

## 目 录

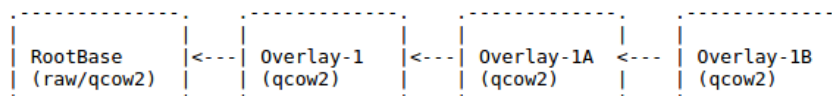
<b>1</b>	<b>QCOW2 backing files 和 overlays</b>	<b>2</b>
1.1	举个例子 . . . . .	2
1.1.1	创建增量镜像的命令 . . . . .	2
1.1.2	查看后端镜像的命令 . . . . .	3
<b>2</b>	<b>快照相关术语</b>	<b>4</b>
2.1	内部快照 . . . . .	4
2.1.1	内部磁盘快照 . . . . .	4
2.1.2	内部系统还原点 . . . . .	4
2.2	外部快照 . . . . .	4
2.2.1	外部磁盘快照 . . . . .	4
2.2.2	外部系统还原点 . . . . .	4
<b>3</b>	<b>创建快照的命令</b>	<b>5</b>
3.1	创建 disk-only 的内部快照 . . . . .	5
3.2	创建 disk-only 的外部快照 . . . . .	5
<b>4</b>	<b>reverting to snapshots</b>	<b>6</b>
4.1	内部快照的回滚 . . . . .	6
4.2	外部快照的回滚 . . . . .	6
<b>5</b>	<b>增量镜像和后端镜像的合并</b>	<b>7</b>
5.1	blockcommit . . . . .	7
5.2	blockpull . . . . .	7
<b>6</b>	<b>删除快照</b>	<b>9</b>
6.1	删除内部快照 . . . . .	9
6.2	删除外部快照 . . . . .	9
6.2.1	方式一 . . . . .	9
6.2.2	方式二 . . . . .	9

# 1 QCOW2 backing files 和 overlays

backing file 的中文是后端镜像，overlay 的中文是增量镜像。

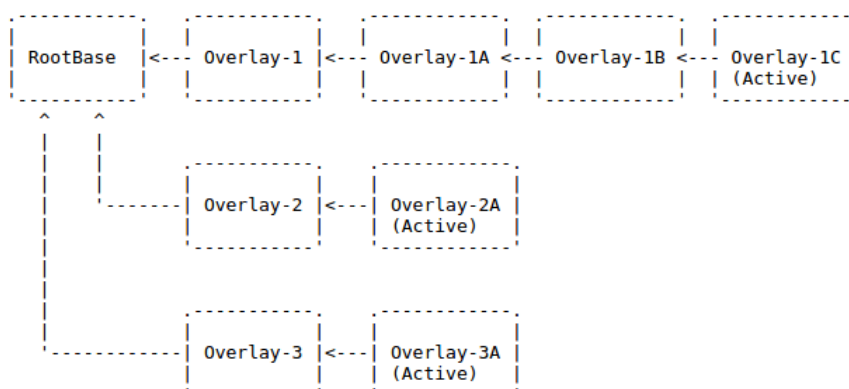
qemu 中经常使用 qcow2 镜像，因为它具有 copy-on-write 特性。这种特性支持：当我们创建虚拟机时，可以先创建一个 base-image，也就是 backing file，也就是后端镜像。后端镜像中包含有完整的 OS 以及引导程序。随后可以根据 base-image 为模板创建多个虚拟机，也就是在 backing file 的基础上创建多个 copy-on-write overlays 镜像。

不止如此，我们还可以在创建好的 overlay 镜像上再创建 overlay 镜像，如下图所示：



在上图中，overlay-1 的 backing file 是 RootBase，overlay-1A 的 backing file 是 overlay-1，overlay-1B 的 backing file 是 overlay-1A。

当然我们也可以在一個 base-image 上创建多个 overlays 镜像，如下图所示：



以下是关于 backing file 的注意事项：

- 1 一个 overlay 镜像创建以后，它的 backing file 是不允许被修改的，只能是 read-only。

## 1.1 举个例子

### 1.1.1 创建增量镜像的命令

现在 FedoraBase.img 中刚刚装好了 Fedora-17 操作系统，现在我们可以如下命令将它变成后端镜像，并且在它的基础上制作增量镜像：

```
1 qemu-img create -b FedoraBase.img -f qcow2 Fedora-guest-1.qcow2
```

有时候，一个虚拟机仅仅是刚刚装好操作系统是不够的，最好是还有完整的软件包更新，此时我们将 Fedora-guest-1.qcow2 中的虚拟机的软件包进行更新，然后在进行如下命令：

```
1 qemu-img create -b Fedora-guest-1.qcow2 -f qcow2 Fedora-guest-with-updates-1.qcow2
```

此时 Fedora-guest-1.qcow2 也变成了一个 read-only 文件，而 Fedora-guest-with-updates-1.qcow2 作为一个增量镜像，它里面已经装好了带有完整软件包更新的操作系统。

### 1.1.2 查看后端镜像的命令

可以通过如下命令查看虚拟机镜像的 backing file：

```
1 qemu-img info Fedora-guest-1.qcow2
```

可以通过如下命令查看虚拟机镜像的 backing chain：

```
1 qemu-img info --backing-chain Fedora-guest-with-updates-1.qcow2
```

## 2 快照相关术语

### 2.1 内部快照

#### 2.1.1 内部磁盘快照

这个将存储 snapshot 时虚拟机磁盘的状态。虚拟机在执行快照之后的 snapshot 和 delta 都存放在原来的 qcow2 文件中。

可以使用“qemu-img snapshot”命令对一台 offline 的虚拟机进行快照管理。关于它更多的描述可以查看《KVM 基础功能》中的“qemu-img 工具”一节。

也可以在 QEMU monitor 中的“savevm”命令对一台 online 的虚拟机进行快照创建。关于它更多的描述可以查看《KVM 高级功能》中的“QEMU 监控器”一节。

#### 2.1.2 内部系统还原点

系统还原点指的是 RAM state、device state 和 disk state。内部系统还原点存放在原来的 qcow2 镜像中。

需要知道的是，只有在运行状态才能创建内部系统还原点。内部系统还原点的建立也是使用“savevm”命令。

### 2.2 外部快照

当一个外部快照建立时，虚拟机被保存的状态将被存放在一个文件中，这个文件也就成为了 read-only 的后端镜像，而同时将产生一个新文件，作为一个增量镜像 overlay 来记录 snapshot 以后的 delta。

#### 2.2.1 外部磁盘快照

虚拟机磁盘信息的快照将被存放在一个文件中，然后 snapshot 之后的 delta 将被一个新的 qcow2 文件记录。

#### 2.2.2 外部系统还原点

虚拟机的磁盘状态将被保存到一个文件中，内存和设备的状态将被保存到另外一个新的文件中。

## 3 创建快照的命令

### 3.1 创建 disk-only 的内部快照

假设我们有一个虚拟机叫做“f17vm1”，可以通过如下命令创建一个内部快照，这个内部快照的名字叫做“snap1”，它的描述是“snap1-desc”：

```
1 virsh snapshot--create-as f17vm1 snap1 snap1-desc
```

可以通过命令查看镜像的快照信息：

```
1 # 方法一
2 virsh snapshot--list f17vm1
3 # 方法二
4 qemu-img info f17vm1.qcow2
```

### 3.2 创建 disk-only 的外部快照

可以使用 virsh 创建外部快照。假设我们有一个“devstack”的虚拟机，它的磁盘名为“hda”，原先的磁盘为“devstack.qcow2”。一旦创建快照以后，“devstack.qcow2”文件就会变成 backing file，而命令行中指定的“file”选项将作为 overlay 镜像，如下图所示：

```
virsh # domblklist devstack
目标      源
-----
hda       /home/pengsida/kvm/openstack/devstack.qcow2

virsh # snapshot-create-as devstack snap1 snap1-desc --disk-only --diskspec hda,
snapshot=external,file=/home/pengsida/kvm/openstack/overlay.qcow2 --atomic
已生成域快照 snap1
virsh # domblklist devstack
目标      源
-----
hda       /home/pengsida/kvm/openstack/overlay.qcow2
```

## 4 reverting to snapshots

### 4.1 内部快照的回滚

内部快照回滚很简单，如下所示：

```
1 virsh snapshot-revert --domain fl7vm1 snap1
```

### 4.2 外部快照的回滚

外部回滚需要将虚拟机的磁盘设备替换为某一版的外部快照文件：

```
1 # 查看 image chain  
2 qemu-img info --backing-chain active.qcow2
```

然后将图中的“file”选项的值修改为某一版的外部快照文件的路径：

```
<disk type='file' device='disk'>  
  <driver name='qemu' type='qcow2' />  
  <source file='/home/pengsida/kvm/openstack/overlay.qcow2' />  
  <target dev='hda' bus='ide' />  
  <address type='drive' controller='0' bus='0' target='0' unit='0' />  
</disk>
```

## 5 增量镜像和后端镜像的合并

增量镜像和后端镜像合并的两种方式：

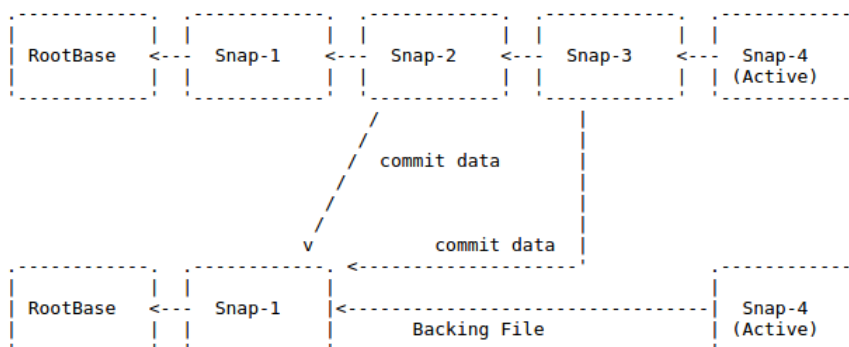
1. blockcommit：将增量镜像融入后端镜像中。
2. blockpull：将后端镜像融入增量镜像中。

### 5.1 blockcommit

blockcommit 将增量镜像融入后端镜像中。

假如存在镜像链  $\text{base} \leftarrow \text{overlay1} \leftarrow \text{overlay2} \leftarrow \text{overlay3}$ ，将  $\text{overlay2}$  融入  $\text{overlay1}$  以后，原先指向  $\text{overlay2}$  的  $\text{overlay3}$  镜像将直接指向  $\text{overlay1}$  镜像。

通过下图很容易理解上一段话：



实现上图中的过程很简单，只要先将  $\text{snap-3}$  融入  $\text{snap-2}$  中，再把  $\text{snap-2}$  融入  $\text{snap-1}$  中就可以：

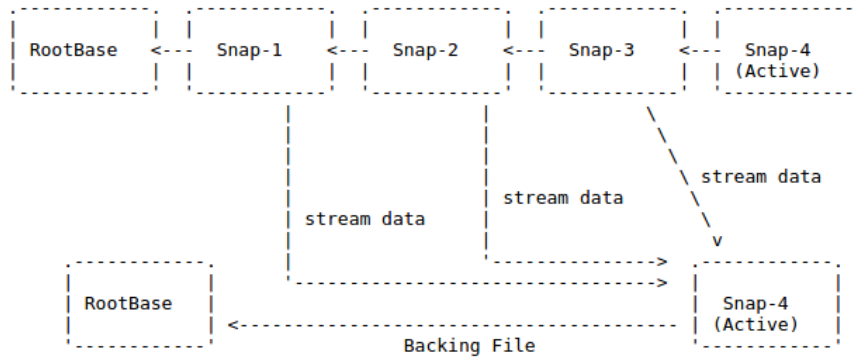
```
1 virsh blockcommit --domain f17 vda --base sn2.qcow2 --top sn3.qcow2 --wait --
  verbose
2 virsh blockcommit --domain f17 vda --base sn1.qcow2 --top sn2.qcow2 --wait --
  verbose
```

### 5.2 blockpull

blockpull 将后端镜像融入增量镜像中。需要注意的是，这里的增量镜像必须是 active 的，也就是镜像链末端的增量镜像。

假如存在镜像链  $\text{base} \leftarrow \text{overlay1} \leftarrow \text{overlay2} \leftarrow \text{overlay3}$ ，将  $\text{overlay1}$  融入  $\text{overlay2}$  以后，原先指向  $\text{overlay2}$  的  $\text{overlay3}$  镜像将直接指向  $\text{overlay1}$  镜像。

通过下图很容易理解上一段话：



可以通过如下命令实现上述操作：

```

1  # 将后端镜像融入增量镜像
2  virsh blockpull --domain RootBase --path var/lib/libvirt/images/active.qcow2 --
   base /var/lib/libvirt/images/RootBase.qcow2 --wait --verbose
3  # 清除原先快照的数据
4  virsh snapshot-delete --domain RootBase Snap-3 --metadata
5  virsh snapshot-delete --domain RootBase Snap-2 --metadata
6  virsh snapshot-delete --domain RootBase Snap-1 --metadata
  
```



## 6 删除快照

### 6.1 删除内部快照

删除内部快照比较简单，命令如下：

```
1 virsh snapshot-delete --domain f17vm --snapshotname snap6
```

### 6.2 删除外部快照

假设我们现在有一个 image chain: base<-sn1<-sn2<-sn3。现在我们想删除外部快照 sn2。

#### 6.2.1 方式一

将 sn2 融入 sn1 中，然后将 sn1 作为 sn3 的后端镜像：

```
1 qemu-img commit sn2.qcow2
2 # -u代表非安全模式，假设新的后端镜像和旧的后端镜像完全相同
3 qemu-img rebase -u -b sn1.qcow2 sn3.qcow2
4 # 最后删除外部快照sn2
5 rm sn2.qcow2
```

#### 6.2.2 方式二

使用安全模式将 sn1 作为 sn3 的后端镜像：

```
1 # 安全模式下，sn3原先的后端镜像sn2只要和sn1有一点不一样，都会被融入sn1中
2 qemu-img rebase -b sn1.qcow2 sn3.qcow2
3 # 最后删除外部快照sn2
4 rm sn2.qcow2
```