的，目前我们支持x86_64和AAarch64两种架构

工具链获取

若需要使用asadm/apexdb-ql/asinfo等工具则请安装apexdbtools和tmc可视化控制台

这两个包无依赖关系，各自为全集成包，解压即用，同时支持x86和aarch64架构，aarch64架构下无法使用aql，请换用公司独有的apexdb-ql替代，已经包含在上面的软件包中

APEXDB-QL操作命令跟原AQL完全一样且优化了较多东西，该版本采用内部开源方式，欢迎加入共建

激活码获取

ApexDB需要激活码方可使用，请联系同盾购买License

操作系统参数

若用户采用TGZ的非ROOT用户模式安装启动，请事先优化下列参数，否则可能会有性能上的损失，对于标准的RPM方式则请忽略这一步，因为RPM/DEB始终都是ROOT模式

```
# 请事先执行下列命令确保ulimit -n的值大于100000，若不满足请提前修改
ulimit -n 100000

# 网卡参数优化，必须ROOT执行
echo 15728640 > /proc/sys/net/core/rmem_max
echo 5242880 > /proc/sys/net/core/wmem_max

# 禁用THP
echo never > /sys/kernel/mm/transparent_hugepage/enabled
```

安装方法

对于CentOS/RedHat系的操作系统（包括银河麒麟），始终建议使用RPM方式安装，如下：

```
sudo rpm -i apexdb-server-pro-3.x-tdxz.x86_64.rpm

# 安装完后ApexDB将以Systemd Service形式存在
# 请参考下面的配置规范修改配置文件后再启动服务，主进程名称为apxd

# 服务启动
sudo service apexdb start
# 服务状态查看
sudo service apexdb status
# 服务日志查看
sudo journalctl -u apexdb | tail
# 服务关停
sudo service apexdb stop
```

对于Debian/Ubuntu系的则建议使用DEB包安装，如下：

```
sudo dpkg -i apexdb-server-pro-3.x-tdxz.x86_64.deb
```

对于其它Linux系统（包括RedHat/Ubuntu在内），可使用TGZ方式安装，如下：

配置规范

我们构建了一个完整的配置手册文档即“APEXDB配置手册”，所有配置请以该文档为准，切记！

ApexDB只有一个配置文件，路径如下：

- RPM / DEB安装：/etc/apexdb/apexdb.conf
- TGZ安装：/安装目录/etc/apexdb.conf

无论是哪种安装方式，其中的配置项都是一致的，示例如下：

```
network {
    service {
        address any
        port 3000
    }

    heartbeat {
        mode mesh
        port 3002
        interval 150
        timeout 10
        # 种子节点一行一个，同集群中所有节点的种子须完全一致
        # 数量不要超过3个
        mesh-seed-address-port 10.10.10.10 3002
        mesh-seed-address-port 10.10.10.11 3002
    }

    fabric {
        port 3001
    }

    info {
        port 3003
    }
}
```

```
namespace ns1 {
    # 副本数，测试环境可设置为1，生产环境不得低于2
    replication-factor 2
    # 主索引内存大小，理论最大值256G
    memory-size 96G
    # 默认TTL
    default-ttl 30d
    # 生产环境建议设置为256，非生产环境无需配置
    partition-tree-locks 256
    # 生产环境建议设置为4096，非生产环境无需配置
    partition-tree-sprigs 4096

    # 以下三种存储模式三选一
    # 测试和POC场景可用模式3，生产环境必须用1或2

    # 存储模式1，块设备模式
    # 重要：如果盘此前安装过ApexDB
    # 请提前对sdb设备执行blkdiscard操作！！！
```

```
    # 存储模式2，文件模式
    storage-engine device {
        # 使用文件系统而不是裸设备
        # 请确保该路径存在且拥有可读写权限
        file /data01/as/data.bin
        # 文件大小限制，配置多少就会使用多少
        filesize 1G
        # 最小可用wblock数量，注意不是最小空间
        min-avail-pct 10
        # 写后读缓存队列长度，默认64，越长消耗的内存越多
        # 计算方式：长度 * write-block-size
        post-write-queue 64
        data-in-memory false
    }

    # 存储模式3，纯内存模式
    storage-engine memory
}
```

高级配置

注意，以下配置为高级功能，一般不需要设置，如果一定要设置请咨询管理员

```
service {
    # 该值决定了最大并发连接数，受限于ulimit中的nofile配置
    proto-fd-max 2048
    # 最大service线程数，默认是CPU核数的5倍
    service-threads 32
    ...
}

namespace test {
    ...
    # 最大磁盘占用比例（wblock维度），超过该比例后将进入高水位状态
    high-water-disk-pct 60
```

运行时监控

可使用APEXDB-MC工具进行可视化监控，或使用命令行工具“asadm”，参考上面的<运维工具>章节

生产环境建议通过prometheus+grafana采集监控项，请提前安装apexdb-exporter

重点关注指标

以下指标均可通过APEXDB-MC可视化控制台获取，也可通过prometheus采集

总数据条数、客户端错误请求数，单机条数不得超过42亿；如果错误数持续在增加应对日志进行分析以排查原因，一般正常情况下错误均为0

内存使用空间、剩余百分比，当剩余空间百分比低于35%时应进行扩容

硬盘使用空间、剩余百分比，当剩余空间百分比低于40%时应进行扩容

硬盘Avail指数，当该指数低于30%时应调高碎片整理阈值，参考配置手册中的“defrag-lwm-pct“配置项，每次调整幅度不能超过5%，最大值不能超过70

机器CPU使用率，当CPU使用率高于 核数*70 时，考虑增加节点

网卡流量，一般不得高于网卡总带宽的80%

LUA脚本注册

一般请使用APEXDB-MC以通过浏览器操作，也可使用apexdb-ql命令进行LUA脚本注册

例如：

```
apexdb-ql -h <IP> -c "register module '/xxx/xxx.lua'"
```

注意，IP填集群任意节点IP，集群会自动将LUA同步至所有节点

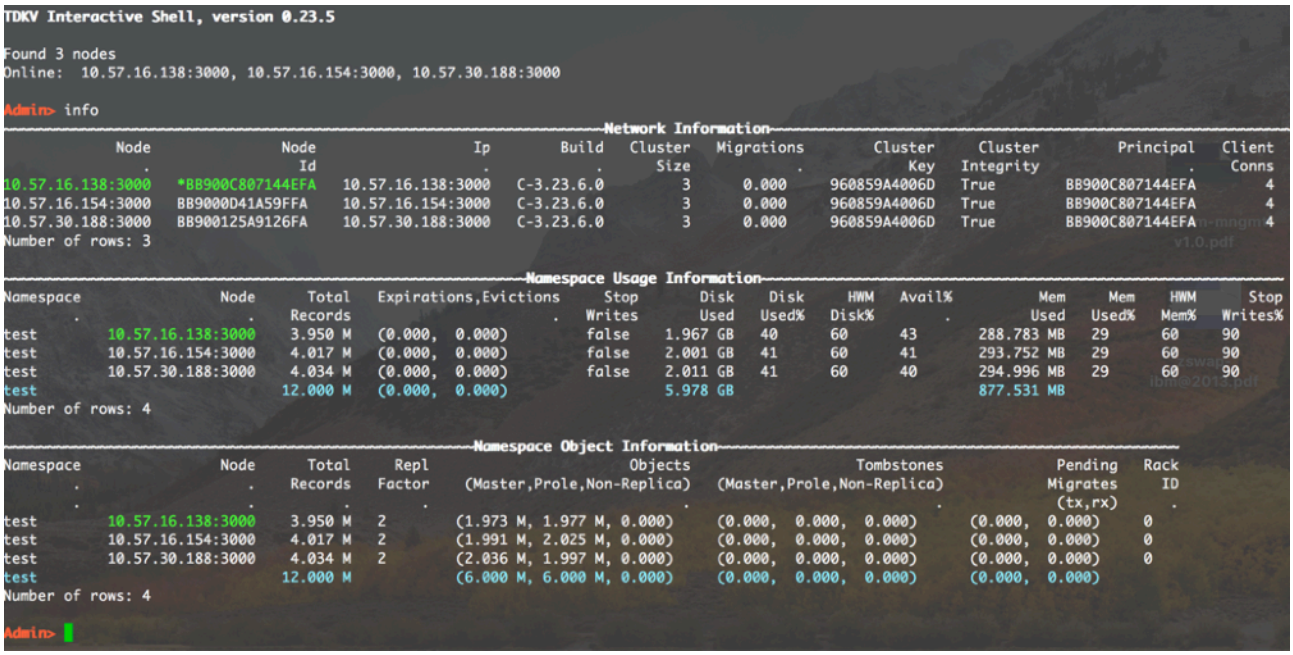
集群构建

在生产环境中请始终使用集群模式，最小3台机器，应尽量保持集群中的机器硬件配置统一

数据库集群节点通过心跳报文感知到彼此的存在，我们推荐使用mesh心跳模式（参考上面的配置文件）

同一个集群中的节点请务必保持种子节点列表完全一致；对于不同集群，绝对不能配置相同的种子节点

使用asadm工具的info命令可查看当前集群状态，例如



集群高可用

数据库固定拥有4096个分区，每个分区拥有一个master和至少1个replica
集群节点之间通过心跳和Paxos算法协商出统一的分区分配表，结构如下：<分区号，master，replica-1，replica-2...>

请确保namespace下的replication-factor配置（即副本数）不低于2，此时每个写入请求都会保证写入2台机器

副本原理概述：客户端向该分区的master节点发送写入请求，该节点完成该请求后会向replica节点发送副本写入请求

当replica节点也写入成功后，master节点才会向客户端发送响应成功

节点宕机

默认情况下每两个节点之间的心跳报文发送间隔是150ms，如果连续10次心跳报文无响应，则认为目标节点失效

当集群某台机器宕机时，集群所有节点的心跳机制将感知到这一事件，然后自动重新触发Paxos算法选举

以构建出新的分区分配表，客户端随后很短时间内也会收到最新的分配表并开始执行新的请求路径
同时，集群会开启一轮数据自动平衡，节点之间会以分区为单位复制数据，直到所有分区均满足副本数要求

宕机恢复

一般情况下无需任何特殊操作，数据将自动平衡；

集群灾备

目前我们有从0到1自研的apex-cp中间件，全称apexdb-cluster-copy，它基于业界最新的GraalVM 21 Native实现，支持快速将集群数据实时复制到其它集群，平均延迟2分钟以内，首次全量同步会消耗较长时间；该中间件支持任意断点续传，即使异常退出也只需重启即可，且对于单条数据是全量幂等写入；此外，支持抽样对比，抽样比例固定为5/10000

该中间件为一个独立运行的命令行程序，且为全静态链接，无需安装任何其它依赖，无需sudo权限，程序占用内存低，只需要1G内存即可实现10万以上的TPS。

具体使用方式可联系我们。

水位评估

管理员应持续关注集群运行状态和当前容量，主要关注点有下面几个：

Mem Used%，确保不高于65（集群规模大于3）或50（集群规模等于3）
Disk Used%，确保不高于60

Avail%，确保不低于30

若以上状态不满足则需要扩容

容量评估

每条数据的每个副本需占用64字节的纯内存空间用于主索引对象
例如1亿条数据2个副本情况下需占用12G内存空间

若集群为内存模式，则还需要加上数据本身的大小

若集群为硬盘模式，每条数据的每个副本在硬盘上额外需要128字节的Header，假设单条数据长度为1000字节，则在硬盘上实际存储长度为1152（逻辑长度为1128字节，每条数据需要按128字节进行对齐，因此额外多出24字节）

集群扩容

一般来说我们建议进行水平扩容，即加节点方式，这种方式方便快捷，集群中已有的机器无需重启，也不会影响正在进行的请求。

执行步骤参考：

1. 确认硬件配置合理性，并对分配给APEXDB的硬盘进行dd或blkdiscard清空操作（仅块设备模式）
    - 若该机器之前已经安装过APEXDB且使用了块设备模式，该操作将尤为重要，也可以用tools包里的PFDF工具进行超快速格式化
    - 如果是普通文件模式或内存模式则不需要该操作
2. 编辑配置文件，保持跟集群其它节点配置文件统一
3. 启动服务，此时APEXDB集群会自动进行数据均衡，期间服务保持正常，无需人工干预
4. 通过asadm的info命令查看集群数据迁移状态，也就是Pending Migraties项，当所有节点该项值均为0时表示迁移完成

注意，如果需要同时扩容多台，可将这些新节点同时拉起，不需要一台台操作

但如果一定要执行垂直扩容则需要对集群中每一台机器进行重启，可提前将这些机器的配置文件进行统一修改，因为配置文件一律都是重启后才生效。

集群重启

某些时候可能需要对集群中的机器执行轮转重启，例如垂直扩容、硬件升级等，此时建议开启APEXDB独有的增量迁移功能

在增量迁移功能启用时，集群间的节点将仅迁移有变化的数据，极大了缩短了迁移时间（相比全量迁移平均缩短时间100倍以上）
若不开启增量迁移，则整个集群全部重启可能会需要几天数据

注意事项：

原始开源版本不具备该功能，因此重启需要较长时间

必须等前一台节点彻底重启并平衡完成后（Pending Migrates归零）才能进行下一台重启

详细操作步骤请参考下面的《增量迁移功能白皮书》

TOOLS工具链

为了更好支撑APEXDB上下游生态的发展，提升日常管理和运维效率并降低维护成本，APEXDB在扩展了很多新的工具，且针对国产环境做了重点关照，它们将是研发和运维的有力助手，我们建议使用统一的tools工具链

组成部分

- APEXDB-MC
- APEXDB-QL
- ASADM
- PFDF
- APEXDB-CP

这些工具均在公司内开源

工具链详细列表

名称	用途	架构	定制	所属包
apexdb-ql	类似aql，用于数据DDL/DML	x64+arm64	独有	apexdbtools
asadm	命令行管理控制台	无	是	apexdbtools
asinfo	原生info协议交互程序	无	否	apexdbtools
pfdf	超快速硬盘格式化	x64+arm64	独有	apexdbtools
apexdb-backup apexdb-restore	备份和恢复	x64	否	apexdbtools
apexdb-mc	强大的可视化Web控制台 基于原Python版AMC重构而来	无	是	独立分发
apexdb-cp	集群间实时复制中间件，可实现灾备、双活等	x64	独有	独立分发

APEXDB-MC

可视化综合控制台，可在浏览器端查看集群运行状态
基于原Python版AMC重构而来，去掉了原AMC中对Eventlet等诸多外部框架的依赖，对企业级实施环境较为友好

功能和优点：

- 同时兼容Python2和Python3，支持各种国产化环境
- 支持连接带服务端认证的集群（公司独有功能）

该版本只需要解压即可使用，无需安装任何其它依赖，特别好
使用方式：解压到任意目录，执行"./bin/tmc-all.sh start”即可启动，控制台默认Web端口为8081

APEXDB-QL

提供x64和arm64两个版本，对应可执行文件分别是./bin/apexdb-ql.x86和./bin/apexdb-ql.aarch64

我们也提供了一个bash脚本的快捷方式，用户可直接执行./bin/apexdb-ql，该快捷方式会自动探测当前系统环境并运行对应程序，使用方式、参数等与原AQL基本一致（-h参数可查看帮助）

```
ApexDB QuickLook 1.1.9 (202501)
usage:
     -u show usage
     -h, --hosts <host1>[:<port1>],...,default: 127.0.0.1
     -t, --timeout <time in ms>    default: 1000
     -c <command string>      execute command then exits
     -p, --port <port> default: 3000
     -U, --user <name>
     -P, --password <password>
     --donut                  draw a living donut
     --xmem-dump <target-dir> dump xmem to an empty dir
     --xmem-restore <target-dir>   restore xmem from dir
     --xmem-view <base-file>       show metadata of xmem base
     --xmem-id <id>           xmem id default: 100
     --xmem-size <size-in-byte>    default: 1073741824(1GB)
```

常用命令和功能描述

命令	功能

show help	查看帮助
show namespaces	查询当前所有的namespace
show sets	查询当前所有的set
set durable_delete <true/false>	对当前会话启用或禁用持久化删除
select * from <namespace>.<set> where pk = <key>	查询单条数据，注意，不支持其它写法
select * from <namespace>.<set>	查询当前set中所有数据，默认最大展示1000条，勿对大表执行该操作
delete from <namespace>.<set> where pk = <key>	删除单条数据
truncate <namespace>.<set>	将整个set中的数据清空，慎重操作
show modules	查询当前集群已经注册的UDF模块
desc module <udf-file-name>	查询已注册的UDF模块代码（仅限Lua-UDF）
register module <udf-file-full-path>	注册UDF模块，请填入完整的模块文件路径
remove module <udf-file-name>	删除UDF模块

ASADM

命令行管理工具，可动态查询服务端状态、各类监控指标和更改服务端配置
相比原始官方版本的asadm，我们改写了较多逻辑，使之同时兼容python2和python3，对各类操作系统均良好兼容，此外还去除python-bcrypt强依赖

```
[admin@cxx-builder ~]$ asadm -u
usage: asadm [-h HOST] [-p PORT] [-U USER] [-P [PASSWORD]] [-c]
             [-l] [-e EXECUTE] [-o OUT_FILE] [--no-color] [-u] [--version]
             [-s] [-a] [--single_node_cluster] [--timeout TIMEOUT]
             [-f LOG_PATH]
```

常用命令和功能描述

命令	功能
info	查询集群和各节点基本数据
summary	查询当前集群总结
show latency	查询当前读写TPS和延迟
show distribution	查询数据TTL和长度的直方统计
show pmap	查看当前分区分配情况

PFDF

即Pretty Fast Disk Formatter，以极快速度格式化已经由ApexDB使用过的硬盘，避免旧数据重新扫描加载，该工具解决了传统linux dd执行时间过长（可能长达几个小时）问题，对于普通SATA-SSD而言只需几十秒秒即可完成

```
[admin@cxx-builder tools]$ ./pfdf

Pretty Fast Disk Formatter 1.0.1
only for ApexDB. max 16 devices supported
usage:
      -d, --devices /dev/sda,/dev/sdb,...
      -r, --readonly          DO NOT format, just print device info
      -e, --erase-header      erase ssd header as well
      -w, --write-block-size  override with this write-block-size
```

其中，-r和-e不可同时出现，而-w则可强制指定新的write-block-size，单位为字节，若需要更换新的write-block-size配置，请务必加上该参数，否则工具将按之前老的write-block-size值进行格式化

执行性能参考：HPE SATA SSD 440G，write-block-size: 256K，全盘格式化总共约40秒

持久化删除功能白皮书

说明

实现可靠的持久化删除功能，确保删除的数据在重启后不再重现出现
最低ApexDB版本要求：v3.23.5.19

适用场景：删除操作较少（例如每天删除1万次）的场景

配置规范

服务端在配置文件service小节中增加下列配置，请确保相关路径的读写权限

```
durable-delete-log "持久化删除元数据文件路径，例如/data01/ddl.bin"
durable-delete-size "持久化删除元数据文件最大长度，例如10M/1G，最小1K，最大16G"
```

注意，每次持久化删除操作会在元数据文件中存储长度固定为24字节的数据
若管理员未配置“durable-delete-size”，则系统默认值为64M

当文件长度到达最大长度后会从文件头循环覆盖因此绝不会超过最大长度，快速估算表：

持久删除操作数量	建议长度
50万	12M
100万	25M
200万	50M
1000万	250M

由于DDL文件写入量很小，对所在硬盘IO要求不高，普通机械盘即可

客户端启用

这里复用了开源标准协议中的DurableDelete功能开关，因此客户端无需升级即可使用

```
WritePolicy writePolicy = new WritePolicy(defaultReadPolicy可继承其他的
writePolicy的设置比较方便);
writePolicy.durableDelete=true; // 启用持久化删除
// 调用api
// 注意！！！这个writePolicy不要跟其它方法（例如set方法）共用，否则会报错，仅能用于
```

以Java客户端为例，WritePolicy#durableDelete设置为true

然后执行标准的delete方法并传入该Policy；若不开启该开关则仍然会执行普通delete操作

注意，业务方应谨慎评估需求，只对有必要的数据执行持久化删除

在QL工具中，可执行"set durable_delete true" 启用持久化删除

注意，若服务端未启用持久化删除但客户端仍然传入了该标记则客户端会收到错误码16（UNSUPPORTED_FEATURE）

服务端工作原理

启动时的全盘扫描结束后，开始顺序遍历DDL文件并过滤掉已经删除的记录（若存在），结束后才会提供服务

DDL遍历性能评估：最小25MB/s，按照最大16G来计算，最长需要约650秒时间，服务端仅需要一次遍历即可完成持久化删除保持逻辑，不会有任何IO压力

日志中会打出下列结果：

```
WARNING (rw): (delete_ce.c:168) ns:{test} durable-delete drops:3083148,
omits:1285870
```

管理员可在启动前删除DDL文件，此时持久化删除功能将不起作用，启动后系统将创建一个新的空白文件

持久化索引功能白皮书

说明

ApexDB在重启时需要进行全盘扫描以在内存中重新构建主索引树，根据盘的容量和盘数量而定，整个扫描过程可能会长达一到两个小时。
对于很多业务尤其是需要快速响应快速迭代的业务，这个缺点带来了诸多不便和运维上的挑战。在新版本中我们已经从0到1实现类似持久化索引功能，索引数据将保存在共享内存（SysV Shared Memory）中，重启时不再需要全盘扫描，从而降低运维管理成本，提升交付效率，提升公司整体技术先进性。

最低版本

3.23.9.x

功能描述

功能代号：PPI（Persistent Primary Index）
启用此功能后ApexDB主索引数据将存放于Linux共享内存段，即标准的System V shared memory（SHM），当ApexDB主进程重启时会重新加载并attach这些共享段到相同的地址，从而避免全盘扫描，整个过程只需要1秒即可完成

SHM是非常成熟可靠的机制，在Oracle Database和IBM DB2中也有广泛使用

跟其它类似软件的实现方式相比，我们额外增加了进程停机时主索引自动快照功能。下次启动时若SHM已经丢失（例如机器已重启）则会自动从快照中恢复主索引，无需人工干预从而有效降低管理负担

技术细节

每个Namespace均有自己独立的共享段区间，起始区间是[0x100, 0x1FF]（只有一个namespace时）；

每个段固定长度1GB，与主索引Stage长度精确对应。管理员可使用ipcs命令查看当前机器上的共享段，例如：

```
[admin@cxx-builder ~]$ ipcs -m

------ Shared Memory Segments --------
key        shmid      owner      perms      bytes
nattch     status
0x00000100 4685824    admin      666        1073741824
```

此外还有一些元数据（100M以内）会在正常停机时写入到work-directory目录的某个文件中，一般无需关心

配置启用

在namespace小节中加入如下配置：

```
# 必选，启用持久化模式的主索引
index-mode xmem

# 可选，若只有一个namespace则此配置可省略，最小值100最大值131，每个namespace必须独立
xmem-id 100

# 可选，启用停机时自动备份，请确保目标空间充足（最小值不小于memory-
```

运行过程中与常规索引模式的集群无任何差异

注意，若该namespace为memory存储引擎则会报错并启动失败

热重启

当持久化索引启用后，管理员可对ApexDB主进程执行“热重启”，此时，执行正常kill即可，注意不能执行kill -9，否则系统将认为进程异常退出，自动退回冷重启

日志中可以看到如下关键信息：

```
Aug 24 2023 08:34:15 GMT: WARNING (namespace):
(namespace_ce.c:337) {test} auto generated xmem-id: 100
Aug 24 2023 08:34:15 GMT: INFO (namespace):
```

机器重启

共享段在机器重启时仍然会丢失，若要解决这个问题，有两种方式：

方式1：使用自动快照和恢复（推荐）

即在配置文件中启用”xmem-autodump”，此后正常停机时系统会将主索引dump至该目录中，性能与APEXDB-QL一致，下次启动时若探测发现共享段数据不存在则会从dump文件中自动恢复

请注意，这种方式会显式增加停机时间，特别是大规模集群更是如此
好处是这种方式几乎无需人工干预，即使停机后发生了机器重启也不影响

有的时候只是需要重启一下进程，例如原地升级版本，此时可临时关闭自动快照，通过asinfo命令执行：

```
asinfo -v "set-
```

此时停机时不会向硬盘写入任何快照数据

方式2：可使用APEXDB-QL工具的主索引快照和恢复功能（备选）

APEXDB-QL最低版本：1.1.0

```
# 注意，不支持远程操作，必须在本地执行

# 将当前机器上的主索引段dump到dir目录中，请事先确保dir目录为空
# 硬盘空间要求与当前ApexDB节点使用的内存一致
# 性能参考：0.8GB/s（SSD盘），0.15~0.3GB/s（机械盘）
# 执行后会在目录中生成多个快照文件，请勿修改文件名
# 该命令只能在ApexDB主进程停机后才能操作，否则会提示错误
apexdb-ql --xmem-dump <dir>

# 在ApexDB启动前，将快照恢复至shm，此时性能较好，大约是dump的2倍
apexdb-ql --xmem-restore <dir>
```

FAQ

- 持久化主索引模式和普通模式性能有无差别
  - 没有任何差别
- 如何强制冷重启，即忽略共享段数据并执行老的全盘扫描
  - 方式1：在启动命令参数中加入--cold-start
  - 方式2：删除work-directory中以shm开头的元数据文件
- 手动快照和自动快照是否有冲突
  - 没有，两者工作方式和数据结构完全一致且生成的文件也完全一致
- SHM或快照数据损坏该怎么办
  - 请参考上述步骤执行强制冷重启
- 应该在什么场景开启这个功能
  - 推荐所有生产集群都使用，原来的ram索引逐步淡化退出

质量保障

配置	描述
环境	线下
集群规模	3节点
单机配置	单机内存4G，硬盘20G，启动自动快照和增量迁移
数据量	约5500万条数据

15项集成测试	通过（含x86和AArch64）
多负载测试	通过（含x86和AArch64），增删改查
随机节点重启	平均间隔5分钟，已触发1500次以上，期间多负载测试持续不间断运行 所有重启均成功，增量迁移正常，数据总量无误

增量迁移功能白皮书

说明

简单增量迁移（Naïve Delta Migration，缩写NDM）旨在消除ApexDB节点重启期间发生的不必要的Migration，从而大大降低重启时间和资源消耗，解决长期以来困扰我们的重启后数据平衡时间过长、读写压力过大的问题

类似的开源软件中提供了相似的功能特性，而ApexDB的NDM算法的实现方式则完全不同，从原理来看NDM性能更高更快，但对使用场景有一定要求

NDM功能禁用

在v3.23.5.8或更高版本中，NDM默认处于禁用状态，管理员可在配置文件中启用，相关配置项为”activate-ndm"（service小节），可取值：never/yes/no
详情请参考配置手册文档

NDM功能启用

ApexDB最低版本：3.23.4.1

默认情况下ApexDB并不会启用NDM，管理员可手动开启NDM，或等待系统启发式自动开启，一旦开启后系统将会在所配置的work-directory目录下生成一个名为”ndm-meta”的元数据文件用于记录集群基线

注意：在增量迁移执行过程中不可修改集群中任何机器的时间

手动开启

管理员在集群结构稳定后显式启用NDM机制，若进行永久缩容需提前关闭NDM，通过asinfo命令执行：

```
#注意，集群中每台机器都需要单独执行

#开启
asinfo -h $IP -v 'set-config:context=service;activate-ndm=ye
#关闭
asinfo -h $IP -v 'set-config:context=service;activate-ndm=nc
#关闭且禁用自动开启
asinfo -h $IP -v 'set-config:context=service;activate-ndm=ne
#查看NDM是否开启，1表示开启，0表示关闭，-1表示已禁用
asinfo -h $IP -v 'get-config:context=service;' | sed 's/;/\n
#设置NDM超时时间，单位小时，默认24小时
asinfo -h $IP -v 'set-config:context=service;ndm-timeout-hou
```

自动开启

为了减轻管理员操作负担，本算法还会进行启发式智能判断，当以下条件达成时自动开启NDM：

- NDM尚未开启且未禁用
- 集群当前无任何尚未完成或正在进行的迁移任务
- 距离上一次集群拓扑结构变化已超过12小时

执行流程

当NDM开启时，系统将以当前这一时刻的集群节点组成作为基线
整体原则：基线内的节点上下线时进行增量迁移，一旦出现基线外的节点则全部节点退回全量迁移

基线节点下线

此时不会发生任何数据迁移，但仍然可从监控上看到迁移任务即Pending Migration，这些任务将在1到2分钟内迅速结束

管理员务必等待至少2分钟后再重新上线旧节点

基线节点重新上线

此时其它节点会将期间有更新的数据传输到重新上线的节点，监控上会看到新一轮Pending Migration任务，这些任务执行时间视期间变化数据量而定，若期间没有任何数据写入则同样会在几分钟内结束

注意，若重新上线的基线节点距离上次启动已超过”ndm-timeout-hour"小时（包括硬盘全盘扫描时间在内）则会退回全量迁移

基线节点磁盘数据变化

若基线节点磁盘数据与下线时相比发生了任何变化，例如更换了其中某一块磁盘，则需要进行全量迁移，管理员可以采取下面两种选择之一：

1. 在该节点上线前手动关闭集群NDM
2. 等待2小时后再上线，无需其它操作

永久缩容

管理员需在节点下线前手动关闭NDM

永久扩容

一旦节点扩容上线后，NDM会再次自动开启（在满足上面条件时）以便将扩容的节点加入到基线中

集群整体关停

在某些时候可能需要将集群全体节点关机并在稍后重新启动，例如机架搬迁等场景，此时增量迁移仍然可以发挥作用

集群中的节点依次启动时不会发生任何数据迁移，此时监控上仍然会看到一闪即过的迁移任务，请确保该操作在”ndm-timeout-hour”小时内完成

- 若未能完成，管理员可对每个节点上的"ndm-meta"文件执行touch操作，否则系统将会退回全量迁移

NDM运行时指标

管理员可通过asinfo命令采集NDM的实际运行状态数据

```
asinfo -h $IP -v 'namespace/ns1' | sed 's/;/\n/g' | grep mig

#输出结果举例
migrate_records_skipped=3361051 #已跳过的记录数（即被NDM判断为无需
migrate_records_transmitted=0 #已实际传输的记录数
```

此外，还可以从日志中获取NDM运行信息，以ndm作为关键词搜索主日志文件，例如

```
Mar 24 2023 02:59:31 GMT: WARNING (partition): (partition_ba
Mar 24 2023 02:59:31 GMT: WARNING (migrate): (migrate_ce.c:1
Mar 24 2023 02:59:48 GMT: WARNING (partition): (partition_ba
```

当集群拓扑结构发生任何变化时会在日志中看到migrate_origin序号，其中：
0表示增量迁移不可用，例如水平扩缩容
1表示基线节点下线，紧跟着会有一条ts marked日志，代表该事件时间戳
2表示自身上线且自身也属于基线
4表示有基线节点上线但基线中仍然有节点尚未启动，这种情况仅仅只会发生在集群全体同时重启时候
6表示有基线节点上线且当前集群已经达到基线最终构成

迁移压制（3.24.9.0以上）

新增了迁移压制功能，在这段时间内分区的迁移将会瞬间完成，不会发生任何实质性数据扫描和传输动作，管理员可以动态配置压制时间，但注意需确保这段时间内集群没有发生任何数据写入，否则会存在副本丢失问题

如果当前有迁移任务正在进行，则在当前正在迁移的分区完成后，从下一个分区开始压制

适用场景如下

- 集群全体关停重启，例如集群搬迁等
- XPS模式下例行停机维护

使用方式

```
# 注意，无论当前集群是否启用增量迁移，都可以开启迁移压制
# 动态配置当前节点压制5分钟（单位为分钟，最大值1440，最小值1）
asinfo -h $IP -v 'set-config:context=service;suppress-mi
```