本文档主要是关于Independent卡（驱动为qla4XXX）的create vmfs的TDs和代码的对应关系。

**ESX::StorageDriver::iSCSI::Functional::CreateVMFS**

**选择合适的Lun新建磁盘分区**

**创建VMFS文件系统,比如vmfs5 datastore**

**查看指定device上的datastore信息，并且得到datastore的uuid**

**图1 在lun上面创建新的datastore**

**关于VMFS的创建中，在Atlas中填写的datastore和lun如果匹配的话，应该就不会重新建vmfs了（新建会比较慢）。（还未找相应的代码）**

**如果二者不匹配的话（通过esxcfg-scsidevs -m查看）会在lun上建一个新的datastore(将原来的datastore中的Vmdk及script拷贝到新的t\_datastore),新建过程如图1所示。创建完后需要验证device和datastore的匹配关系**

**大致的vmfs的创建过程及对应的代码已经找到，具体见后面几页。。关于TDs中新建了datastore后卸载和删除的部分没有找到。**

1.选择一个lun,得到它的磁盘分区信息；创建一个新的vmfs分区。

对应脚本 **WorkloadManager.py**🡪 **CreateVMFSDatastore**(……)函数中的

rc, result = remote.ExecuteCmd("**sh /vmfs/volumes/%s/Linux/createVMFS.sh /vmfs/devices/disks/%s"%(datastore,devicename)," %s %s"%(t\_datastore,startPartition),True**)

在**createVMFS.sh 脚本中有具体的分区createPartition ()及创建VMFS过程**

**createPartition () {**

**disk1=$1**

**start=$2**

**end=$3**

**ver=`getESXVersion`**

**if [ "$ver" == "MN" ];then**

**`partedUtil mklabel $disk1 gpt`**

**rr=`partedUtil setptbl $disk1 gpt "1 $start $end AA31E02A400F11DB9590000C2911D1B8 0"` //创建磁盘分区**

**else**

**rr=`partedUtil set $disk1 "1 $start $end 251 0"`**

**fi**

**if [ $? -ne 0 ]**

**then**

**echo "fail: can not create Partition"**

**exit 1**

**fi**

**}**

**createVMFS () {**

**disk1="$1:1"**

**datastore1=$2**

**ver=`getESXVersion`**

**if [ "$ver" == "MN" ];then**

**rr=`vmkfstools -C vmfs5 -b 1M -S $datastore1 $disk1` //** Create vmfs5 datastore

**else**

**rr=`vmkfstools -C vmfs3 -b 1M -S $datastore1 $disk1`**

**fi**

**if [ $? -ne 0 ]**

**then**

**echo "fail: can not create VMFS on partition"**

**exit 1**

**fi**

**}**

**2.检查在指定的device上datastore是否存在。**

**WorkloadManager.py🡪 CheckVMFS()**

def CheckVMFS(self,t\_datastore,devicename):

remote = STAFRemote(self.esxhost)

log.info("Checking VMFS on %s" % devicename)

rc, result = remote.ExecuteCmd("**esxcfg-scsidevs ", "-m | grep %s"%(devicename), True**)

**//查看新创建的datastore信息**

if rc != ReVal.PASSED:

log.error("Failed to get VMFS info on %s. Return code is %s" % (self.esxhost, ReVal.toString[rc]))

return False

rc, result2 = remote.ExecuteCmd("**vmkfstools -Ph /vmfs/volumes/%s| grep -i uuid | awk -F: '{print $2}'" % t\_datastore)//得到 t\_datastore的uuid ??**

if rc != ReVal.PASSED or not result2:

log.error("Fail to get uuid for datastore %s" % t\_datastore)

return False

if (t\_datastore in result) and result2:

return True

return False