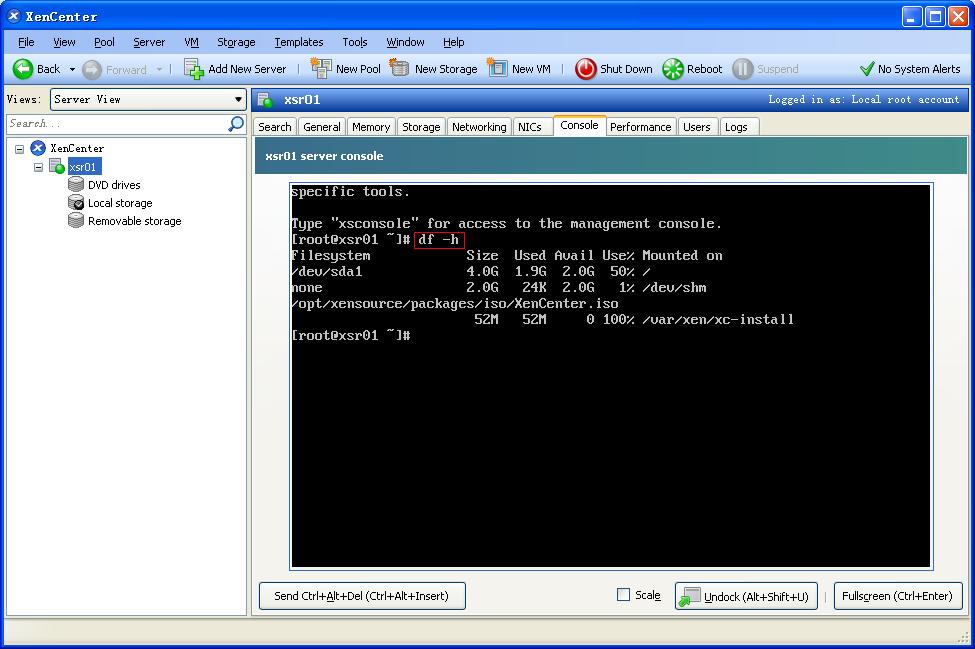
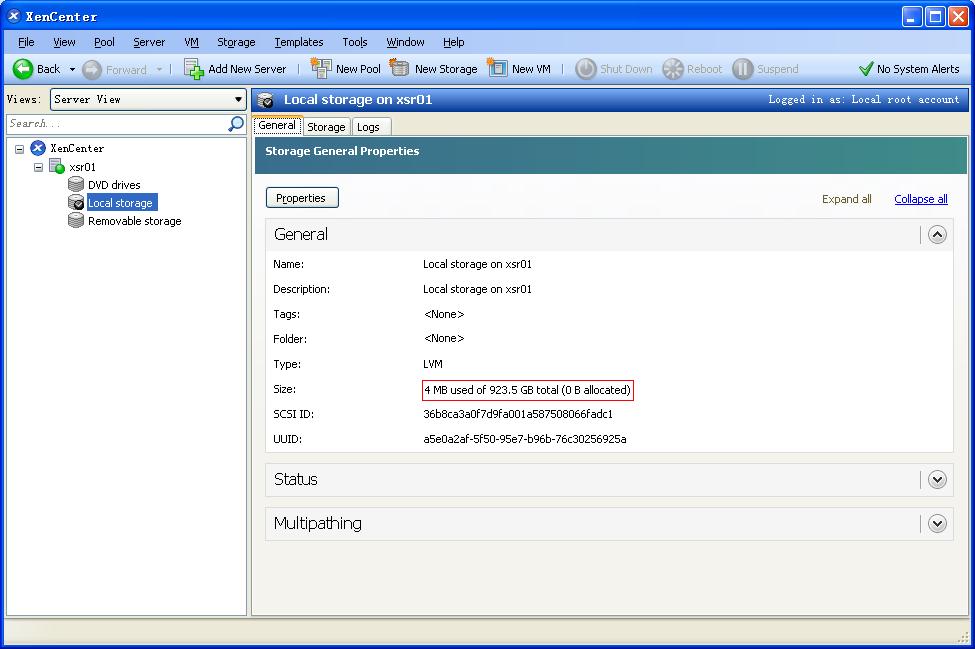
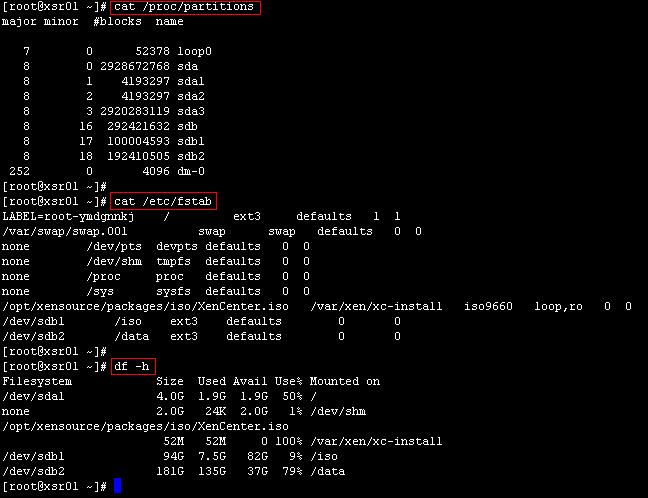
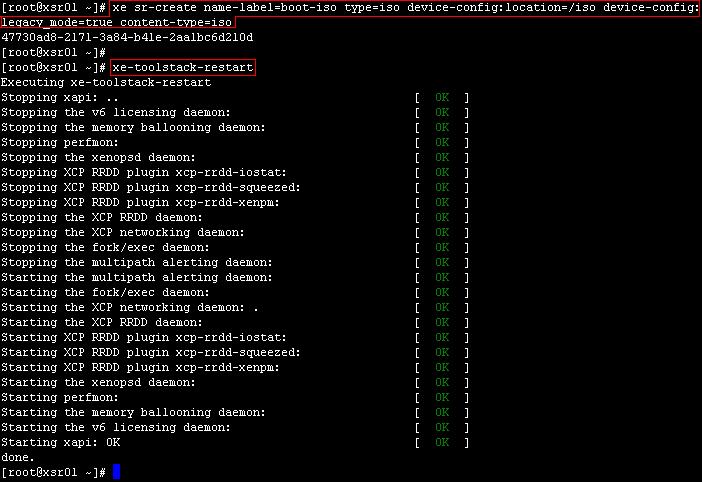
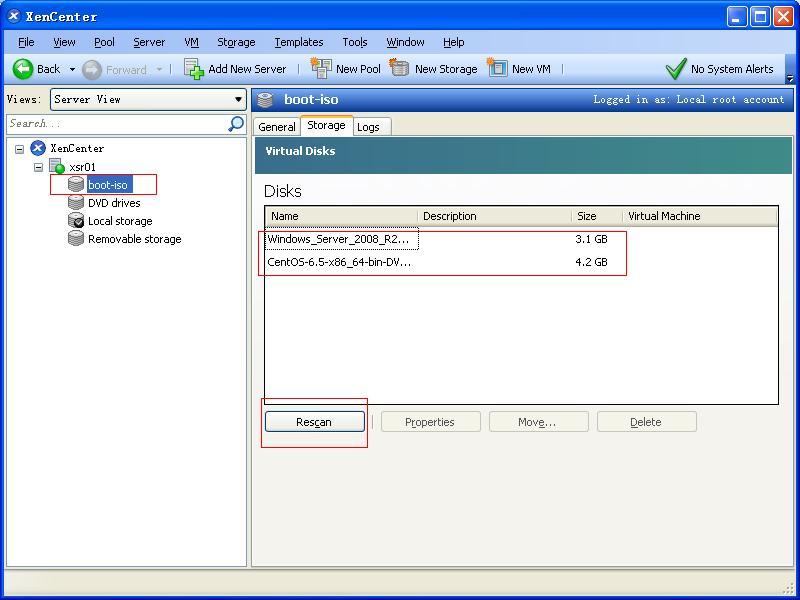
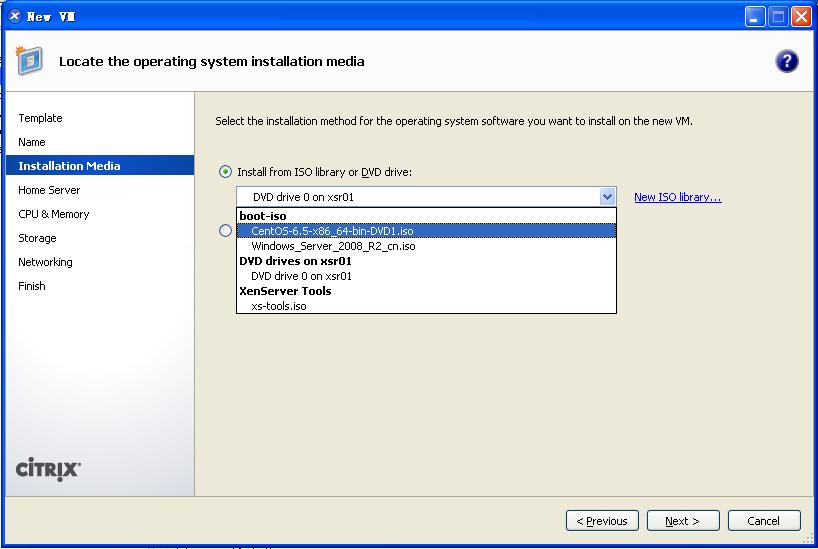
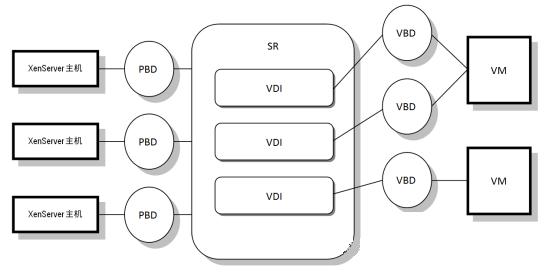
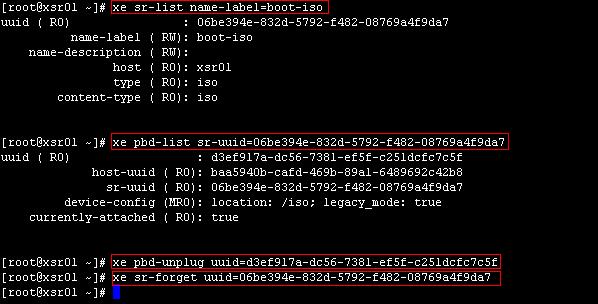
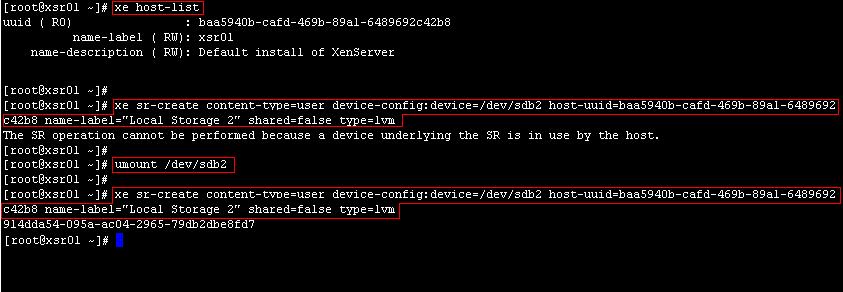
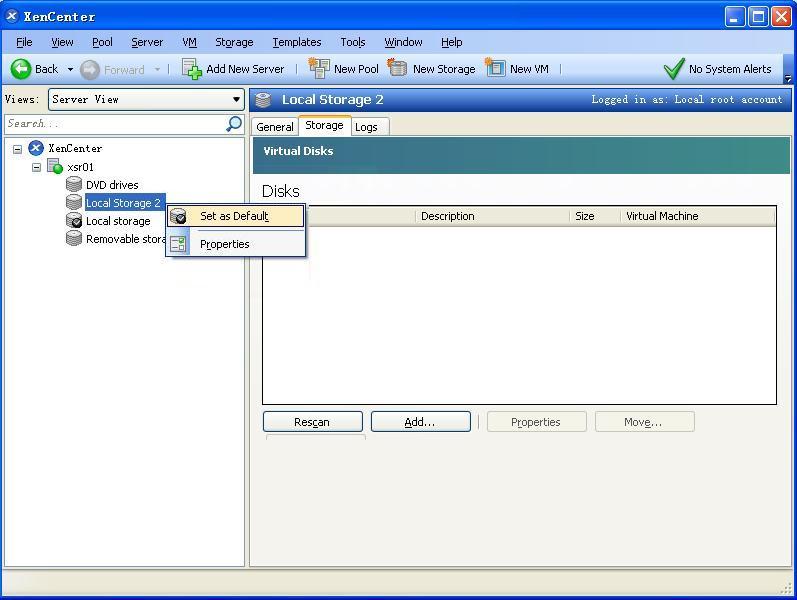
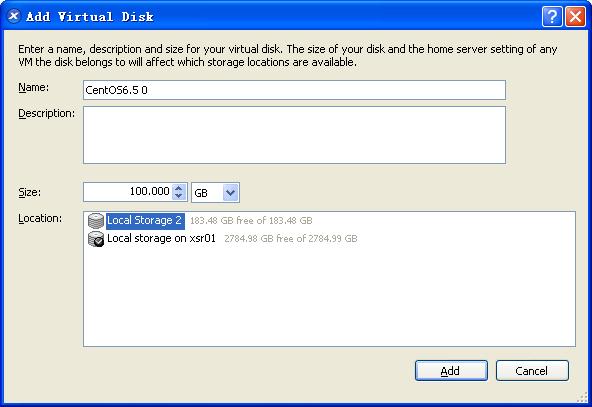
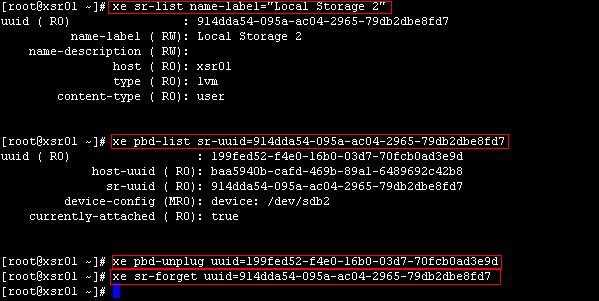
[XenServer部署系列——添加本地存储](http://blog.chinaunix.net/xmlrpc.php?r=blog/article&uid=24540106&id=4313862)

XenServer部署系统之03添加本地存储  
  
**作业环境**  
  
服务器#1  
OS：XenServer 6.2  
Hostname：xsr01  
IP：192.168.0.241/24  
Gateway：IP：192.168.0.1/24  
  
  
**一、关于XenServer存储**  
  
XenServer的安装过程，没有分区的操作，系统安装完成后，查看磁盘情况，会发现，即便是1T的硬盘，也只留了4G的空间，并且XenServer系统文件就已经占用了其中的1.9G，只剩下2G的小得可怜的空间，连一个CentOS6.5的系统镜像都放不下。放不了ISO镜像文件，那怎么能安装虚拟机呢？对于初学者而言，这是个头疼的问题。  
  
  
  
  
  
据资料介绍，XenServer的默认分区一般由三部分组成：第一个4G分区为XenServer根分区；第二个4G分区为升级操作的备份分区；第三个分区一般为磁盘的剩余空间，用于Local Storage使用。后两者都是看不见、不用作于其他存储的。  
  
那么，XenServer如何实现存储扩充呢？在XenServer中的引入存储仓库的概念，存储仓库（SR - Storage Repositories）是用来存储ISO或者VDI的。Citrix官方推荐使用共享存储，这也是XenServer设计的初衷和目标，与VMWare的产品设计很不相同。目前XenServer支持FC SAN、ISCSI SAN和NFS三种存储类型，根据虚拟化类型不同，划分不同的LUN类型来满足使用。此处涉及到存储服务器，对设备成本是有要求的。  
  
如果应用环境相对简单，比如说，只有一台Dell R720服务器，那又该如何把XenServer玩起来呢？方法当然是有的，就是通过添加一个硬盘，来达到扩充存储的效果。  
  
  
**二、添加第二块硬盘**  
  
本案中，添加一个大小为300G的SAS硬盘，现要将其划分两个分区，分别挂载到/iso和/data两个目录，前者用于存放iso镜像存储，以作为本地ISO存储库；后者用作数据存储，用于存放导出的虚拟机模板。  
  
添加硬盘的操作命令及过程，跟Red Hat Linux是一样的，按划分分区、激活分区、格式化分区、加载分区的步骤操作一遍，属于基本操作，就不详述了。  
  
以下是本案添加硬盘、分区的命令，没用到lvm，操作相对简单。  
fdisk -l  
fdisk /dev/sdb  
partx -a /dev/sdb （在这里将sdb划分为sdb1、sdb2两个分区）  
cat /proc/partitions   
mkfs.etx3 /dev/sdb1  
mkfs.ext3 /dev/sdb2  
mkdir /iso  
mkdir /data  
vi /etc/fstab （在这里将sdb1、sdb2写入到加载列表）  
mount –a  
  
操作完成后，结果显示如下（注：我这里两个分区都已经装有数据了）：  
  
  
  
  
**三、添加本地ISO储存仓库**  
  
下面为该XenServer服务器建立一个本地iso档型式的储存仓库。其实也很简单。  
  
首先，登入console，执行以下指令，创建存储仓库，  
xe sr-create name-label=< Name of the SR > type=iso device-config:location=<Storage Directory> device-config:legacy\_mode=true content-type=iso  
  
再执行以下命令，重启toolstack，  
xe-toolstack-restart  
这样XenCenter中才会同步显示iso档。但笔者在实践中，发现很多时候不执行该命令也无所谓。注：执行这个命令，网络会掉线3秒左右，然后自己恢复。  
  
本案中具体的命令下图所示，这里定义项目名称为“boot-iso”，其对应的存储目录为“/iso”。系统返回一个UUID（这个UUID在XenCenter上选择对应的SR然后选择“常规”选项可以看到），表示操作成功。  
  
  
  
新增成功，在XenCenter中，可以看到多了一个名为“boot-iso”的项目，在其右侧内容框，切换到“Storage”选项，此时框内的列表内容为空，点击“Rescan”按钮，可以看到已上传到目录/iso的iso镜像以列表形式显示。此后，新上传iso镜像到/iso目录，在XenCenter中，点击此处的“Rescan”重新扫描，即可看到。  
  
  
  
笔者在实践过程中，曾出现过以下情况：  
  
Rescan后，列表内容仍然为空，到系统中一查，发现目录/iso对应的/dev/sdb1已经自动被umount了，使用命令mount /dev/sdb1或mount –a（前提是已经在/etc/fstab文件中填写了挂载目录），再在XenCenter中Rescan，可看到镜像列表。  
  
这种情况是在执行命令xe sr-create name-label=boot-iso type=iso device-config:location=/iso device-config:legacy\_mode=true content-type=iso后发生的。之后，即使是重启系统，也不会再出现自动umount的情况。当然，如果是在/iso下再建一个iso文件夹，将目录/iso/iso作为boot-iso的话，在执行以下xe sr-create命令时，就不会出现这种情况了。  
  
另外，有些资料说到，在添加本地ISO储存的操作前，要在XenServer本地开启nfs服务，把写/iso目录写入到/etc/exports文件里，以使得该目录成为nfs共享目录。但是笔者在实践中，这个操作不做，并没影响。  
  
本地ISO存储库添加成功后，通过XenCenter新建虚拟机时，在“Installation Media”一步中，便可以选择里面的本地ISO镜像进行系统安装了。如下图所示。  
  
  
  
  
**四、删除本地ISO储存仓库**  
  
这里先科普一下。  
在XenServer中，基本的存储库（SR）、物理块设备、虚拟磁盘映像、虚拟块设备之间的关系如下图：  
  
  
  
其中，PBD为物理存储设备的抽象，基本无法改变，SR为XenServer对物理存储设备的抽象，起到组织管理的作用，SR的类型对虚机VDI的格式以及性能有一定的影响，但是SR的类型受限制于物理存储的形式，所以，暂停也跳过，VBD与VDI对应，而最终VM拿到的虚拟块设备VBD，则为VDI的虚拟，所以，SR中VDI的格式对存储空间消耗以及性能产生比较重要的影响。  
  
存储仓库通过物理块设备（PBD）与XenServer相连接，PBD包含了存储仓库（SR）的配置信息。当要从XenServer主机上分离或者删除SR之前，很有必要先将PBD断开。  
  
如果建立错误，或要重复新增，想删除储存仓库时，在CenCenter中，右击对应的SR选择Detach。如果Detach不是一个可选项，那么这个SR正被占用着（对ISO类型的SR来说有DVD挂载着，或者说有虚拟机正在其上运行着）。如果你已经分离了所有DVD驱动上的ISO并且停止了所有运行在此SR上的所有虚机，按照以下命令进行操作即可删除本地ISO存储库。  
  
首先， 运行下面的命令来确定SR的UUID：  
xe sr-list name-label=<Name of the SR>  
其次，确定对应PBD的UUID：  
xe pbd-list sr-uuid=<UUID of SR>  
再次，断开PBD：  
xe pbd-unplug uuid=<UUID of PBD>  
最后，删除记录  
xe sr-forget uuid=<UUID of SR>  
注，也可以在XenCenter上右击对应的SR，如果Forget是可选项的话，点击Forget即可。   
  
本案中具体的命令如下图所示。  
  
  
  
  
**五、添加本地虚拟机储存仓库**  
  
本案中，安装XenServer系统的硬盘大小为1T，自身存放10来个虚拟硬盘为50G的虚拟机不成问题。但在实际应用中，由于通过nfs等方式使用异地存储，本地硬盘用作存放虚拟机的元数据，因此安装XenServer系统的硬盘，即sda，一般都不会用大容量硬盘，以免浪费。  
  
但是，假如只有一台服务器用来搭建XenServer平台，且作为系统使用盘的sda容量比较小的话（比如说是一个146G的SAS硬盘），就需要通过增加硬盘设备（比如增加一个大小为300G的SAS硬盘，即sdb），以提供足够的存储空间给新增的虚拟机。  
  
本案中已增加了一个硬盘sdb，并划分了两个分区sdb1和sdb2，格式化并挂载了，sdb1被用作ISO存储仓库，下面以sdb2来演示如何增加本地lvm储存仓库（通常把这个操作说成是，增加本地存储Local Storage）。本地Lvm储存仓库，是可以被用来存储新增的虚拟机的。  
  
操作其实也很简单，与增加ISO存储库类似。   
  
首先，登入console，执行以下指令，查看本机的host uuid，  
xe host-list  
然后执行创建存储仓库的命令  
xe sr-create content-type=user device-config:device=/dev/<sdbx> host-uuid=<host-uuid> name-label="Local Storage 2" shared=false type=lvm  
  
注，另一种方法，可通过分区的序列号来实现，  
xe host-list  
ll /dev/disk/by-id  （这一步查看硬盘的信息）  
xe sr-create content-type=user device-config:device=/dev/disk/by-id/<scsi-xxxxxxxxxxxxxx> host-uuid=<host-uuid> name-label="Local Storage 2" shared=false type=lvm   
  
本案中具体的命令下图所示，这里定义项目名称为“Local Storage 2”，其对应的设备分区为sdb2。系统返回一个UUID（这个UUID在XenCenter上选择对应的SR然后选择“常规”选项可以看到），表示操作成功。  
注：这里由于sdb2在开始时，作为数据空间被加载了，导至sr-create命令无效。需要先把sdb2 umount，再进行操作。  
  
  
  
新增成功，在XenCenter中，可以看到多了一个名为“Local Storage 2”的项目。XenServer的默认Storage是“Local Storage”，即sda，在“Local Storage 2”右击工菜单中有一个选项“Set as Default”可将其设置为默认存储。  
  
  
  
以后，在新建虚拟机时，在为其分配虚拟硬盘时，既可以选择存储在Local Storage上，也可以选择存储在Local Storage 2上。如下图所示。关于如何创建虚拟机，后续篇章再述，此处略过。  
  
  
  
  
**六、删除本地lvm储存仓库**  
  
操作过程，与删除本地iso储存仓库相同，使用前文已述的命令：  
  
首先， 运行下面的命令来确定SR的UUID：  
xe sr-list name-label=<Name of the SR>  
其次，确定对应PBD的UUID：  
xe pbd-list sr-uuid=<UUID of SR>  
再次，断开PBD：  
xe pbd-unplug uuid=<UUID of PBD>  
最后，删除记录  
xe sr-forget uuid=<UUID of SR>  
  
本案中具体的命令如下图所示。  
  
  
  
  
**七、其他**  
  
本案中，sdb2的定位本是作为单纯的数据存储的，用于存储导出的后缀为.xva的虚拟机文件的。也就是常规Linux系统中的数据空间。因此XenServer默认的可用数据空间实在太小了，只有4G。  
  
在笔者的实验环境中，sdb2是一直作为数据存储空间用的，存放了很多资料，但经过上述第五、六两步的操作，再试图将sdb2 mount到系统作为数据空间里，出错，系统提示说“mount: you must specify the filesystem type”。悲剧啊，这个分区的格式被改了。  
xe sr-create content-type=user device-config:device=/dev/disk/by-id/<scsi-xxxxxxxxxxxxxx> host-uuid=<host-uuid> name-label="Local Storage 2" shared=false type=lvm   
这条命令指定sdb2的类型为lvm，但lvm显然不是分区格式，不能被挂载。暂时没找到解决的方法，只好使用命令mkfs.ext3 /dev/sdb2将其重新格式化了。  
  
注，在以后的篇章中，sdb2均是作为数据存储空间，被挂载到目录/data上使用的。