CSE 5031 Operating Systems
2020/21 Fall Term

Project: 6
Topic: Mass Storage Management and File Systems
Date: 27/12 /2020 – 07/01/2021

Objectives:

- to add disk storage to a system
- to create and mount File Systems
- to experiment with file and directory sharing

References:

- **Red_Hat_Enterprise_Linux-7-Storage_Administration_Guide**
  https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/storage_administration_guide/index
- **Red Hat Enterprise Linux-7 System Administrators Guide**
  https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/index.html
- **Oracle VM VirtualBox User Manual** (Help->Contents or https://www.virtualbox.org/manual/UserManual.html)

Section A. Project Definition

A.1 Problem Statement

The project consists of two phases.

i) The **first phase** aims at emulating the process on adding a **new disk storage** unit to a **Linux** system; mounting it under File System Hierarchy; and using it.

ii) The **second phase** focuses on the study of UNIX/Linux **file** and **directory sharing** on the platform created at the first phase. You will experiment with file and directory sharing by creating **hard links** to files that are stored in the **same file system**; and **symbolic (soft) links** to directories, and to files that are located in **different file systems**.

A.2 Implementation Constraints

i) Cloning the Guest Virtual Machine

Storage management involving mounting and unmounting File Systems in the FSH are risky operations, prone to procedural errors that may corrupt your **OS**; you may have difficulties to trace the problem(s) and remedy them. Thus you are highly recommended to clone the **Guest VM "C7 Prj"** you were working with.

In the remainder we will refer to the **cloned VM** as "**FS-tst**" and the **default machine** folder you have created for **VirtualBox** platform as "**C:\VMs**". In case you are using a different nomenclature and locations, substitute these two terms in the project text with your labels.

✓Note that, you may refrain from cloning procedure if your disk storage is limited. In case of failure, you have however refer to **Project 1 Part 1** guidelines to recreate your **Guest VM**.

ii) Backup Mount File / Reinitialize Mount Directory

The "**/etc/fstab**" file defines the list of File Systems the **OS** should mount when it starts. Display the contents of this file using the "cat" command; it should only contain **3 Linux** partitions sda1, sda2 and sda3 that are defined at system generation forming the initial **FSH**.

Note that each filesystem is described on a separate line; **fields** on each line are separated by tabs or spaces. Lines starting with '#' are comments, blank lines are ignored (refer to "man fstab").

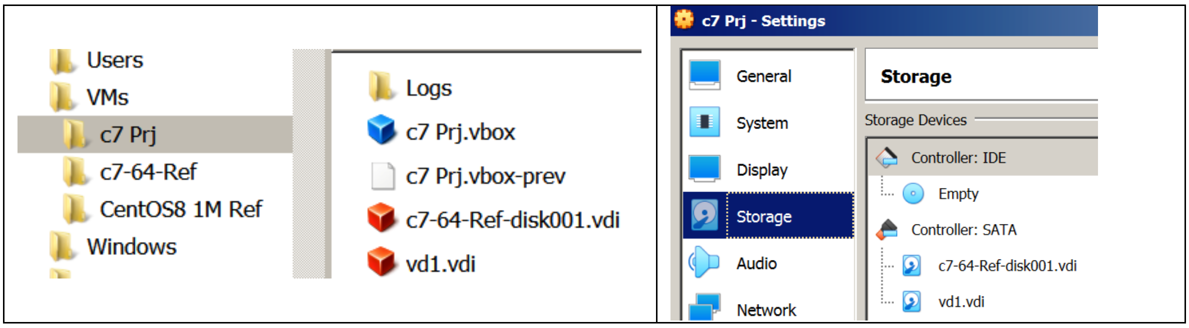
- ○ The first field is the id of the block device to be mounted **"/dev/sda1"** or its **UUID**. You may list the mapping of block device names to their UUID using the "**sudo blkid**" command.
- ○ The second field is the mount point in FSH.
- ○ The third field is the **type of the File System** stored in the partition.
- ○ The forth field are the **mount options** (rw, setuid, defaults etc.)

As the **root** user save (copy) the "**/etc/fstab**" file under a different name. You will copy it back if "**/etc/fstab**" gets corrupted by your **mount/unmount** operations.

The **mount directory** to be used in the project is "**/mnt**" do not forget to **clean it up** as necessary.

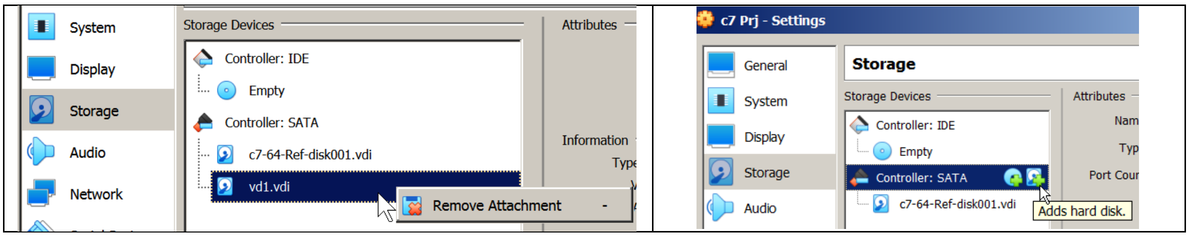
iii) Maintaining VirtualBox Virtual Disk Definitions

VirtualBox creates its **virtual disks** under the home directory of the **Guest VM** such as **"C:\VMs\C7 Prj".** The screen shot on the left shows **two disk unit** definitions in the VM directory, and right screen shut here after shows that these disc units are connected to the SATA controller on the **Guest VM**.



You may **remove** a **disc unit** by right clicking on its icon e.g. **"vd1.vdi"** in the **Settings/Storage** window (left screen shut here after), and clicking on the **OK** button located at the bottom of the right pane.

The **disc unit** is **disconnected** from the controller, but the **disk "vd1.vdi"** remains **intact** in the home directory, and you may **mount it** again by **clicking on** "Add hard disk" icon (right screen shut here after) and **double clicking** on the **"vd1.vdi"** line displayed in the windows that opens.

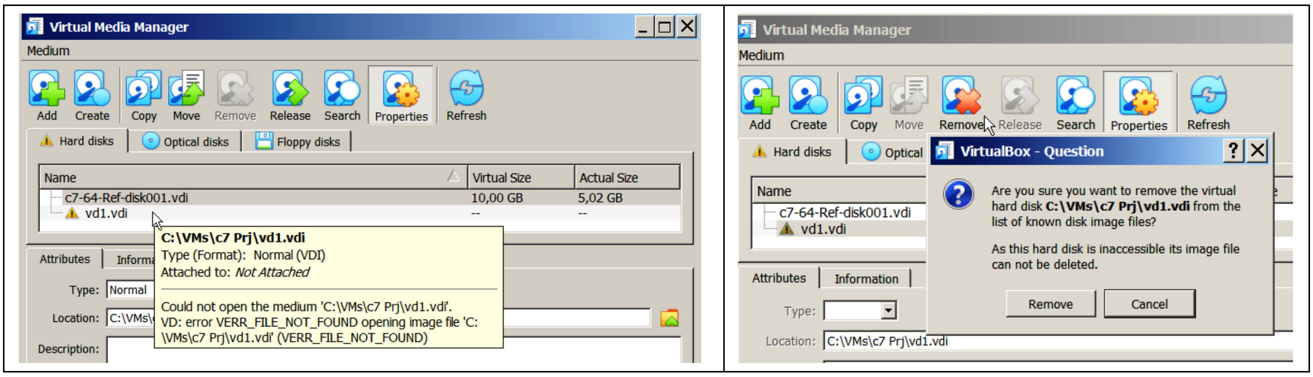


You may **discard** a **disc unit** definitively that is disconnected by deleting its file stored in the home directory e.g. **"C:\VMs\C7 Prj\vd1.vdi".**

✓ Note that this operation is correct it leaves a **dangling pointer** of in VirtualBox platform as the "**Virtual Media Manager**" keeps a record of **"vd1.vdi"** existence. You may remove it as explained here after.

Open "**Virtual Media Manager**" using the **File** tab of the "**VirtualBox Manager**":

- the disk **"vd1.vdi"** will be displayed with a warning (left screen shut here after). The error message will explicitly warn you that the **FILE IS NOT FOUND**.
- Select disk **"vd1.vdi"** definition and click on the "Remove" icon on the top (right screen shut here after) and proceed with the deletion of this reference.



Name	Virtual Size	Actual Size
c7-64-Ref-disk001.vdi	10,00 GB	5,02 GB
vd1.vdi	--	--

Type: Normal

Location: C:\VMs\

C:\VMs\c7 Prj\vd1.vdi
Type (Format): Normal (VDI)
Attached to: *Not Attached*

Could not open the medium 'C:\VMs\c7 Prj\vd1.vdi'.
VD: error VERR_FILE_NOT_FOUND opening image file 'C:\VMs\c7 Prj\vd1.vdi' (VERR_FILE_NOT_FOUND)

Are you sure you want to remove the virtual hard disk **C:\VMs\c7 Prj\vd1.vdi** from the list of known disk image files?

As this hard disk is inaccessible its image file can not be deleted.

Type:

Location: C:\VMs\c7 Prj\vd1.vdi

Now you may **create again new disks** from scratch as explained in the next section.

Section B. Adding a New Disk to a Linux System

B.1 Implementation Steps

On a **real system** adding a new disk to a system involves its provisioning and connection to a **suitable controller**. Under **ORACLE VM VirtualBox** platform these **two steps** correspond to the creation of the virtual disk and its connection to a virtual controller when the VM of powered off (as in the real case – no hot plug in).

The remaining procedures to mount the new disk and make it available under Linux **File System Hierarchy** are alike on both **real** and **virtualized** platforms. They consist of:

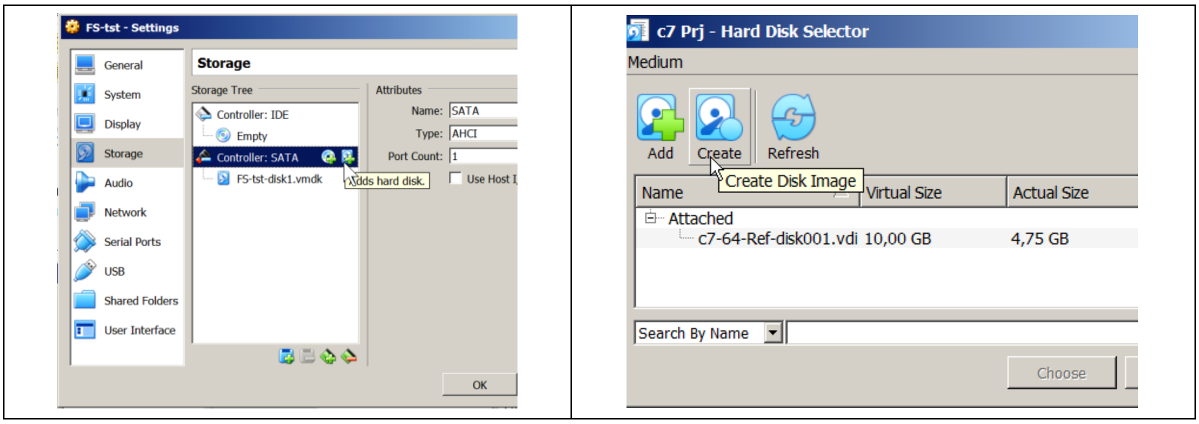
- ✓ formatting disk unit (MBR or GPT);
- ✓ partitioning it (in at least 1);
- ✓ installing a File System (ext4, ntfs, FAT32 etc.) on each partition;
- ✓ mounting partitions in the **FSH**, and defining suitable **file attributes** (rwx).

B.2 Creating and Connecting a Virtual Disc

In case you have already created and removed virtual disk units make sure that "**Virtual Media Manager**" is in clean state by following the guidelines provided in section **A.2.iii.**

To create a **virtual disk** and connect it to **FS-tst** SATA controller perform the steps outlined here after.

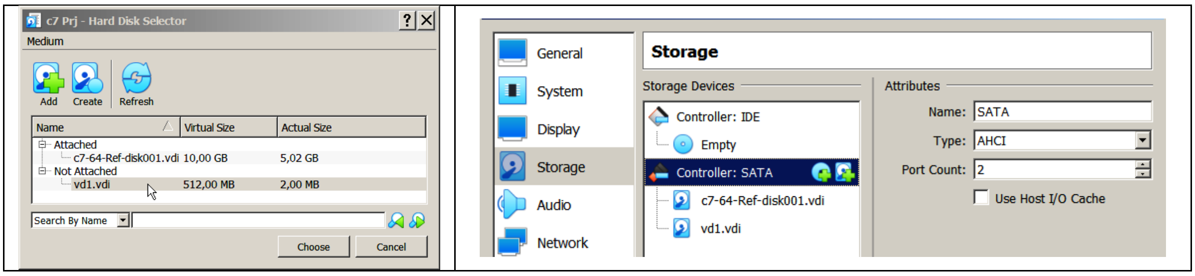
- ✓ Open **VM VirtualBox Manager,** select **FS-tst** on the left pane, expand "**Storage**" entry (left screen shut).
- ✓ Select "**Controller: SATA**" and click on the "**Add Hard Disk**" icon (left screen shut).
- ✓ On the "**Hard Disk Selector**" menu select "Create" tab icon (right screen shut).
- ✓ Select successively the options: (i) **"create a New Virtual Disk**"; (ii) **VDI** (VirtualBox Disk Image) type; (iii) **"Dynamically Allocated**" space.
- ✓ Name the virtual disk as "vd1", and set its size exactly to **512 MiB**; click on '**Create'** then on '**OK'.**



Name: SATA
Type: AHCI
Port Count: 1

Name	Virtual Size	Actual Size
Attached		
c7-64-Ref-disk001.vdi	10,00 GB	4,75 GB

- ✓ Select virtual disk "vd1" entry shown on the left screen shut here after; and to connect double click on it.
- ✓ You have created the virtual disk and connected it to SATA port #1 as shown on the right screen shut.



Name	Virtual Size	Actual Size
Attached		
c7-64-Ref-disk001.vdi	10,00 GB	5,02 GB
Not Attached		
vd1.vdi	512,00 MB	2,00 MB

Name: SATA
Type: AHCI
Port Count: 2
☐ Use Host I/O Cache

✓Note that "vd1" is connected to the **SATA Port#1**, whereas the system disk is on the **SATA Port#0**

B.3 Inspecting Block Devices and Mounted File Systems

Start **FS-tst** and perform all the procedures listed herein as the "root" user.

1. Display all **block** devices that are connected to the system using the "**lsblk**" command (**System Administration** Guide Chapter **21.4.1**), and note that:
   - ✓ 2 **SATA** disks labelled respectively as "**sda**", and "**sdb**" are connected to the system.
   - ✓ "**sda**", the disk that contains **basic OS partitions**, is divided into **3** partitions named respectively as:
     - ○ "**sda1**", the "boot" partition;
     - ○ "**sda2**", the "swap" partition; and
     - ○ "**sda3**"; the "root" partition.

     and "**sda1**" and "**sda3**" are mounted in **Linux** FSH at mount points "**/boot**" and "**/**" respectively.
   - ✓ the new virtual disk "**sdb**" is **not** partitioned and not mounted!

2. Verify your findings in step (1) using the following commands:
   - ✓ list the "/dev" directory system using the "**ls -la /dev | grep sd**" command and confirm that the new **hard disk** is defined as a **block device** file; chech its major and minor numbers;
   - ✓ display the format of connected block devices with the "**parted -l**" command (**Storage Administration** Guide Chapter **13.1**):
     a. **"/dev/"sda**" format is MBR,
        - ○ its physical attributes: size sector size are provided,
        - ○ contents of 3 partition tables provide their location on the disk, the File System installed;

     b. **"/dev/"sdb**" exhibits an **error message** since it is not formatted, but its physical attributes are listed.

3. List **mounted file systems** on **FS-tst** using the "**blkid -o list**" command (**System Administration** Guide Chapter **21.4.2**).
   Identify the **UUID** assigned to each partition that can be used instead of its file name e.g. "**UUID=33c4 ....XXX**" instead of the device name "/dev/sda1".

4. Display disc utilization ratios of **mounted file systems** on **FS-tst** using the "**df -h**" command (**System Administration** Guide Chapter **24.4.4**).

5. Display the contents of the "**/etc/fstab**" file and identify the **file systems** mounted at system startup (refer to section **A.2.ii**); and crosscheck the device file name using the **blkid**" command,

✓ **Note that**, Linux supports a large number of **file systems** and provide a homogenous **API** to all of them through its **virtual file system** layer. Nevertheless, only a limited number of **file systems** is included in the kernel configuration; those selected during **OS** generation; other **file systems** can be installed, configured, loaded **at run time** as needed.

The list of **file systems** that are currently available is defined in the "**/proc/filesystems**" file; display its contents, and verify that the "**xfs**" file system that is installed on "**/dev/sd1**" and "**/dev/sd3**" is present.

B.4 Formatting – Partitioning – Installing a File System on a Disk Storage

UNIX/Linux provides the following **command line** tools to manage disk storage units:

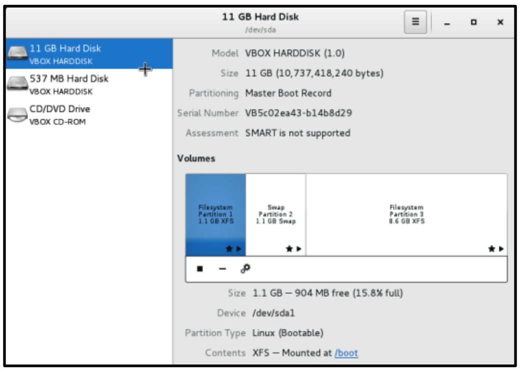
- ➢ formatting and partitioning may be performed with the "**fdisk**" or the "**parted**" command, and
- ➢ installation of a the **file system** with the "**mkfs**" command.

These command line tools offer comprehensive set of configuration options, and they are the only way to manage storage units on servers without a user GUI.

In this project you will use the graphical "**Disks**" utility provided by **GNOME**, the GUI installed on our systems.

i) Using GNOME Disk Utility

- ✓ Log on **FS-tst** as the "**root**" user, and start **GNOME Disk** tool from **Applications -> Utilities -> Disks**.
- ✓ Select the **first** entry on the left pane the "**/dev/ sda**" disk layout with its 3 required **Linux** partitions.
- ✓ Display the characteristics of each partition by clicking on its graphical representation as shown on the right screen shut.
- ✓ Examine for each partition:
  - + partition name e.g. "**sda1**";
  - + file system type;
  - + mount point in the **FSH**.
- ✓ Select CD/DVD drive and perform similar examinations



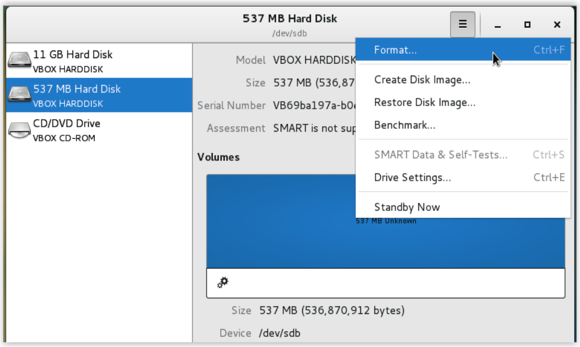
Model VBOX HARDDISK (1.0)
Size 11 GB (10,737,418,240 bytes)
Partitioning Master Boot Record
Serial Number VB5c02ea43-b14b8d29
Assessment SMART is not supported

Volumes

Size 1.1 GB — 904 MB free (15.8% full)
Device /dev/sda1
Partition Type Linux (Bootable)
Contents XFS — Mounted at /boot

ii) Formatting "/dev/sdb/" with GNOME Disk Utility

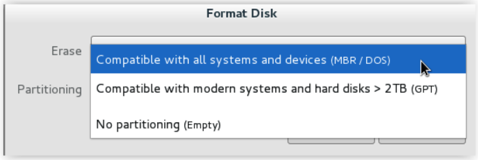
- ✓ Select "**/dev/sdb**" device on the left pane of **GNOME Disk Utility** window (left screen shut here after).
- ✓ Open "**operations list**" by clicking its icon on the top; and select "**Format**" option.
- ✓ Use "**Format Disk**" pull down menu to select "**MBR**" partition format (the top right screen shut here after).
- ✓ Click on "**Format**" button (the bottom right screen shut here after).
- ✓ Verify "**MBR**" partition format information with the "**parted -l**" command.



Model VBOX HARDDISK
Size 537 MB (536,87
Serial Number VB69ba197a-b0
Assessment SMART is not sup

Volumes

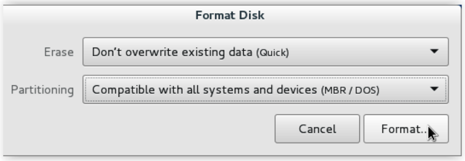
Size 537 MB (536,870,912 bytes)
Device /dev/sdb


Format Disk

Erase

Partitioning

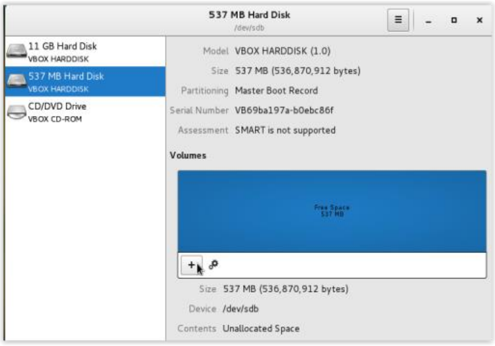
Compatible with all systems and devices (MBR / DOS)
Compatible with modern systems and hard disks > 2TB (GPT)
No partitioning (Empty)


Format Disk

Erase Don't overwrite existing data (Quick)
Partitioning Compatible with all systems and devices (MBR / DOS)

iii) Creating Partitions "/dev/sdb1" & "/dev/sdb2"and Installing the File System "ext4"

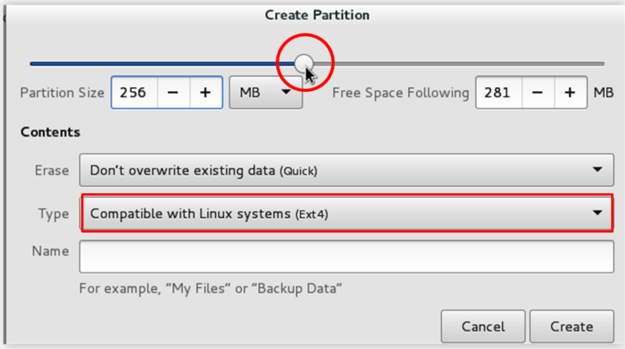
- ✓ Select "**/dev/sdb**" device on the left pane of **Disk Utility;** and click on the **add partition icon" +**" at the bottom of the right pane (left screen shut here after).
- ✓ Set the partition size of "**sdb1**" to **256 MiB** using the **ruler** located on the top of the "**Create Partition**" menu (right screen shut here after).
- ✓ Keep the "**ext4**" option for partition's **File System type**; and press the button "**Create**".



Model VBOX HARDDISK (1.0)
Size 537 MB (536,870,912 bytes)
Partitioning Master Boot Record
Serial Number VB69ba197a-b0ebc86f
Assessment SMART is not supported

Volumes

Size 537 MB (536,870,912 bytes)
Device /dev/sdb
Contents Unallocated Space


Create Partition

Partition Size 256 MB Free Space Following 281 MB

Contents

Erase Don't overwrite existing data (Quick)
Type Compatible with Linux systems (Ext4)
Name

For example, "My Files" or "Backup Data"

- ✓ Create "**s**db2" partition by clicking on the "**Free Space**" area on the right pane of the **Disk Utility** window; and perform the actions listed above to create and configure it with the "**ext4**" FS type.
- ✓ Verify that **"/dev/sdb1**", and**"/dev/sdb2**" partitions are defined and the your file systems are installed with th "**blkid -o list**" command, and note they are but not yet mounted in the **FSH**.
- ✓ Cross check these results with the "**parted -l**" the output of the command.
- ✓ Display the mount table "**/etc/fstab**" and verify that it only contains "/**dev/sda***" partition definitions.

Section C. Mounting File Systems on Linux File System Hierarchy

On **UNIX/Linux file systems** on different partitions and removable devices (CDs, DVDs, or USB flash) should be attached to a certain point (the mount point) in the **FSH**, in order to be accessible, and can be detached again.

For instance, if the **FS** created on the "/dev/sdb1" device/partition is mounted at the "/mnt/mylib" point, its contents will be the accessible using the path "/mnt/mylib**/xxxxxx**".

C.1 Mounting File Systems Manually

To attach or detach a file system, we use the "**mount**" or "**umount**" command respectively; refer to manual pages ("**man mount**") and to the **System Administration** Guide 5.1 for further detals. The command:

```
mount   [-t  FS_type]  [-o  options]  [device]  [directory (mount_point)]
```

tells the kernel to attach the filesystem found on **device** of **FS type** at the directory **mount_point**.

The previous contents (if any) and owner and mode of **mount_point** become invisible, and as long as this filesystem remains mounted, the pathname **mount_point** refers to the root of the filesystem on device.

i) Mounting "ext4" File System installed on /dev/sdb1

To mount "**ext4**" file system installed on **/dev/sdb1** device on the **FSH** perform the steps defined here after.

- ✓ As the **root** user create the mount directory "**/mnt/vdb1**", a mnemonic that refers to the partition definition, use the "**mkdir /mnt/vdb1**" command.
- ✓ Mount **"/dev/sdb1**" on "**/dev/vdb1"** with the "**mount /dev/sdb1 /mnt/vdb1**"command.
- ✓ List mounted file systems with the "**df -h**" command to verify the correctness of the operation.
- ✓ Check if mounted file system operates correctly:
  - ○ by copying **/etc/group** file under **/mnt/vdb1**
  - ○ by displaying **/mnt/vdb1** contents with with the "**ls -al /mnt/vdb1**"command.
- ✓ List the attributes of the mount point **/mnt/vdb1** with "**ls –ali**"; and change them to grant others access rights to list its content, to create directories/files, and to perform "cd", using "**chmod o=rwx /mnt/vdb1".**
- ✓ To test that **/mnt/vdb1** is accessible to others:
  - ○ login as "**sysadmin**";
  - ○ create the "**/mnt/vdb1/work**" directory;
  - ○ copy **/etc/passwd** file as **/mnt/vdb1/work/pw**; and list its contents.

ii) Mounting "ext4" File System installed on /dev/sdb2

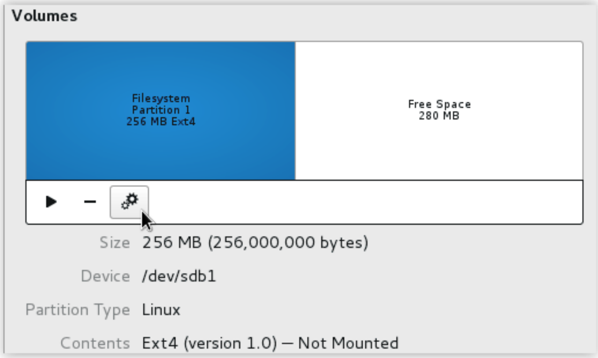
To mount the **ext4 file system** installed on **/dev/sdb2** onto mount directory "**/mnt/vdb2**" by repeat the operations defined in (i).

To verify your setup:

- ✓ login as "**sysadmin**";
- ✓ create with **gedit** the **"/mnt/vdb2/xx.txt**" file that contains the text "**Ordinary file xx.txt is stored in the directory /mnt/sdb2**".
- ✓ list the file contents.

C.2 Automating Mounting of File System /dev/sdb1

The **OS** mounts **File systems** defined in the "**/etc/fstab**" file at startup, if they are specified with the "**auto**" option. You have the option to edit this file to insert your File System definitions, or use the **Disk Utility** as follows.



Volumes

Filesystem
Partition 1
256 MB Ext4

Free Space
280 MB

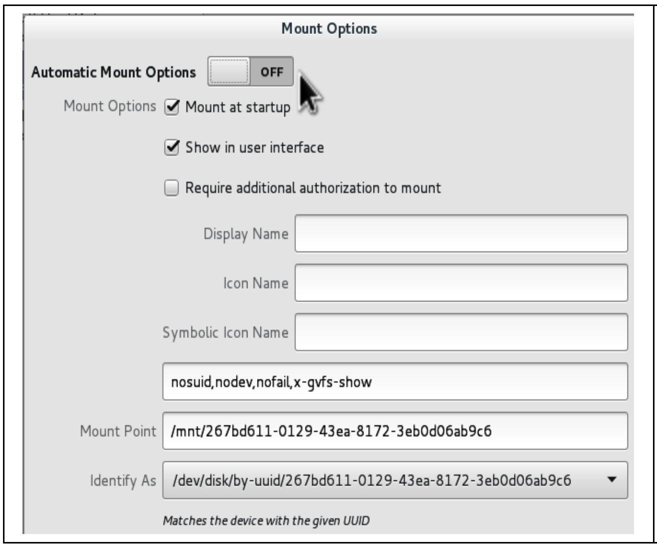
Size 256 MB (256,000,000 bytes)

Device /dev/sdb1

Partition Type Linux

Contents Ext4 (version 1.0) — Not Mounted

- ✓ Log on **FS-tst** as the "**root**" user, and start the **GNOME Disk Utility**.
- ✓ Select the on the left pane the "**/dev/sdb**" disk
- ✓ On the right pane click on the partition depicting the "**/dev/sdb1**" unit.
- ✓ Open the **tool box** (the gear icon) as shown on the right screen shut.
- ✓ Select "**Edit mount Options**" entry from the drop down menu.
- ✓ Switch the "**Automatic Mount Options**" **OFF** as shown on the left screen shut here after.
- ✓ Note that the **Disk Utility** has:
  - ○ defined for **/dev/sdb1** the **Mount Point** "/**mnt/267bdxxxx|9c6**" using its **UUID** ;
  - ○ chosen to Identify **/dev/sdb1** also with its UUID as "**/dev/disc/by-uuid//267bdxxxx|9c6**"
  - ○ defined mount attributes (nosuid, nofail …);
  - ○ selected the "**x-gcfs-show**" attribute (second tick box marked) is to inform the **GNOME** GUI to display the icon of **/dev/sdb1** on the window**.**
- ✓ Redefine these settings as shown on the right screen shut here after; and press the button **OK**.
- ✓ Display the "**/etc/fstab**" file and examine the line appended by the Disk Utility.



Mount Options

Automatic Mount Options OFF

Mount Options ☑ Mount at startup

☑ Show in user interface

☐ Require additional authorization to mount

Display Name

Icon Name

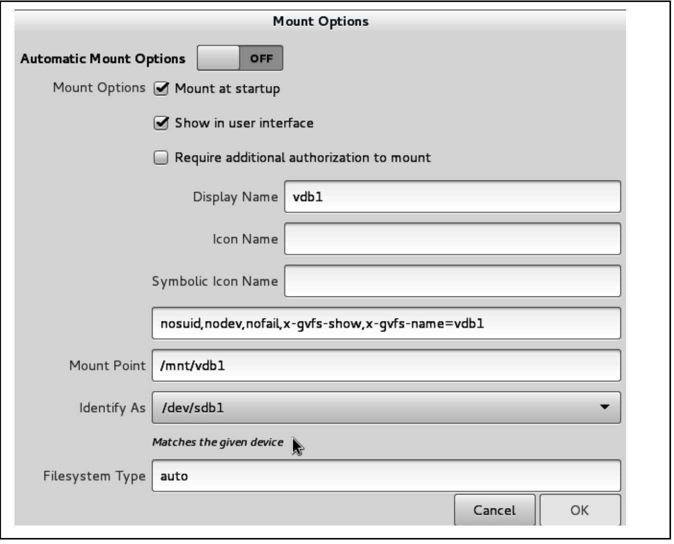
Symbolic Icon Name

nosuid,nodev,nofail,x-gvfs-show

Mount Point /mnt/267bd611-0129-43ea-8172-3eb0d06ab9c6

Identify As /dev/disk/by-uuid/267bd611-0129-43ea-8172-3eb0d06ab9c6

Matches the device with the given UUID



Mount Options

Automatic Mount Options OFF

Mount Options ☑ Mount at startup

☑ Show in user interface

☐ Require additional authorization to mount

Display Name vdb1

Icon Name

Symbolic Icon Name

nosuid,nodev,nofail,x-gvfs-show,x-gvfs-name=vdb1

Mount Point /mnt/vdb1

Identify As /dev/sdb1

Matches the given device

Filesystem Type auto

Cancel OK

- ✓ Enter the "**mount -a**" command to force **OS** to mount all unmounted **FS** defined in the "**/etc/fstab**" file
- ✓ List mounted file systems with the "**df -h**" command to verify the correctness of the operation.
- ✓ Login as "**sysadmin**" and check if you can access the **/mnt/vdb1/work/pw** file.
- ✓ Reboot the system; login as "**sysadmin**";
  - ○ List mounted file systems with the "**df -h**" command.
  - ○ List the **/mnt/vdb1/work/pw** file (icon of the mounted disk should be displayed).

C.3 Automating Mounting of File System /dev/sdb2

Perform similar operations define in section C.2 to mount /**dev/sdb2** on **/mnt/vdb2** and allow its automatic mounting at boot time. Restart **FS-tst** ; logon as "**sysadmin**"; and verify that you can list the files on "**/mnt/sdb2**".

Section D. File and Directory Sharing

In this part you will experiment with **UNIX/Linux** file/directory **sharing** by creating **hard links** to files that are stored in the same **file system**; and **symbolic** (soft) **links** to directories, and to files that are located in different **file systems**.

Restart **FS-tst**; logon as the user “**sysadmin**”; and set your **current working directory** to **“/mnt/vdb2**”.

D.1 File Sharing in the Same File System

- ✓ Rename **“/mnt/vdb2/xx.txt**“ as “**yy**”, create the **hard link** “**yy**” with “**ln /mnt/vdb2/xx.txt yy”**
- ✓ List the contents of **“/mnt/vdb2/xx.txt**“” with the “**cat yy**” command.
- ✓ Display and analyze the type and the **link count** of “**yy**” and **“xx.txt**“ using “**ls –ali** ”.

D.2 Directory Sharing

- ✓ Shorten the path to **“/mnt/vdb1/work**” directory and **rename** it as “**base**” from current directory, create the **symbolic link** with the “**ln -s /mnt/vdb1/work base”** command.
- ✓ List directory contents of **“/mnt/vdb1/work**” using “**ls base**” command.

D.3 File Sharing over Different File Systems

- ✓ Shorten the path to **“/mnt/vdb1/work/pw**” file and **rename** it as “**zz**” in the current directory, create the **symbolic link** “**zz**” with “**ln -s /mnt/vdb1/work/pw zz”** command**.**
- ✓ List the contents of **“/mnt/vdb1/work/pw** using “**cat zz**” command.
- ✓ Analyze the attributes of “**zz**” with “**ls –ali**” command.

Section E. Project Report

Perform the following to prepare your project report:

- ✓ Login as the **root** user’ and make sure that all your file systems are mounted.
- ✓ Run the command: “**blkid -o list > mounted.txt”**
- ✓ Run the commands: “**ls -ali /mnt/vdb1 > shared.txt” and** : “**ls -ali /mnt/vdb1/work >> shared.txt”**
- ✓ Add a brief in the “shared.txt” file explaining the link counts of the files and directories stored in “shared.txt”.
- ✓ Add your name and student in in each file.

Store the files “**mounted.txt**” and “**shared.txt**” in the “**Prj6**” folder, located at the course web site under the tab **CSE5031 - OS Section -X/Assignment**; where “**X**” stands for (1,2,3,4) your laboratory session group.

Warning

You are encouraged to discuss the implementation procedures and general concepts behind the projects with your fellow students. However, **plagiarism is strictly forbidden**! Submitted report should be the result of **your personal work**!

Be advised that you are **accountable** of your submission not only for this project, but also for the mid-term, and final examinations. Your project grade may be reevaluated retrospectively, had you fail to answer correctly the same or a similar examination questions that you have solved with success in your submissions.