这就又回到之前的问题了，现在是通过执行如下命令 virsh qemu-monitor-command instance-00000885 --pretty '{"execute":"drive-mirror","arguments":{"job-id": "mirror0","device":"libvirt-1-format","target":"rbd:sata\_pool/volume-ec0e79b1-059e-4202-9fae-fdfe381786f1:auth\_supported=none:mon\_host=172.31.189.55\\:6789","sync":"full","format":"raw"}}' 对挂载给虚机的rbd卷进行热迁移，比如卷大小是10GB，使用了400MB，但是热迁移完成后，发现该卷磁盘占用大小就是10GB，等于说卷膨胀了，有一个detect\_zeroes参数可以进行传递，写法如下 target->detect\_zeroes = BLOCKDEV\_DETECT\_ZEROES\_OPTIONS\_ON; 当时这行代码是在qmeu-2.6.0版本里加的，但是我现在看到qemu版本是6.2.0，也没有针对detect\_zeroes参数做配置，我现在就是想通过这个参数来避免卷热迁移产生的膨胀问题

已深度思考（用时 37 秒）

好的，用户现在遇到了一个关于Ceph RBD卷热迁移后卷膨胀的问题。他们之前执行了`drive-mirror`命令进行热迁移，结果目标卷的大小和源卷一样，导致存储空间的浪费。用户了解到QEMU有一个`detect\_zeroes`参数，可以检测并处理零块，从而避免不必要的空间占用，特别是在迁移时。然而，他们发现在QEMU 6.2.0版本中并没有正确配置这个参数，导致迁移后的卷仍然膨胀。

首先，我需要回顾一下用户的历史对话，确保理解问题的来龙去脉。用户之前尝试通过`drive-mirror`命令进行热迁移，但目标卷占用了和源卷相同的空间，而源卷实际只使用了400MB。用户希望利用`detect\_zeroes`参数来优化迁移，避免空间膨胀。

接下来，我需要确认`detect\_zeroes`参数在QEMU中的作用。这个参数用于检测写入的零块，并将其标记为未使用，从而节省存储空间。可能的选项包括`off`（关闭）、`on`（检测但不释放空间）和`unmap`（检测并释放空间）。用户需要的是`unmap`选项，以便在迁移时，零块被识别并触发存储后端的空间回收，比如Ceph的`discard`操作。

然后，检查用户提供的命令，发现他们在`drive-mirror`命令中没有包含`detect-zeroes`参数。这可能就是问题所在。在QEMU 6.2.0中，虽然支持该参数，但需要显式配置才能生效。用户可能没有在命令中正确添加这个参数，或者没有在虚拟机配置中启用相关选项。

此外，用户提到在QEMU 2.6.0版本中有一行代码设置`detect\_zeroes`为`ON`，但在6.2.0版本中可能配置方式不同，或者需要结合其他参数（如`discard`）一起使用。需要确认QEMU 6.2.0的文档或源码，确保`detect-zeroes`和`discard`参数的用法。

还需要考虑Ceph RBD的配置是否支持`discard`操作。Ceph集群需要启用相应的特性（如`fast-diff`和`exclusive-lock`），并且镜像的配置要允许`discard`。用户可能需要检查Ceph端的设置，确保目标卷支持空间回收。

接下来，用户需要修改他们的热迁移命令，添加`detect-zeroes`和`discard`参数。例如，在`drive-mirror`命令中加入`"detect-zeroes": "unmap"`和`"discard": "unmap"`。同时，确保虚拟机的磁盘配置中也启用了这些选项，以便在迁移后继续保持优化。

此外，用户可能需要检查迁移后的卷是否真的应用了这些参数。可以通过查询块设备的状态或检查Ceph端的存储使用情况来验证。如果在迁移过程中仍有空间膨胀，可能需要进一步调试，比如查看QEMU日志，确认参数是否正确传递，或者检查Ceph是否正确处理了`discard`请求。

最后，建议用户进行测试，使用一个已知包含大量零块的卷进行迁移，观察目标卷的空间占用情况。如果参数配置正确，目标卷的空间使用应该接近实际数据量，而不是全部分配。如果问题依旧存在，可能需要深入分析QEMU和Ceph的交互日志，查找可能的错误或配置遗漏。

在 QEMU 中通过 **detect\_zeroes** 参数（尤其是 **unmap** 模式）结合 **discard=unmap** 可以有效避免热迁移后卷膨胀的问题。以下是完整解决方案：

### ****1. 根本原因分析****

**全量同步（sync: full）默认行为**：即使源卷存在大量零块（未使用空间），QEMU 会忠实地将全量数据（包括零块）写入目标卷，导致目标卷预分配全部容量（如 10GB）。

**detect\_zeroes=unmap 的作用**：检测写入的零块并触发 **discard** 操作，通知存储后端（如 Ceph RBD）释放对应块的空间。

### ****2. 解决方案步骤****

#### ****(1) 修改热迁移命令，添加**detect-zeroes**和**discard**参数****

bash

复制

下载

virsh qemu-monitor-command instance-00000885 --pretty '{

"execute": "drive-mirror",

"arguments": {

"job-id": "mirror0",

"device": "libvirt-1-format",

"target": "rbd:sata\_pool/volume-ec0e79b1-059e-4202-9fae-fdfe381786f1:auth\_supported=none:mon\_host=172.31.189.55\\:6789",

"sync": "full",

"format": "raw",

"detect-zeroes": "unmap", # 关键参数

"discard": "unmap" # 必须同时启用

}

}'

#### ****(2) 确保虚拟机磁盘配置启用**detect\_zeroes**和**discard**

在虚拟机 XML 的磁盘配置中添加：

xml

复制

下载

运行

<disk type='network' device='disk'>

<driver name='qemu' type='raw' detect\_zeroes='unmap' discard='unmap'/>

<source protocol='rbd' name='sata\_pool/volume-...'/>

<target dev='vda' bus='virtio'/></disk>

#### ****(3) 验证 Ceph RBD 镜像支持**discard**

bash

复制

下载

*# 检查 RBD 镜像特性是否包含 fast-diff 和 exclusive-lock*

rbd info sata\_pool/volume-ec0e79b1-059e-4202-9fae-fdfe381786f1

*# 若未启用，添加特性*

rbd feature enable sata\_pool/volume-ec0e79b1-059e-4202-9fae-fdfe381786f1 fast-diff exclusive-lock

*# 确认 Ceph 集群配置允许 discard*

ceph config set osd osd\_op\_queue mclock\_scheduler *# 确保调度器支持 discard*

#### ****(4) 在客户机内启用 TRIM 支持****

bash

复制

下载

*# 在虚拟机内部执行*sudo fstrim -av *# 手动触发 TRIM# 或修改 /etc/fstab 添加 discard 选项*

/dev/vda1 / ext4 defaults,discard 0 0

### ****3. QEMU 6.2.0 关键代码逻辑****

在 QEMU 6.2.0 中，**detect-zeroes** 和 **discard** 参数的处理位于以下位置：

#### ****(1) 块设备配置解析****

**代码文件**：**blockdev.c**

**逻辑**：将 **detect-zeroes** 和 **discard** 参数传递给块设备驱动：

c

复制

下载

if (detect\_zeroes) {

bs->detect\_zeroes = detect\_zeroes;

if (detect\_zeroes == BLOCKDEV\_DETECT\_ZEROES\_OPTIONS\_UNMAP) {

bs->supported\_zero\_flags |= BDRV\_REQ\_MAY\_UNMAP;

}}

#### ****(2) RBD 驱动的**discard**支持****

**代码文件**：**block/rbd.c**

**逻辑**：当收到 **discard** 请求时，调用 **rbd\_discard**：

c

复制

下载

.bdrv\_co\_pdiscard = qemu\_rbd\_co\_pdiscard,

#### ****(3) 零块检测与处理****

**代码文件**：**block/io.c**

**逻辑**：在 **bdrv\_co\_writev** 中检测零块并触发 **discard**：

c

复制

下载

if (flags & BDRV\_REQ\_MAY\_UNMAP) {

bdrv\_co\_pdiscard(bs, offset, bytes);}

### ****4. 验证迁移结果****

#### ****(1) 检查目标卷空间占用****

bash

复制

下载

rbd du sata\_pool/volume-ec0e79b1-059e-4202-9fae-fdfe381786f1*# 输出示例：PROVISIONED=10G, USED=400M*

#### ****(2) 查看 QEMU 块设备状态****

bash

复制

下载

virsh qemu-monitor-command instance-00000885 --pretty '{"execute":"query-block"}'*# 检查输出中的 detect-zeroes 和 discard 字段*

### ****5. 高级调试方法****

#### ****(1) 启用 QEMU 调试日志****

bash

复制

下载

qemu-system-x86\_64 -D /var/log/qemu.log \

-chardev socket,id=monitor,path=/var/lib/libvirt/qemu/instance-00000885.monitor \

-monitor chardev=monitor \

-trace 'bdrv\_co\_writev,bdrv\_co\_pdiscard'

#### ****(2) 检查 Ceph 集群 OSD 日志****

bash

复制

下载

tail -f /var/log/ceph/ceph-osd.\*.log | grep 'discard'

### ****6. 总结****

| **步骤** | **操作** | **验证命令** |
| --- | --- | --- |
| 修改热迁移命令 | 添加 **detect-zeroes="unmap"** 和 **discard="unmap"** | **virsh qemu-monitor-command** |
| 配置虚拟机磁盘参数 | XML 中设置 **detect\_zeroes** 和 **discard** | **virsh dumpxml <domain>** |
| 启用 Ceph RBD 特性 | 添加 **fast-diff** 和 **exclusive-lock** | **rbd info <pool>/<image>** |
| 客户机启用 TRIM | 修改 **/etc/fstab** 或定期执行 **fstrim** | **lsblk --discard** |
| 验证存储空间 | 检查 RBD 卷的 **USED** 值 | **rbd du <pool>/<image>** |

通过上述配置，热迁移时零块将被检测并触发 **discard**，Ceph RBD 会回收未使用的块，从而避免卷膨胀。若仍存在空间问题，需检查 QEMU 版本兼容性及 Ceph 集群的 **discard** 支持状态。