

get_pools命令的实现机制

1.get_pools命令从c-sch获取需要的后端，本身c-sch不会主动和后端交互，而是被动的接受c-vol上报的后端信息。

2.get_pools命令的调用链如下：

Cinder_API

```
cinder.api.contrib.scheduler_stats.SchedulerStatsController.get_pools  
调用cinder.scheduler.rpcapi.SchedulerAPI.get_pools收集pool信息  
调用cinder.api.views.scheduler_stats.ViewBuilder.pools 组装并显示
```

Cinder_schedule

```
cinder.scheduler.rpcapi.SchedulerAPI.get_pools  
cinder.scheduler.manager.SchedulerManager.get_pools  
cinder.scheduler.filter_scheduler.FilterScheduler.get_pools
```

FilterScheduler封装了：

cinder.scheduler.host_manager.HostManager (通过配置文件可以配置，
scheduler_host_manager)

cinder.scheduler.host_manager.HostManager.get_pools ---->执行者！！！该方法从
HostManager.backend_state_map中获取信息返回到c-api

3.HostManager.backend_state_map中的信息由c-vol定时的更新：

```
cinder.Manager.SchedulerDependentManager--  
>cinder.volume.manager.VolumeManager
```

向scheduler上报状态

VolumeManager是c-vol的管理类，派生自SchedulerDependentManager。
该类封装了一个scheduler_rpcapi，并且定义了一个定时任务
_publish_service_capabilities，
定时任务定时调用shedule_api的update_service_capabilities和
notify_service_capabilities上报self.last_capabilities中的信息

从back_end获取状态

同时VolumeManager中还有一个定时任务_report_driver_status，这个任务会
定时的从后端获取信息存入self.last_capabilities