



链滴

docker 数据容器运行, 备份, 管理

作者: [Smiteli](#)

原文链接: <https://ld246.com/article/1577674239433>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

数据容器

常见的使用场景是使用纯数据容器来持久化数据库、配置文件或者数据文件等。[官方的文档](#)上有详细解释。例如：

```
$ docker run --name dbdata postgres echo "Data-only container for postgres"
```

该命令将会创建一个已经包含在Dockerfile里定义过Volume的postgres镜像，运行echo命令然后退。当我们运行docker ps命令时，echo可以帮助我们识别某镜像的用途。我们可以用-volumes-from命令来识别其它容器的Volume：

```
$ docker run -d --volumes-from dbdata --name db1 postgres
```

使用数据容器的两个注意点：

- 不要运行数据容器，这纯粹是在浪费资源。
- 不要为了数据容器而使用“最小的镜像”，如 busybox或scratch，只使用数据库镜像本身就可以。你已经拥有该镜像，所以并不需要占用额外的空间。

备份

如果你在用数据容器，那做备份是相当容易的：

```
$ docker run --rm --volumes-from dbdata -v $(pwd):/backup debian tar cvf /backup/backup.tar /var/lib/postgresql/data
```

该示例应该会将Volume里所有的东西压缩为一个tar包（官方的postgres Dockerfile在/var/lib/postgresql/data目录下定义了一个Volume）

权限与许可

通常你需要设置Volume的权限或者为Volume初始化一些默认数据或者配置文件。要注意的关键点，在Dockerfile的VOLUME指令后的任何东西都不能改变该Volume，比如：

```
FROM debian:wheezy
RUN useradd foo
VOLUME /data
RUN touch /data/x
RUN chown -R foo:foo /data
```

该Docker file不能按预期那样运行，我们本来希望touch命令在镜像的文件系统上运行，但是实际上是在一个临时容器的Volume上运行。如下所示：

```
FROM debian:wheezy
RUN useradd foo
RUN mkdir /data && touch /data/x
RUN chown -R foo:foo /data
VOLUME /data
```

Docker可以将镜像中Volume下的文件挂载到Volume下，并设置正确的权限。如果你指定Volume主机目录将不会出现这种情况。

如果你没有通过RUN指令设置权限，那么你就需要在容器启动时使用CMD或ENTRYPOINT指令来执行（译者注：CMD指令用于指定一个容器启动时要运行的命令，与RUN类似，只是RUN是镜像在构建要运行的命令）。

删除Volumes

这个功能可能会更加重要，如果你已经使用docker rm来删除你的容器，那可能有很多的孤立的Volume仍在占用着空间。

Volume只有在下列情况下才能被删除：

- 该容器是用 `docker rm -v`命令来删除的（-v是必不可少的）。
- `docker run`中使用了--rm参数

即使用以上两种命令，也只能删除没有容器连接的Volume。连接到用户指定主机目录的Volume永远不会被docker删除。

除非你已经很小心的，总是像这样来运行容器，否则你将会在/var/lib/docker/vfs/dir目录下得到一僵尸文件和目录，并且还不容易说出它们到底代表什么。