**Docker数据卷**

数据卷是一个可供一个或多个容器使用的特殊目录，它绕过 UFS，可以提供很多有用的特性：

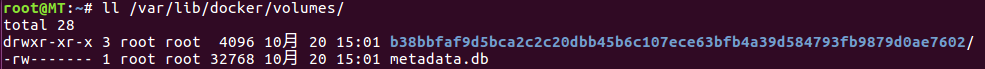
数据卷可以在容器之间共享和重用

对数据卷的修改会立马生效

对数据卷的更新，不会影响镜像

数据卷默认会一直存在，即使容器被删除

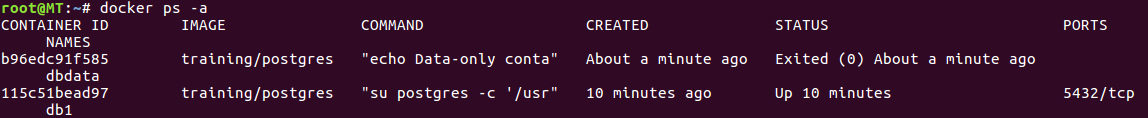
sudo docker run -d -P --name web -v /webapp training/webapp python app.py



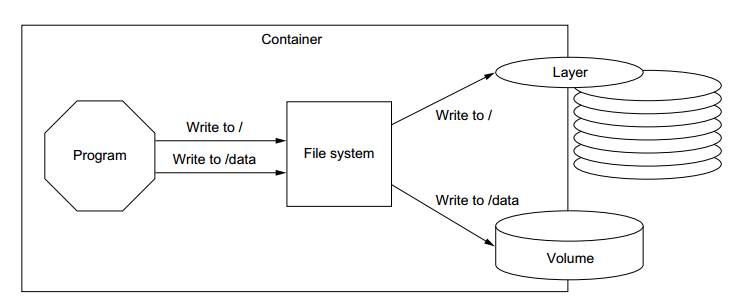
sudo docker run -d –v /dbdata –name dbdata training/postgres echo Data-only container for postgres

sudo docker run –d –volumes-from dbdata –name db1 training/postgres

可以从其他已经挂载了数据卷的容器来级联挂载数据卷



使用 --volumes-from 参数所挂载数据卷的容器自己并不需要保持在运行状态



**Volume 提供了独立于容器的数据管理功能**

Volume 能用来保存和共享数据，其所属领域和生命周期都独立于单个容器。

类似的数据有：

* 数据库软件 VS 数据库数据
* Web 应用 VS 日志
* 数据处理应用 VS 输入和输入数据
* Web 服务器 VS 静态内容
* 产品 VS 支撑的工具

Volume 有助于架构组件的模块化。映像适合打包和分发相对静态的文件，比如程序; 而 Volume 用来保存动态数据或定制内容。这种区别使得能复用映像。

例如，MySQL 程序都是相同的，可以只在一个容器中运行，而不同的数据库内容可以使用 Volume 来注入。

# Volume 类型

共有两种 Volume 类型。每种 Volume 都是主机目录树的一个位置对应到容器内的目录树上的一个挂载点，其不同只在于主机上的位置。

* 第 1 种叫绑定挂载的 Volume (bind mount volume)：用户指定将主机上的某个目录或文件挂载到容器中的目录树上。
* 第 2 种叫受管理的 Volume (managed volume)：所使用的主机上的位置是由 Docker daemon 创建并管理的，这些位置称为 Docker managed space。