# A.Overview Step

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Once you insert new hard disks into your system, you’ll typically use utilities like [fdisk](https://www.thegeekstuff.com/2010/09/linux-fdisk/) or [parted](https://www.thegeekstuff.com/2011/09/parted-command-examples/) to create partitions. Once you create a partition, you’ll use [mkfs command](https://www.thegeekstuff.com/2013/01/mke2fs-examples/) to create ext2, ext3, or ext4 partition.  Once you create a partition, you should use mount command to mount the partition into a mount point (a directory), to start using the filesystem   |  | | --- | | Quota is based upon filesystems, but you can always create a virtual filesystem and mount it on a specific (empty) directory with the usrquota and/or grpquota flags. In steps this will be: 1. create the mount point 2. create a file full of /dev/zero, large enough to the maximum size you want to reserve for the virtual filesystem 3. format this file with an ext3 filesystem (you can format a disk space even if it is not a block device, but double check the syntax of every - dangerous - formatting command) 4. mount the newly formatted disk space in the directory you've created as mount point, e.g.  Code:  mount -o loop,rw,usrquota,grpquota /path/to/the/formatted/disk/space /path/of/mount/point  5. Set proper permissions 6. Set quotas and the trick is done. |  |  | | --- | | Detail Steps  1. Create the mount point  Code:  # mkdir /var/virtual\_disks/directory\_with\_size\_limit  2. Create a file full of /dev/zero, large enough to the maximum size you want to reserve for the virtual file-system  Code:  # touch /var/virtual\_disks/directory\_with\_size\_limit.ext3  # dd if=/dev/zero of=/var/virtual\_disks/directory\_with\_size\_limit.ext3 bs=QUOTA\_SIZE count=1  3. format this file with an ext3 file-system (you can format a disk space even if it is not a block device, but double check the syntax of every - dangerous - formatting command)  Code:  # mkfs.ext3 /var/virtual\_disks/directory\_with\_size\_limit.ext3  4. mount the newly formatted disk space in the directory you've created as mount point, e.g.  Code:  # mount -o loop,rw,usrquota,grpquota /var/virtual\_disks/directory\_with\_size\_limit.ext3 /path/of/mount/point  As a result you now have a directory in /path/of/mount/point with a size limitation.  If you wish to add more space to (trim the size of) the directory:  Code:  # umount /path/of/mount/point  # e2fsck -f /var/virtual\_disks/directory\_with\_size\_limit.ext3  # resize2fs -p /var/virtual\_disks/directory\_with\_size\_limit.ext3 NEW\_SIZE  # mount -o loop,rw,usrquota,grpquota /var/virtual\_disks/directory\_with\_size\_limit.ext3 /path/of/mount/point |  |  |  | | --- | --- | | Once I increased the size of a virtual filesystem on a Fedora Core 5 machine and never experienced any problem with it. I had a filesystem of 150 Mb in /usr/virtual-disk/data.ext3 and I mounted it on /data. The following commands worked for me  Code:   |  | | --- | | umount /data  **e2fsck -f** /usr/virtual-disk/data.ext3  **resize2fs** -p /usr/virtual-disk/data.ext3 300M  mount -o loop,rw,usrquota,grpquota /usr/virtual-disk/data.ext3 /data |   I doubled the size of the filesystem and no data was lost. Anyway, better to do a backup of any valuable data when performing filesystem operations. | |

# ---------------------------------------

# 1. dd command explain

|  |
| --- |
| Check the explaination in E:\Castis\_Current\_Working\System\_performance\Disk\_Performance |

# 2.1. fdisk

|  |
| --- |
| How Do I Create a New ext3 File System If a Disk Was Added To The Server / Desktop Under Any Linux Operating Systems? |

# 2.2. mkfs – format linux filesystem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Whatever may be your reason(s) all file system creating involves creations of superblock, inode and other filesystem metadata structure. Fortunately, Linux comes with mkfs command to format filesystem. It is used to build a Linux file system on a device, usually a hard disk partition.   |  | | --- | | General syntax of mkfs is as follows: mkfs -t filetype /dev/DEVICE OR mkfs.ext3 /dev/DEVICE OR mkfs.ext4 /dev/DEVICE  WHERE:   * -t filetype : File system type, it can be ext3, ext2, etc4, vfat etc * /dev/DEVICE : Your device name i.e. partition /dev/hda1 or /dev/sda1 etc. |  An example Suppose you would like to format /dev/hda5 with ext3 file system.   |  | | --- | | Step #1 Create the new filesystem with following command (first login in as a root user) # mkfs.ext3 /dev/sda5 Sample outputs:  mke2fs 1.35 (28-Feb-2004)  Filesystem label=  OS type: Linux  Block size=1024 (log=0)  Fragment size=1024 (log=0)  30120 inodes, 120456 blocks  6022 blocks (5.00%) reserved for the super user  First data block=1  15 block groups  8192 blocks per group, 8192 fragments per group  2008 inodes per group  Superblock backups stored on blocks:  8193, 24577, 40961, 57345, 73729  Writing inode tables: done  Creating journal (4096 blocks): done  Writing superblocks and filesystem accounting information: done  This filesystem will be automatically checked every 38 mounts or  180 days, whichever comes first. Use tune2fs -c or -i to override. |  |  | | --- | | Step # 2: Create mount point directory for the file system # mkdir /datadisk1 Step # 3: Mount the new file system # mount /dev/sda5 /datadisk1 |  |  | | --- | | Step # 4: Finally make sure file system /dev/hda5 automatically mounted at /datadisk1 mount point after system reboots. You need to add partition to /etc/fstab file. Use text editor such as vi to add following entry # vi /etc/fstab  Add/append following entry to file: /dev/sda5 /datadisk1 ext3 defaults 0 2  Where,   * /dev/sda5 : File system or parition name * /datadisk1 : Mount point * ext3 : File system type * defaults : Mount options (Read man page of mount command for all options) * 0 : Indicates whether you need to include or exclude this filesystem from dump command backup. Zero means this filesystem does not required dump. * 2 : It is used by the fsck program to determine the order in which filesystem checks are done at reboot time. The root (/) filesystem should be specified with a #1, and otherfilesystems should have a # 2 value.   Save file and exit to shell prompt. For more information see mkfs(8) page. | |

# 3.mount umount

|  |
| --- |
| Once you insert new hard disks into your system, you’ll typically use utilities like [fdisk](https://www.thegeekstuff.com/2010/09/linux-fdisk/) or [parted](https://www.thegeekstuff.com/2011/09/parted-command-examples/) to create partitions. Once you create a partition, you’ll use [mkfs command](https://www.thegeekstuff.com/2013/01/mke2fs-examples/) to create ext2, ext3, or ext4 partition.  Once you create a partition, you should use mount command to mount the partition into a mount point (a directory), to start using the filesystem.  The general mount command syntax to mount a device:  mount -t type device destination\_dir |