

Operating Systems Linux Installation

Florido Paganelli
Lund University
florido.paganelli@hep.lu.se

Outline

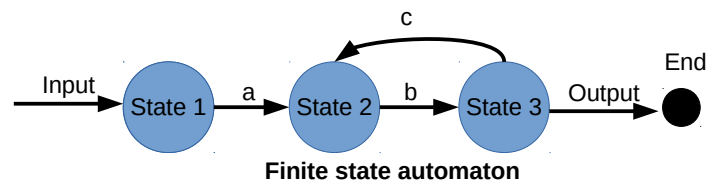
- What is a computer?
 - A computer as a finite state machine
 - Brief history and architecture overview
- Operating Systems
 - Why do we need it?
 - Examples of operating systems
 - Linux and distributions
- What we're going to do today
 - Use of virtualization technology
 - Linux Installation
 - Use of a pre-built virtual machine

What is a computer?

- A **programmable machine** that can store, retrieve and process **data**.
- Most of modern computers are based on electronic circuits.
- The activity of manipulating data is usually called **information processing**.

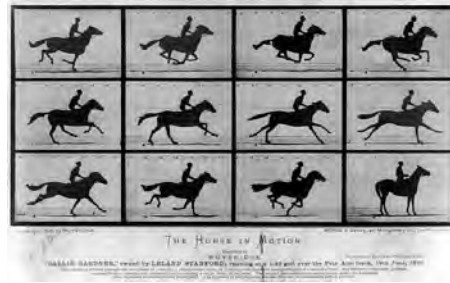
Finite State Machine

- A mathematical object that represents a sequence of events and their possible outcomes
- You can use this model for:
 - Evolution of a closed system
 - Card games
 - Movies cutting
 - Storytelling
 - Music



Finite state machines

- A modern computer is modeled by a finite state machine.
 - A "state" is the contents of "memories" of the machine
 - If we could stop time, the computer would stay in a defined state
- ▶ A state can be restored by restoring the machine's "memories"
 - Examples: hibernation, **virtualization (some about it later)**



States

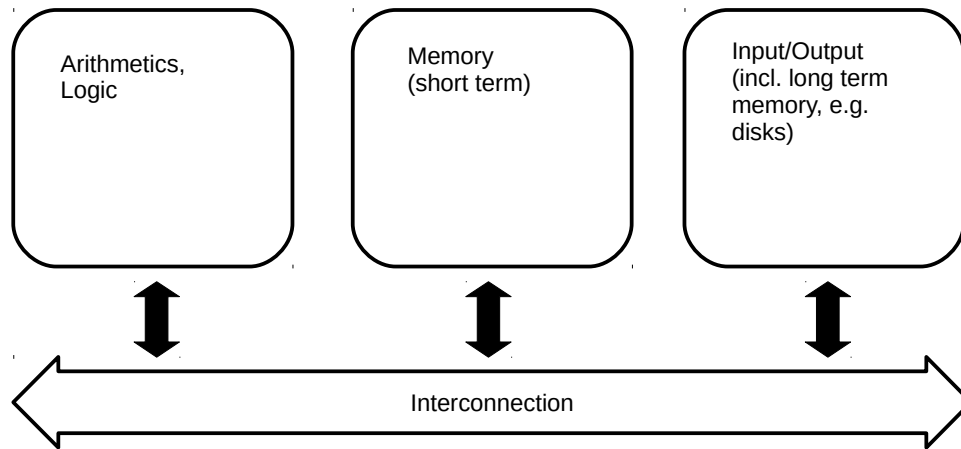
States during execution
(only visible in libreoffice)

VERY Brief history of computing

- 1945 Von Neumann's paper[1] defines the modern computer architecture
- 1960-70 Various researchers start improving the communication between components
- 2014: We still use the same basic approach, with lots of improvements.

[1] First Draft of a Report on the EDVAC, John von Neumann, Contract No. W-670-ORD-4926, June 30, 1945

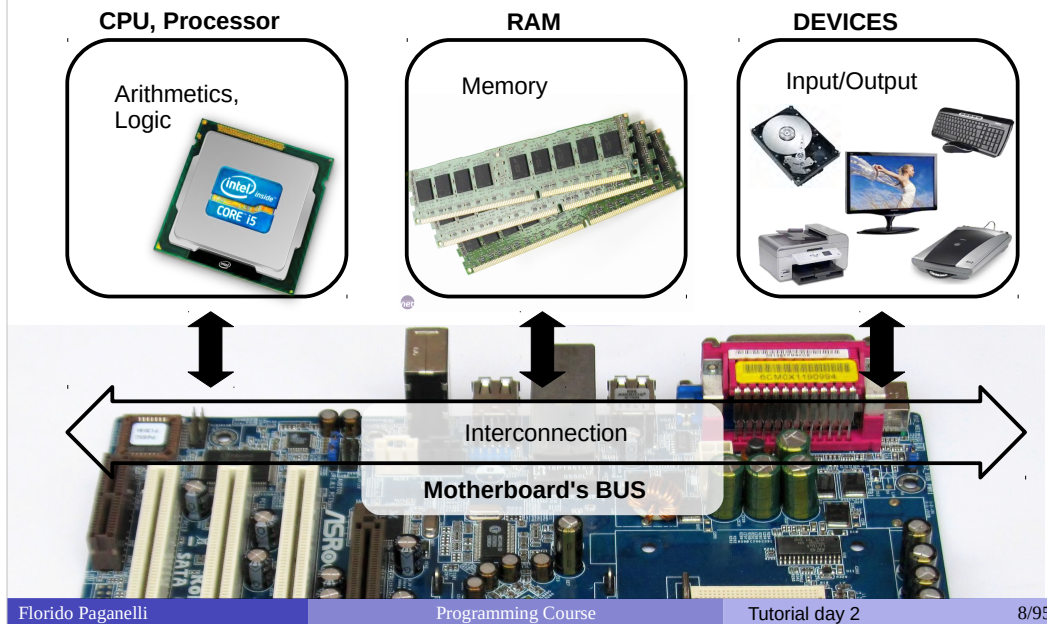
Von Neumann-based modern architecture



- Arithmetics and Logic
- Memory:
 - Short term
 - Long term
- Interconnection
- Interaction with external world

Hardware:

electronic components of a computer



- Arithmetics and Logic: CPU
- Memory:
 - Short term: RAM
 - Long term: discs/USBdongle/cloud
- Interconnection: BUS(PCIE,SATA,USB)
- Interaction with external world: Networking, Screen, Keyboard, Touch screen...

Software

- Anything that is designed to **run** or **execute** in a computer , that is, the information that is manipulated by the hardware.
- Can be of different kinds:
 - **System software:** used to interact directly with the hardware, usually as an interface between the hardware and other kind of software.
Examples: device drivers, operating systems, firmware...



- **User software:** something with which a user interacts directly to perform a task. Also called **Programs** or **Applications** (shortened: **Apps**). It is usually run inside an operating system.



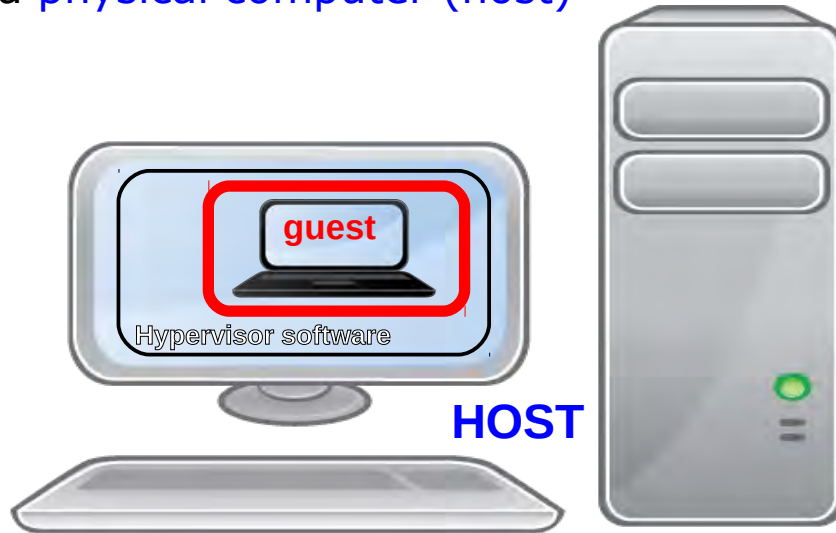
- **Development software:** software that is used to develop and create other software.
Examples: SDK (Software Development Kit), libraries, compilers...

Hardware-Software equivalence

- Everything that can be modelled via software can be created in hardware and vice-versa
- This poses the foundation for machine **simulation** and **emulation**
 - **Simulation**: software that behaves exactly like some piece of hardware, internally and externally. For prototypes and testing
 - **Emulation**: write software that whose external behaviour is like a piece of hardware. The internals can differ. It "pretends" to be some hardware.

Virtualization

- Running a **virtual computer (guest)** inside a **physical computer (host)**



- Running a **computer (guest)** inside a computer (**host**)
- The **guest** machine is usually called **Virtual Machine**.
- The **Host** machine manages the guest machine using something called **Hypervisor**
- The **host** offers software simulated or emulated hardware, plus it can offer **real** hardware to the guest machine
- The **guest** machine sees all the software simulated/emulated/virtualized hardware as it was real hardware, but it can also be aware that it is virtualized to boost performance

The operating system



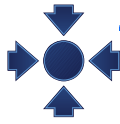
- Is a **collection of programs** running in your computer all the time it is turned on.
- **Orchestrates** the interaction between all components of a computer.
- Usually allows the machine to run **multiple programs** at the same time (multitasking).
- It is meant to bring the machine closer to the **user**.
- It is usually **installed** on a **long-term storage memory** like a CD, a USB pen, and HardDisk or a ROM (for example in mobile phones)

Examples of operating systems

- Microsoft Windows
- Mac OS X
- UNIX
- GNU/Linux
- Android
- Symbian
- ...



Basics of operating systems



- **Kernel:** a program more important than the others. It's a **software** that is the **core** of an operating system.



- **Drivers or Modules:** set of programs that allow the kernel to interact directly with the **hardware**



- **User interfaces:** set of programs allowing the users to communicate with the computer and use **software**.

GNU/Linux or Linux

- One of the **full community based** accessible operating system, based on UNIX (proprietary)
- GNU (GNU's not Unix!) project: community that wrote most of the **drivers, basic tools and user interfaces**.
Founder Richard Stallman (father of Free Software)
- Linux: the operating system whose **kernel** was developed by Linus Thorwalds

Linux, why we're using it

- Popular **free (as in free beer)** alternative to many proprietary operating systems
- **Free access (as free software)** to source code: anybody has **right to**
 - **Run the programs**
 - **Read and modify the programs**
 - **Redistribute modifications to friends**
 - **Propose changes to the community for everybody to benefit**
- Components developed by many universities (e.g. MIT) on a free-for-all knowledge basis
- Accessible libraries to build software upon
 - most of scientific software is written on it
- After 20 years, it still "scales" better than others on cheap hardware – used for intensive computation
 - Many universities install it on workstations, clusters, servers...
- Huge community based effort to keep it up to date and to keep it accessible for everybody

What is a Linux Distribution

- A **selection of software** that runs on top of a GNU/Linux operating system. This includes:
 - Installation tools (to install the operating system in a computer)
 - Software installation programs (to add new software to a computer)
 - **Configuration** and **management** software
 - Graphical interfaces
 - Office programs alternatives
 - Development tools
 - Communication software
 - Scientific software
 - Videogames
 -
- Bound to a place on the internet where such software is available, often called **software packages repository** or in short **repository**.



Popular Linux distributions

- **Debian**, community based   **ubuntu**
 - **Ubuntu, Kubuntu, Lubuntu** based on Debian, commercially maintained by Canonical
 - Derivatives: Linux Mint ...
 - **Arch Linux**: community based 
 - **RedHat**: first commercial open source Linux
 - Derivatives: Mandrake Linux, ...
 - **CentOS**, community based   
 - **Scientific Linux**, developed at CERN
 - **Novell SuSE**: commercial Linux
 - **OpenSuSE**: community based SuSE 
- Lots more: <http://distrowatch.com/>

Lubuntu: the one we will use

- Best user community effort backed up by the Canonical company, although with lots of criticism by other communities
- Probably the easiest for you to try at home
- L stands for lightweight, that means that we doesn't require much power.



Linux installation: basic concepts



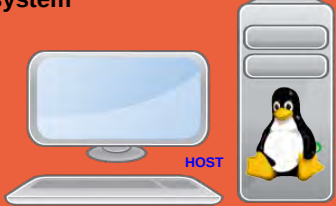


- The operating system usually comes in a **DVD** image, that one can put on a DVD or on a USB pen. The image contains also an **installer software**.

- The operating system is usually **installed on an hard disk**.



- Popular operating systems like Windows and OSX **do not like to share the hard disk** with other systems: this poses installation problems.
- The Linux community has found many solutions to this limitation, in order for Linux to be installed together with other systems.

Linux installation: three strategies

<p>Requires to be administrator</p> <p>1. Linux as the <i>only</i> operating system</p>  <p>3. Dual boot with GRUB2: two systems installed, but only <i>one</i> runs at every restart</p> 	<p>Administrator must install Virtualization software</p> <p>2. Virtual Machine: operating systems running <i>at the same time</i></p> 
--	---

Florida Paganelli Programming Course Tutorial day 2 21/95

- 1 Single operating system:** Linux deletes everything on the disk and it becomes the one and only operating system for that computer
 - Constraint: user must own the computer (administrator)

- 2 Virtual Machine:** Run Linux in a virtual machine where it is the one and only system. The virtual machine is run on an hypervisor that is running in the existing OS.
 - Constraint: User must be able to **install software**, i.e. VirtualBox hypervisor

- 3 Dual boot:** Linux shares the disk with other operating systems. Requires an alternative boot loader (e.g. GRUB2), a small program that is loaded **BEFORE** any other installed system. This program must be installed at the beginning of the disk.
 - Constraint: user must own the computer (administrator)

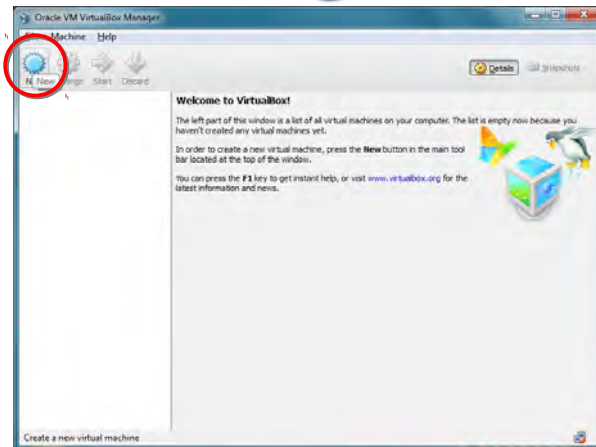
Today's tutorial

1. Creation of a Virtual Machine (VM)
2. Starting and Stopping a VM
3. Download of Lubuntu Linux
4. Installation of Lubuntu Linux
5. First steps with Lubuntu, moving around the desktop
6. Installing software from repositories
7. Power off and Restart, Software update
8. Customizing the work environment
9. Repositories and other installation methods
10. Installation of the Lubuntu prebuilt custom machine for the course

1. Create a virtual machine

Goal: create a virtual machine to run
Lubuntu Linux.

1.1. Open virtualbox  . Click on "New"



Follow the instructions in

<https://www.virtualbox.org/manual/ch01.html#gui-createvm>

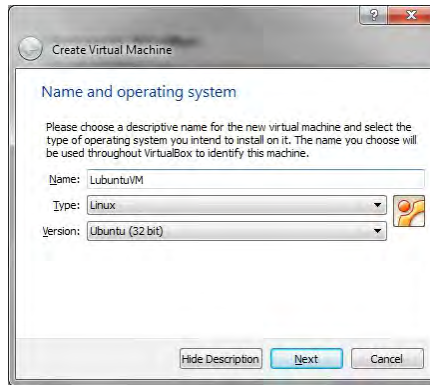
using the following information:

VM Name	LubuntuVM
Operating System Type	Linux
Version	Ubuntu (32 bit)
memory (RAM)	1536 MB (that is, 1,5 Gigabytes)

When you reach step 4, go to slide 23
(virtual disk creation).

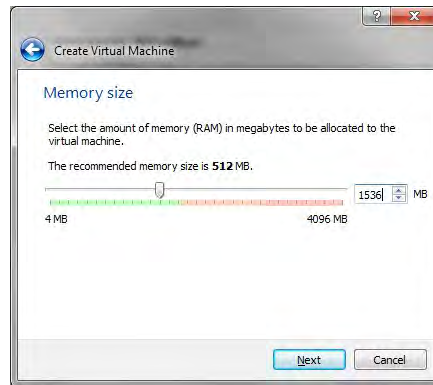
1. Create a virtual machine

1.2. Insert the following information when asked:



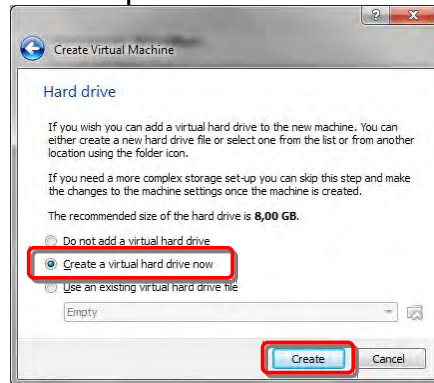
1. Create a virtual machine

1.3. Set memory size to 1536 MB:



1. Create a virtual machine – virtual disk

1.4 Create a **new virtual harddisk** for the machine, click create as shown in the picture.



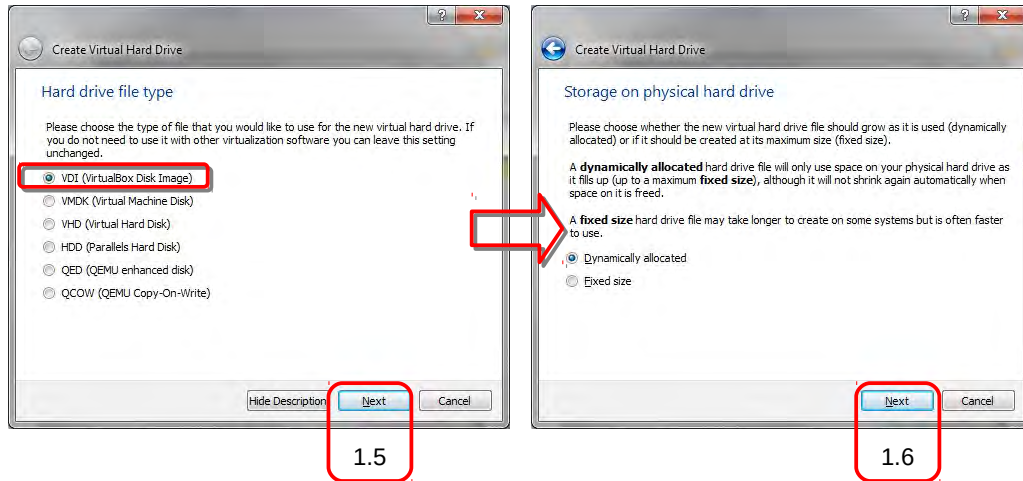
- Create a **new virtual harddisk** for the machine (step 4 in <https://www.virtualbox.org/manual/ch01.html#gui-createmv>)
- When asked, choose "Create a virtual hard drive now" and click "Create"
- Use the following parameters:

Disk Type	VDI (VirtualBox Disk Image)
Storage on physical hard drive	Dynamically allocated
Name	LubuntuLinuxVirtual
Size	30,00 GB

1. Create a virtual machine – virtual disk

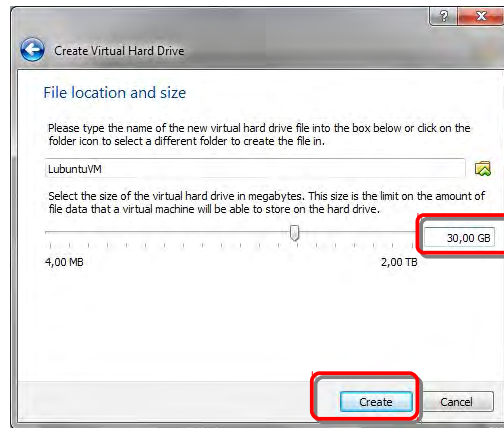
1.5. Select VDI as disk type

1.6. Choose "Dynamically allocated" (saves disk space)



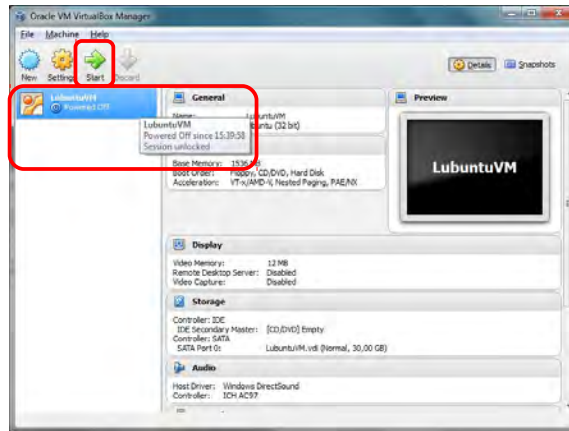
1. Create a virtual machine

1.7. Set the disk size as shown in the picture and create the disk:

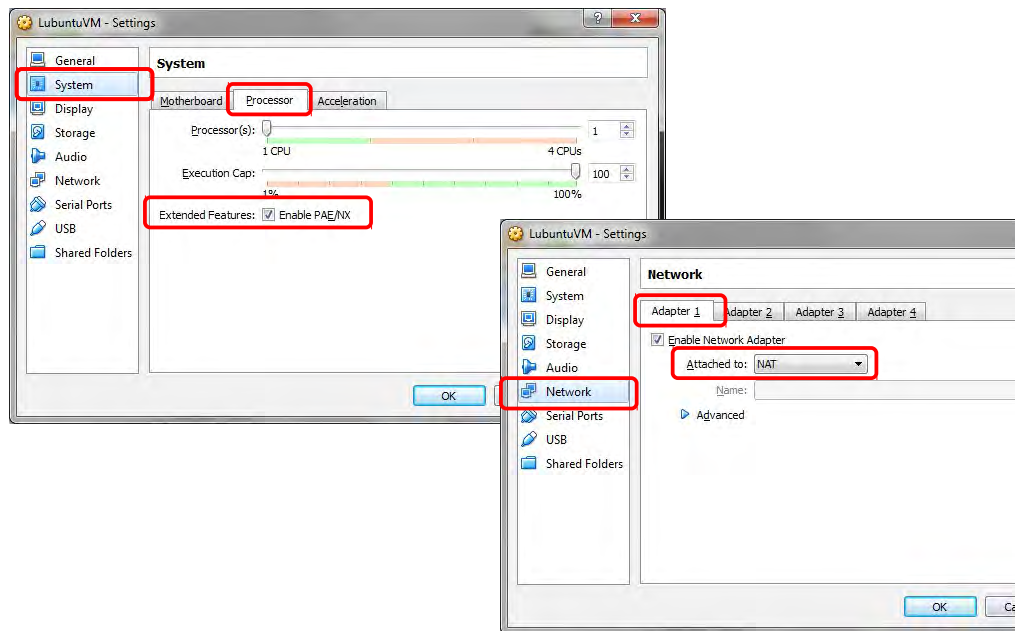


2. Create a virtual machine - start it!

1.8. Select and Start the virtual machine. What happens? Discuss with the teacher.



Enable PAE and NAT



3. Download Lubuntu

3.1. Go to

<https://help.ubuntu.com/community/Lubuntu/GetLubuntu/14.04.1>

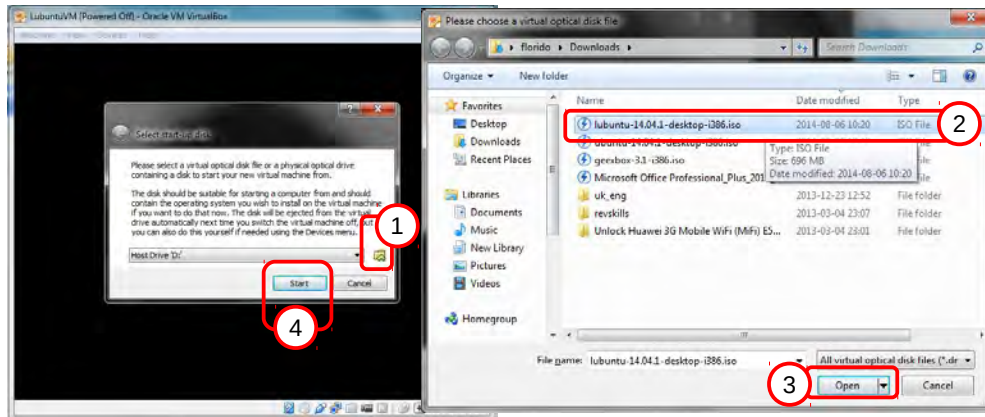
3.2. Click on the link "PC 32bit"

<http://cdimage.ubuntu.com/lubuntu/releases/14.04.1/release/lubuntu-14.04.1-desktop-i386.iso>

3.3. Save the file on the desktop when the dialog appears.

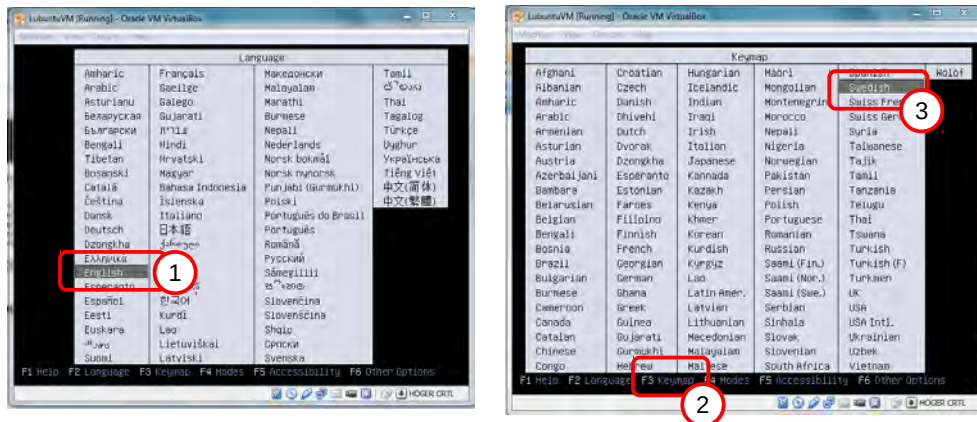
4. Install Ubuntu in UbuntuVM

4.1 Add the ISO file just downloaded to UbuntuVM and Start the virtual machine:



4. Install Lubuntu in LubuntuVM

4.2. Select English as installation language, press F3 and change keyboard to Swedish.

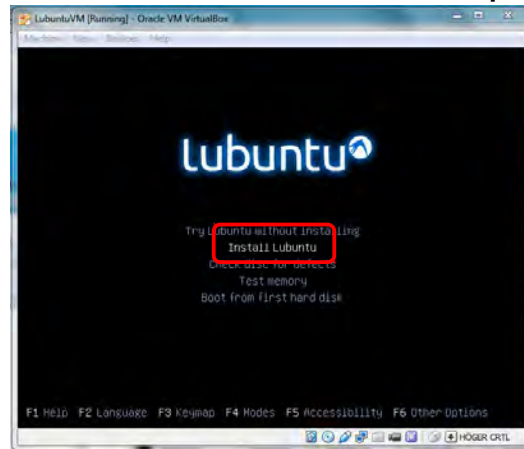


- Start the virtual machine with the Start button with the arrow.
- At the lubuntu prompt:
 - Choose "English"
 - Press F3. Select the Swedish keyboard.
 - Select "Install lubuntu"
- Follow the on-screen instructions. When prompted, insert the informations/choices in the following table:

Installation dialog	Click continue button twice
Installation type dialog	Select "Erase disk and install ubuntu" and click continue
User name	courseuser
Computer name	Lubuntu-VirtualBox
User name	courseuser
password	coursepassword (insert twice)

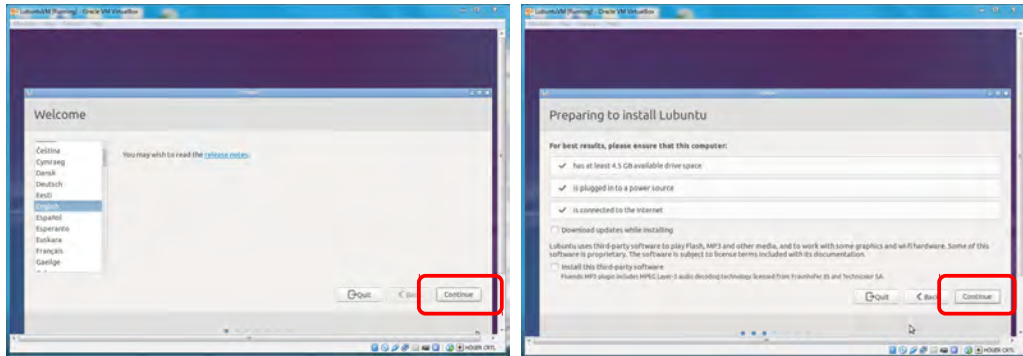
4. Install Lubuntu in LubuntuVM

4.3. Select Install Lubuntu and press enter.



4. Install Lubuntu in LubuntuVM

4.4. Click on continue twice

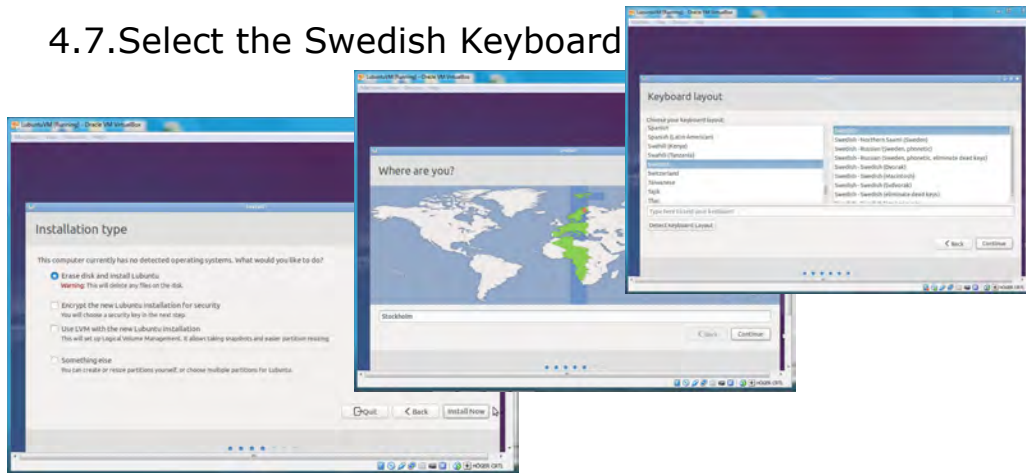


4. Install Ubuntu in UbuntuVM

4.5. Choose installation type "Erase disk and install ubuntu"

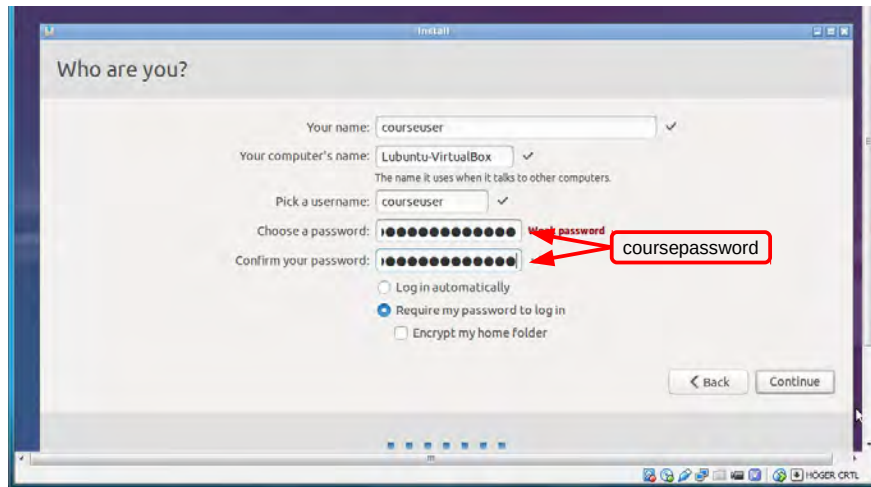
4.6. Select the Stockholm timezone

4.7. Select the Swedish Keyboard



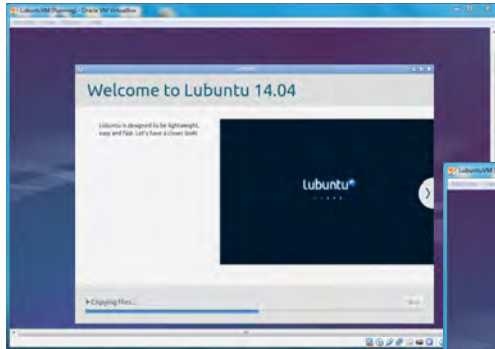
4. Install Ubuntu in UbuntuVM

4.8. Insert computer name and username/password as in the picture:

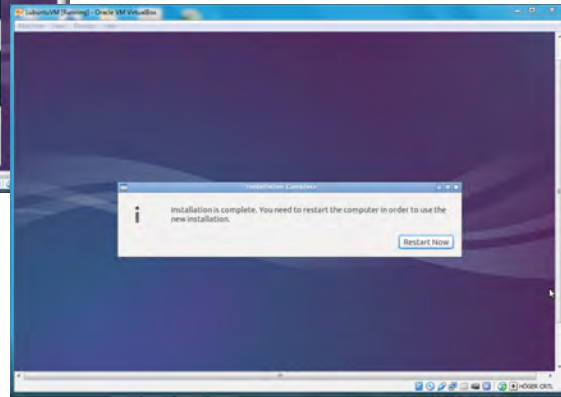


4. Install Lubuntu in LubuntuVM

4.9. Let's wait for installation to complete!

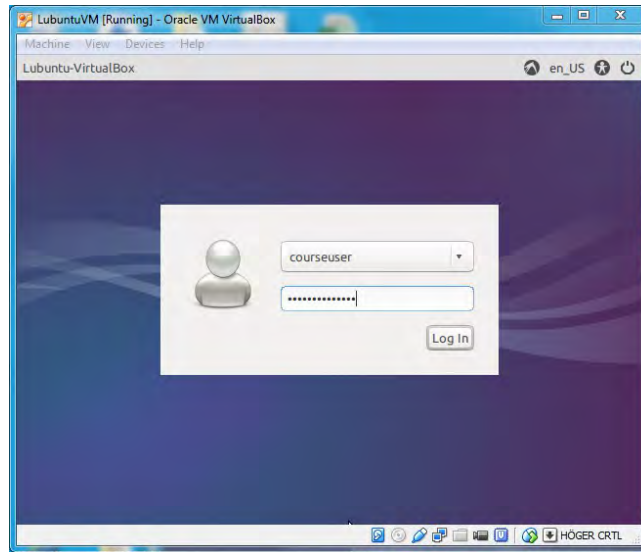


In the meantime: How does a computer work? Click!



5. First steps in lubuntu usage

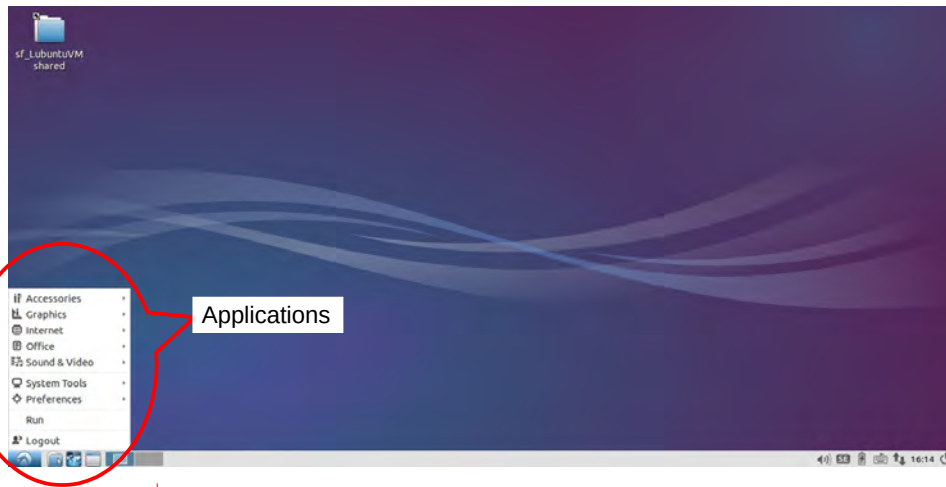
Login using the user name



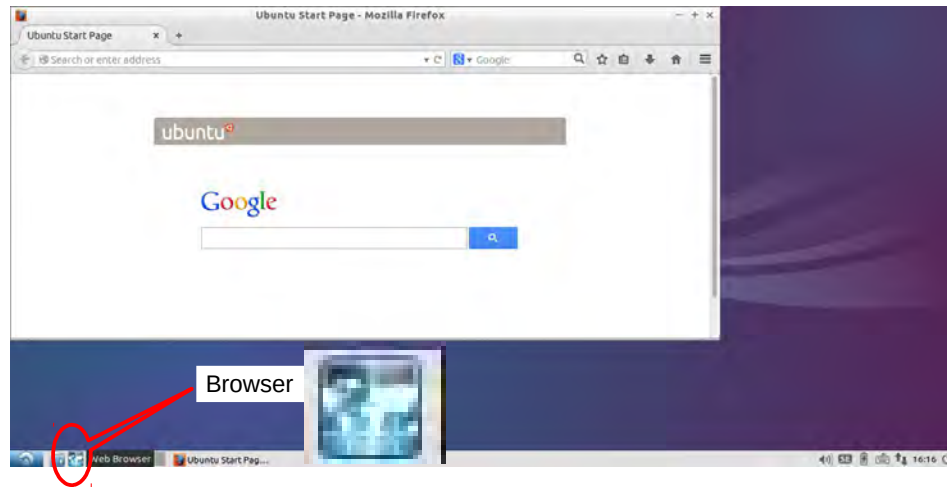
5.1 Moving around the desktop



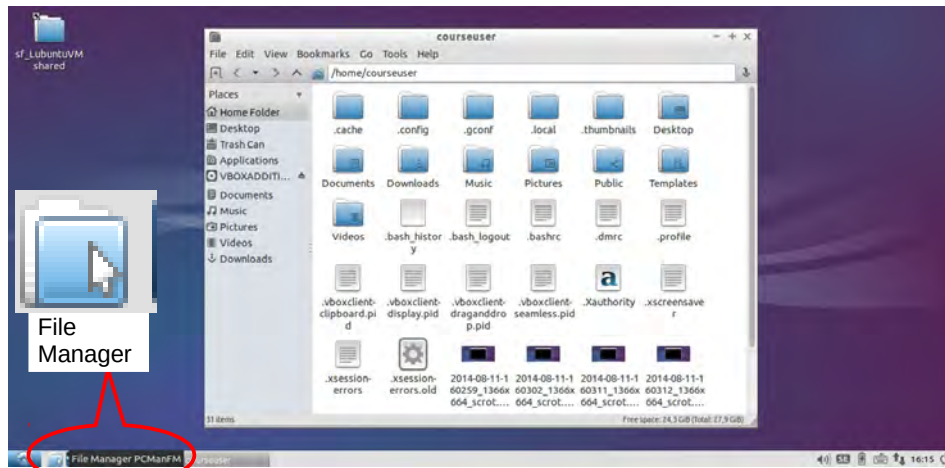
5.1 Moving around the desktop



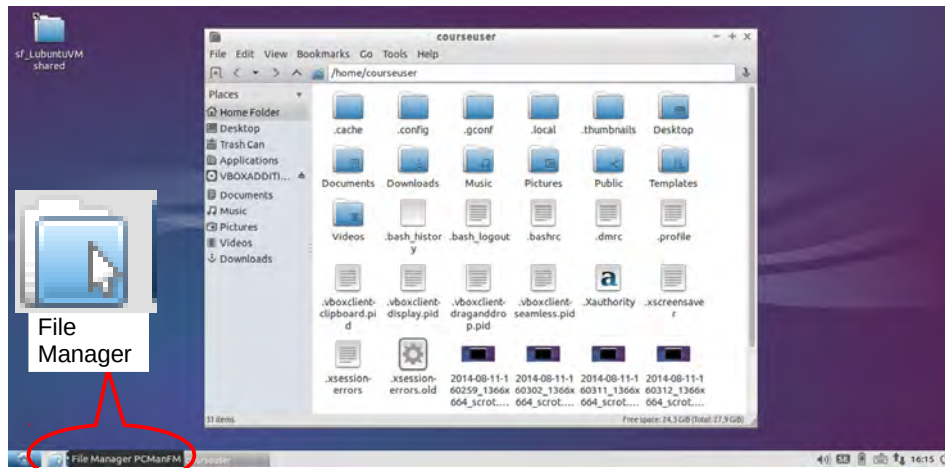
5.1 Moving around the desktop



5.2 Moving around the desktop

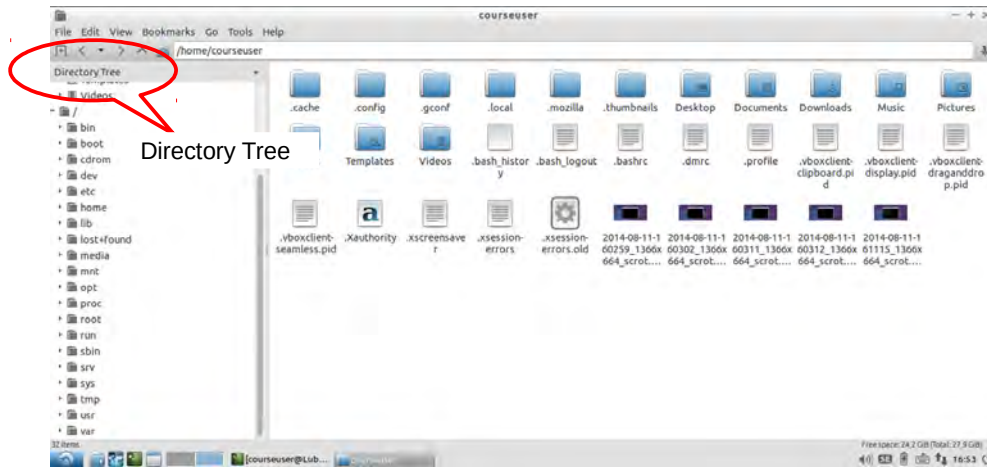


5.2 Moving around the desktop



5.2 Moving around the desktop

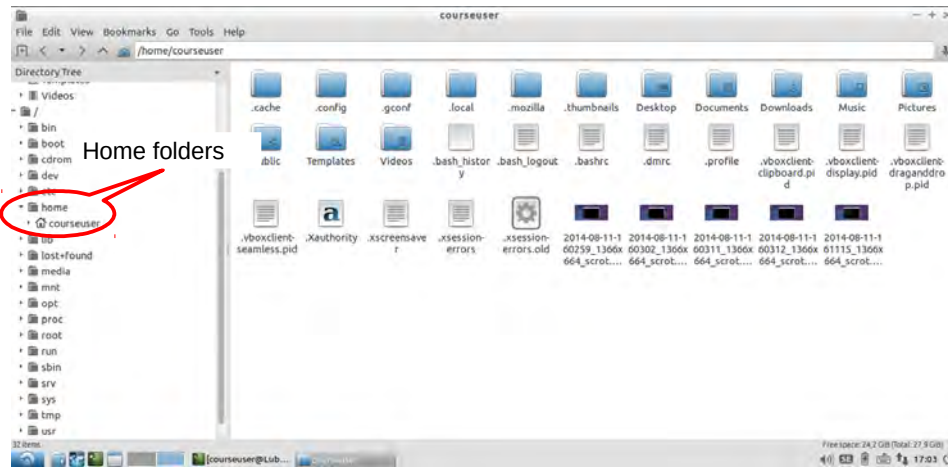
Organization of files in Linux



the filesystem "root": /
Home directory: /home/courseuser

5.2 Moving around the desktop

Organization of files in Linux



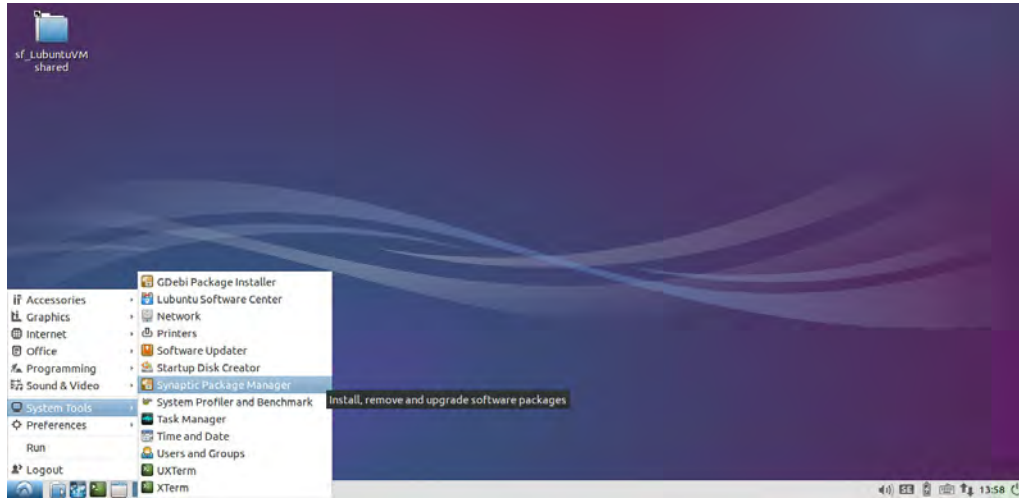
the filesystem "root": /
Home directory: /home/courseuser

6. Installing software Enhancing the virtual machine

- Did you notice how small is the screen? This is because **video drivers** for the **virtual video card** are missing.
- Ubuntu is aware it is running in a virtual machine, but needs to know **how to access the virtual hardware**.
- We will install the so-called **guest additions**

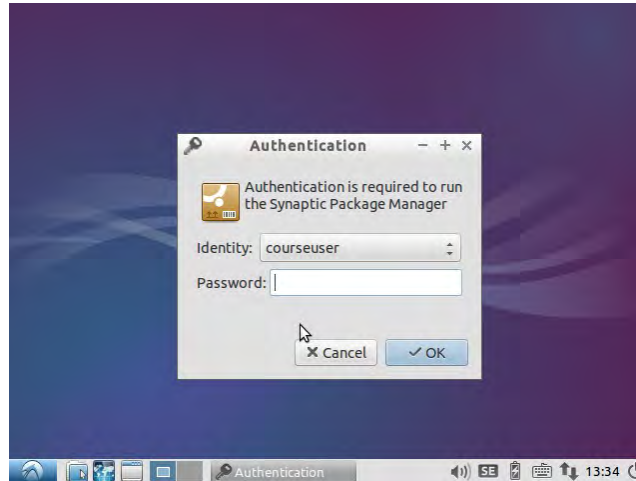
6. Installing software from repositories

- Scientist: Synaptic, search for libraries, packages



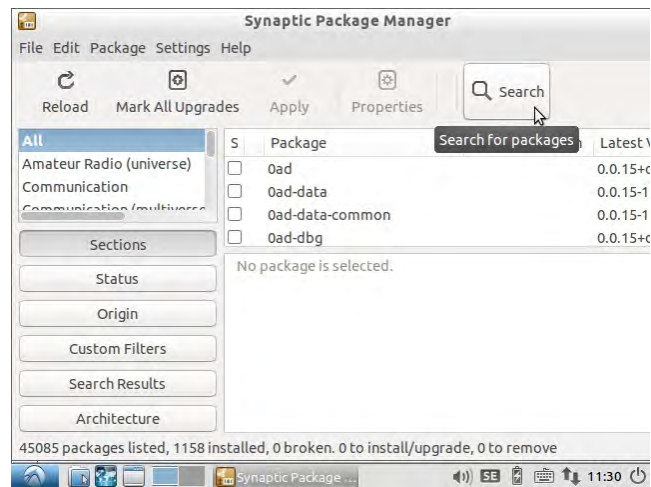
6. Installing software from repositories

- Adding software requires superuser privileges



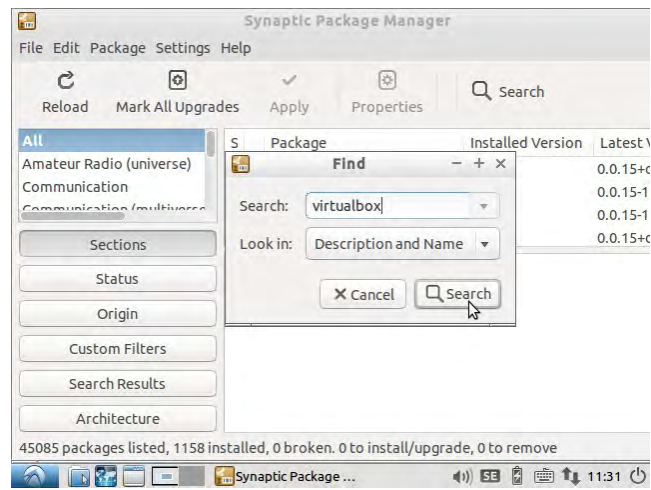
6. Installing software from repositories

Installing guest additions



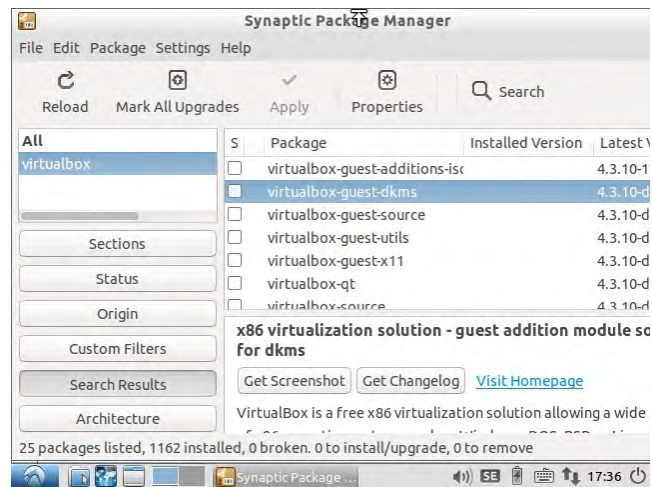
6. Installing software from repositories

Installing guest additions



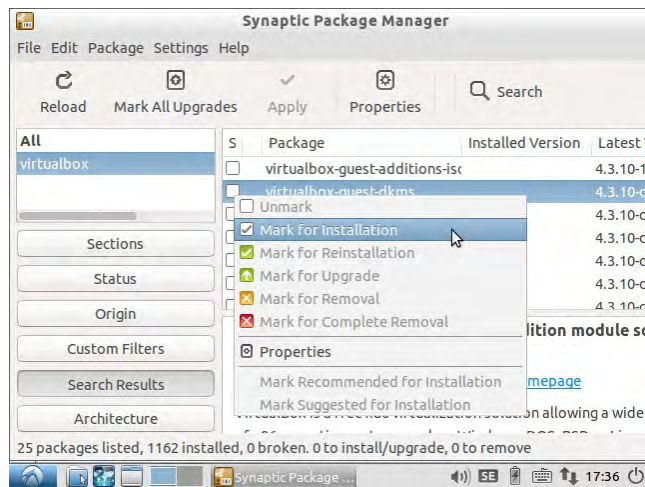
6. Installing software from repositories

Installing guest additions



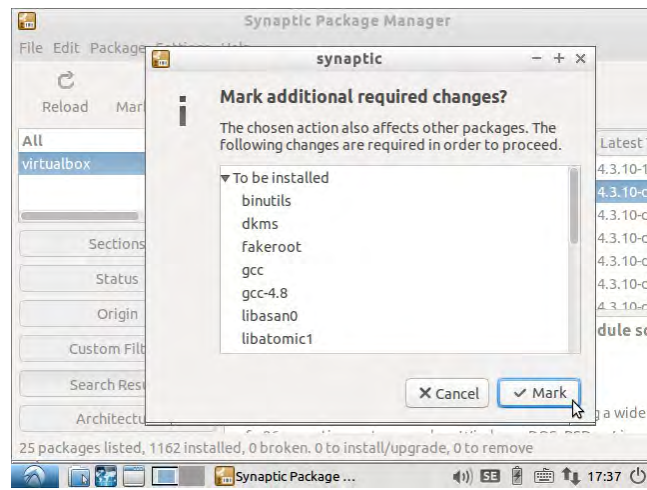
6. Installing software from repositories

Installing guest additions



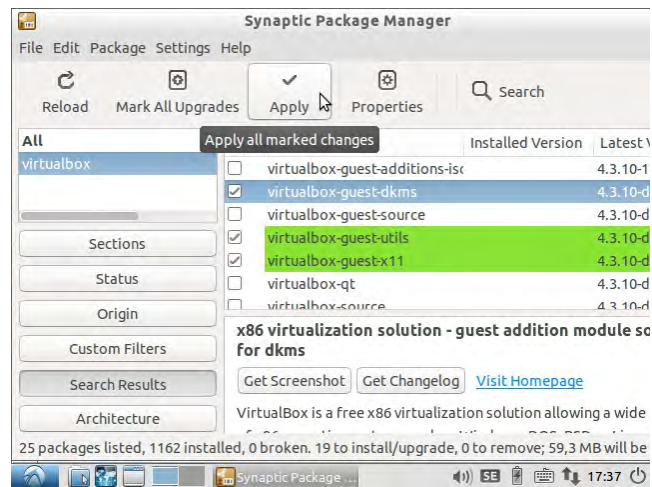
6. Installing software from repositories

Installing guest additions



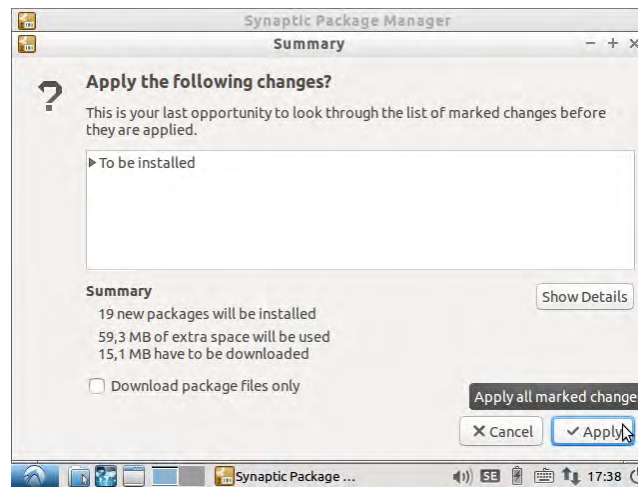
6. Installing software from repositories

Installing guest additions



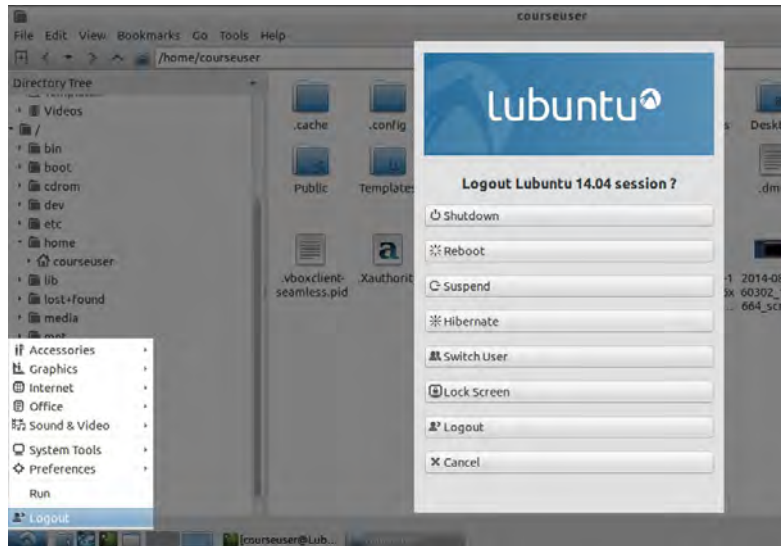
6. Installing software from repositories

Installing guest additions

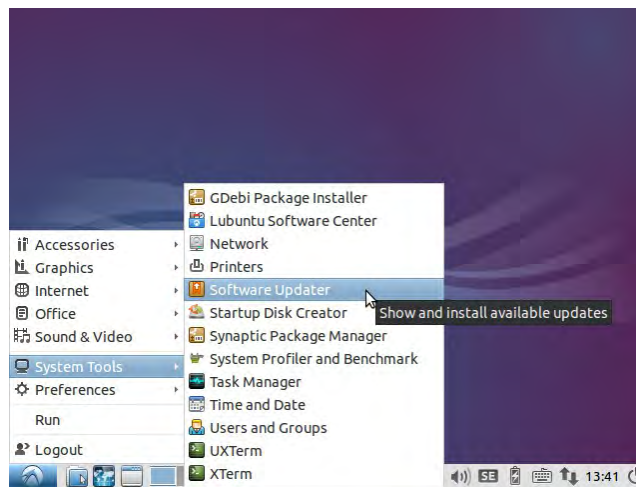


7.1 Poweroff and restart

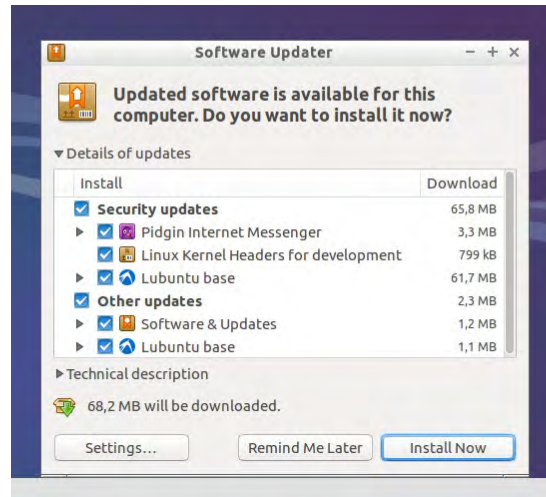
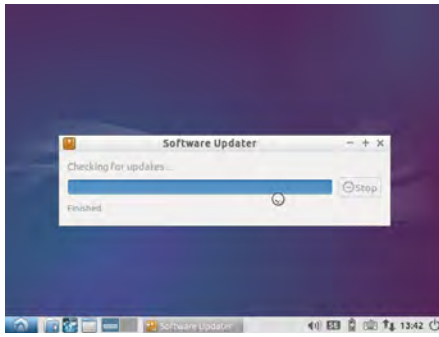
Poweroff the machine



7.2 Software update

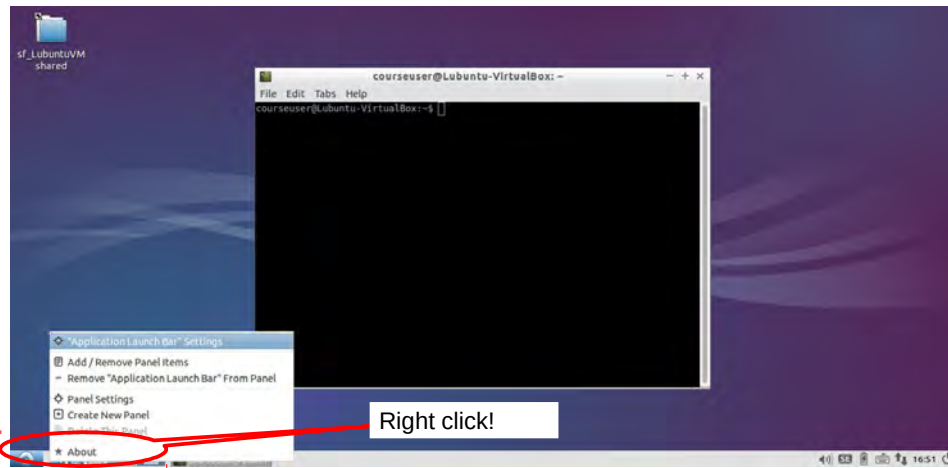


7.2 Software update



8. Customizing the desktop

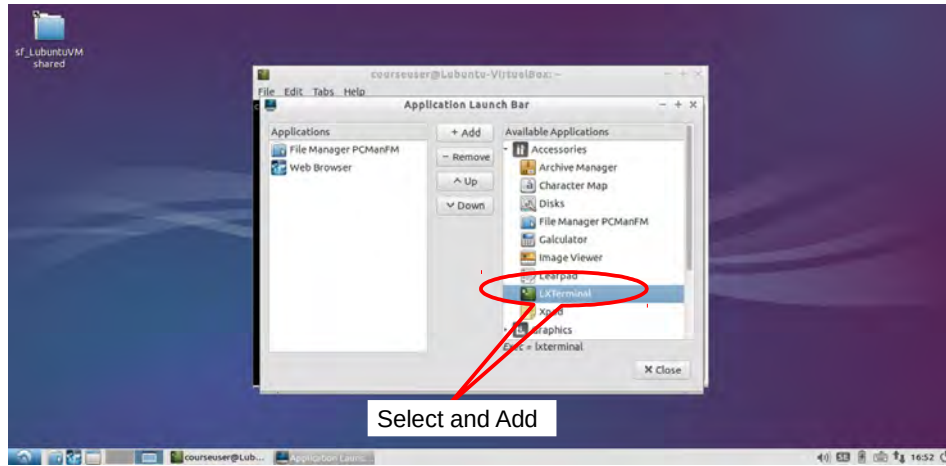
Customization of the desktop: application shortcuts



Click on "Application Launch Bar" settings

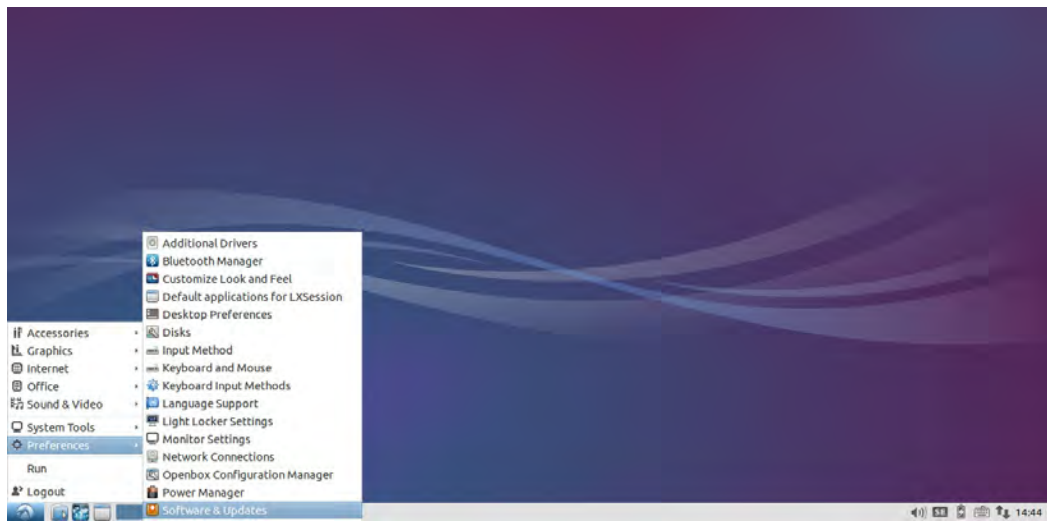
8. Customizing the desktop

Customization of the desktop: application shortcuts



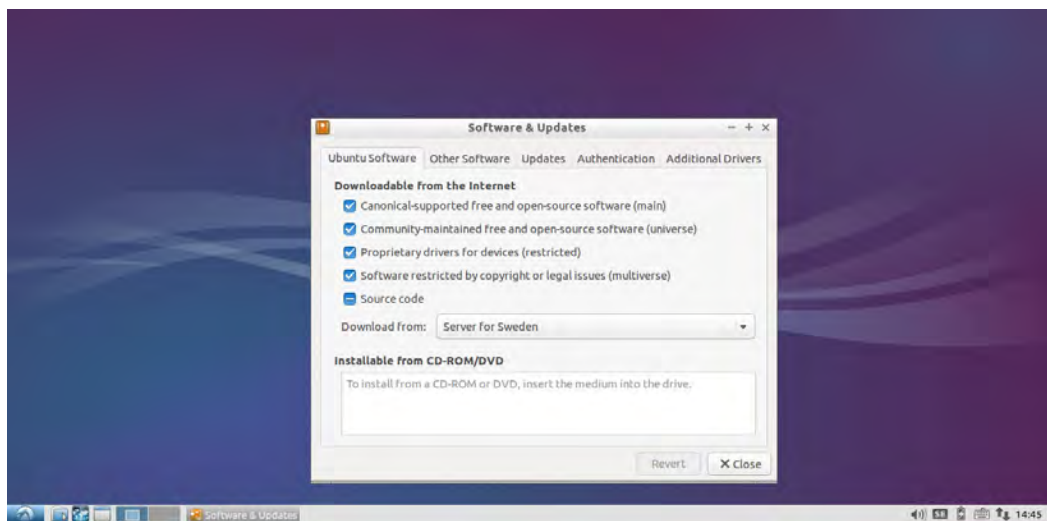
Add LXTerminal to Launchbar

9.1 Software Repositories

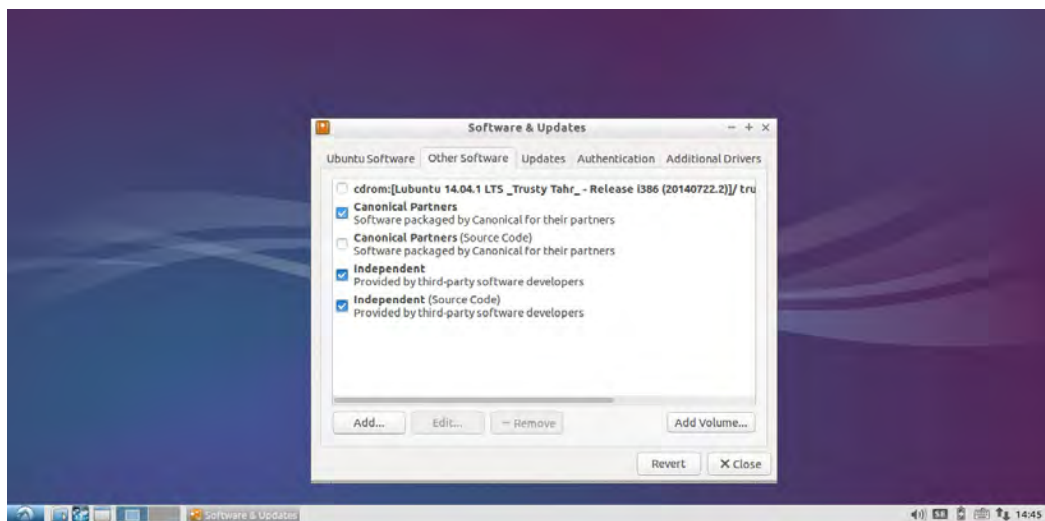


[Click here for the slide about repositories](#)

9.1 Software Repositories

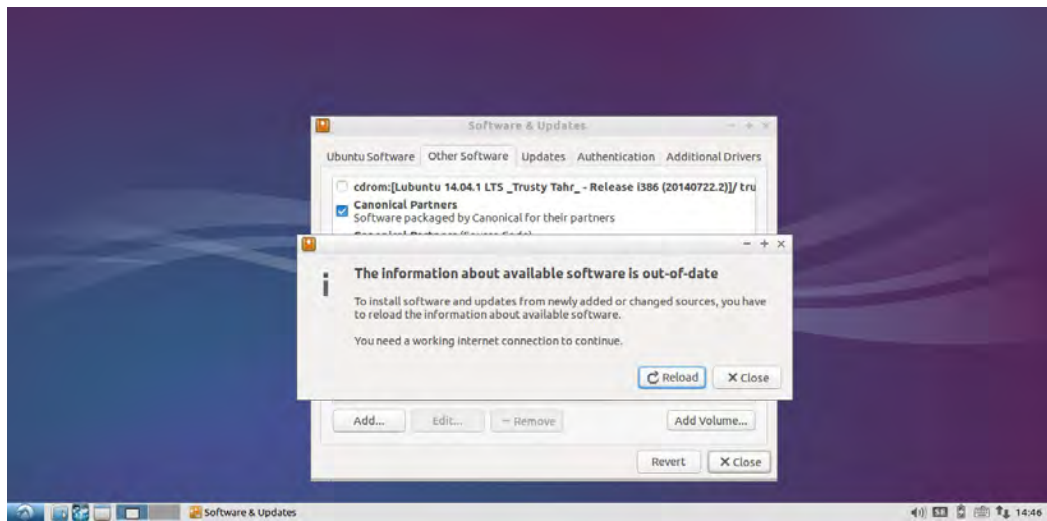


9.1 Software Repositories

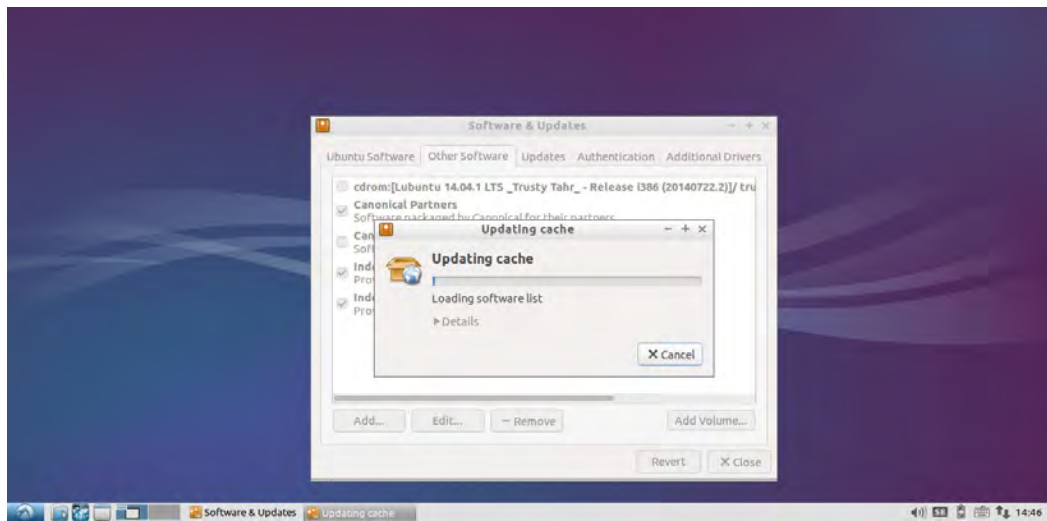


Tick "Canonical Partners". You might need to enter the password.

9.1 Software Repositories

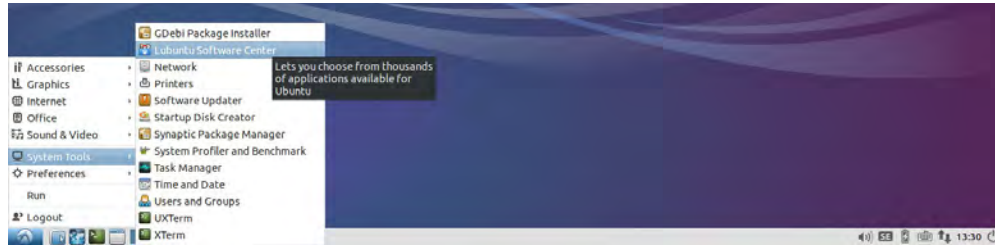


9.1 Software Repositories



9.2 Other installation methods

- Normal user: Lubuntu Software Center



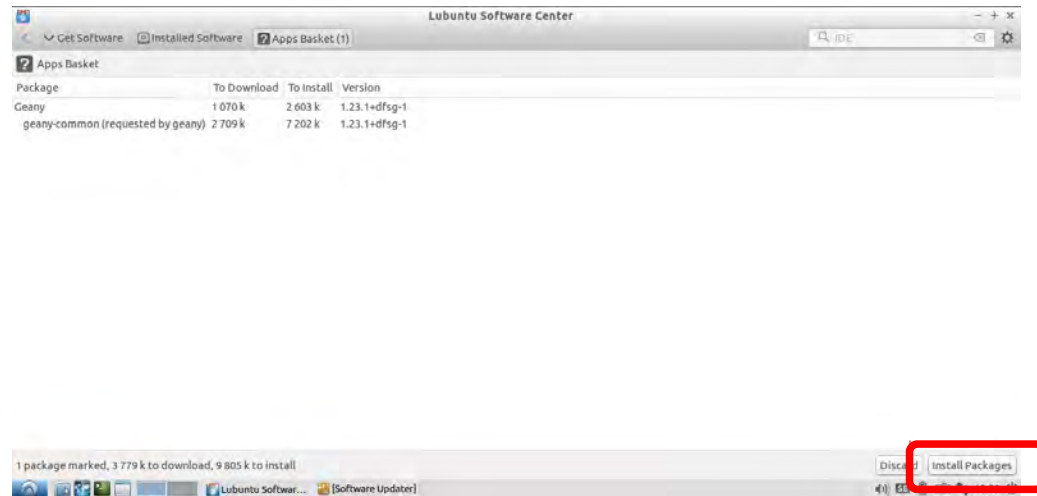
9.2 Other installation methods

- Installing Geany editor



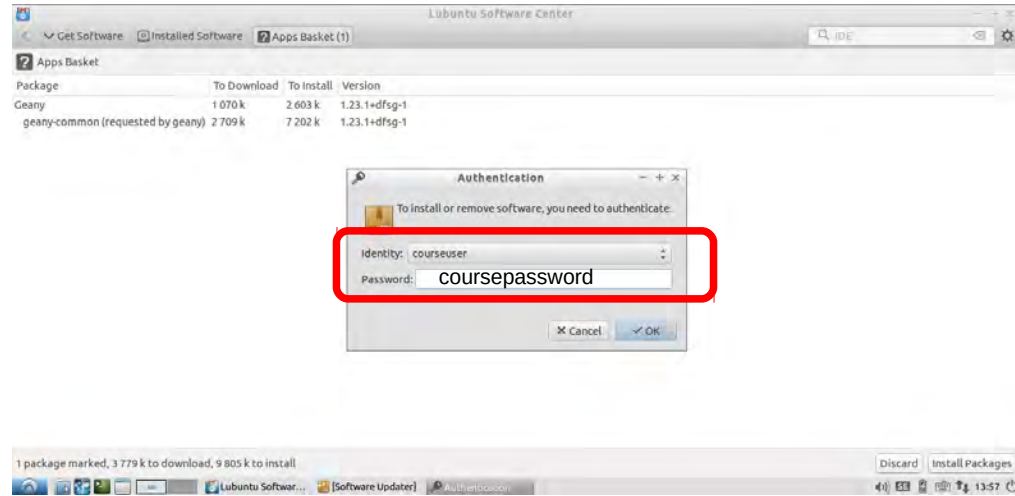
9.2 Other installation methods

- Installing Geany editor



9.2 Other installation methods

- Installing Geany editor



9.2 Other installation methods

- **CLI** (Command Line Interface)

You'll see a short example tomorrow. But here's a list of common commands:

- Search for a package:
`apt-cache search <something>`
- Install a package:
`apt-get install <packagename>`
- Remove a package:
`apt-get remove <packagename>`
- Update package list:
`apt-get update`
- Software update:
`apt-get upgrade`

9.2 Other installation methods

- **Compile** and **install** your own software.
- **Compile:** You will see this during the course.
- **Install:** copy files somewhere
- It **does not require administrative privileges**, so you can do everywhere!

10. Install the course custom VM

10.1. For better usage, the teacher set up a fine-tuned machine for the course.*

10.2. Download it from/copy link:

<http://www.hep.lu.se/staff/paganelli/files/LubuntuVM.zip>

10.3. Extract it into:

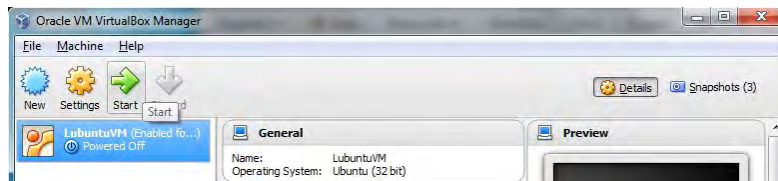
C:\Users\\VirtualBox VM\LubuntuVMCourse\

10.4. Open the machine with VirtualBox:

Machine → Add...

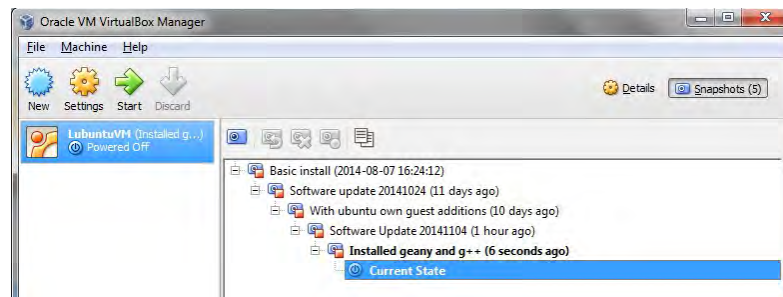
10.5. **Enable PAE if not enabled**

* For more information contact the teacher privately.



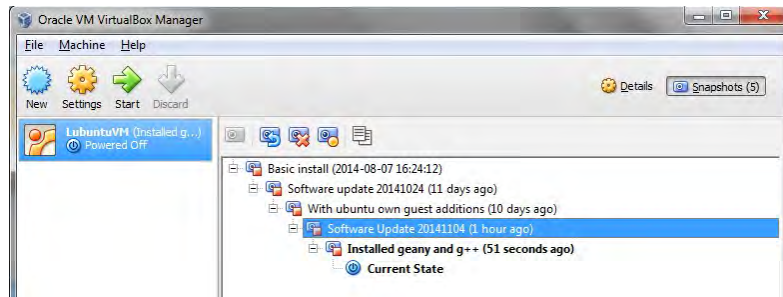
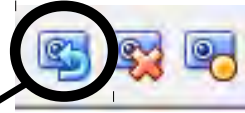
10. Snapshots in the custom VM

- A **snapshot** is the state of the virtual machine in a defined point in time. (See slide about states)
- Snapshots can be created, activated, or deleted by using the highlighted button
- Once a snapshot is set as starting point, the machine execution will start from that point.
- Try to save the current state by pressing the create snapshot button!
- For various technical reasons, the best is to snapshot when the VM is shutdown.



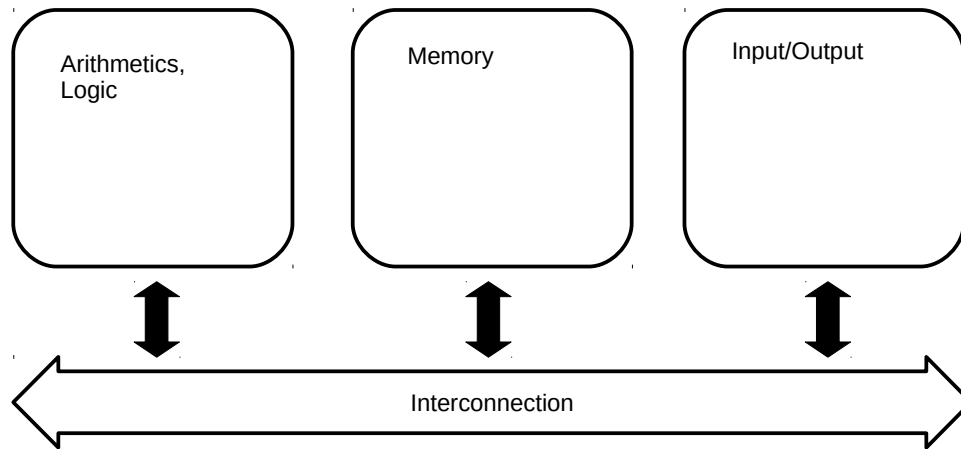
10. Snapshots in the custom VM

- Using the buttons, one can move the machine back and forth in time.
- Let's try to restore a state!
 - Select a snapshot
 - Press the restore snapshot button



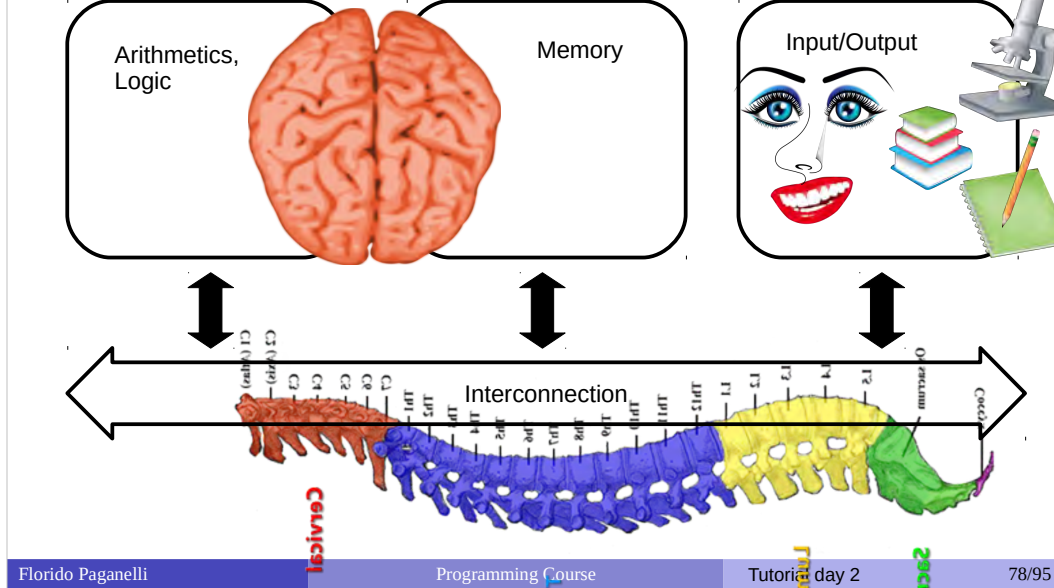
Backup slides

Von Neumann-based modern architecture



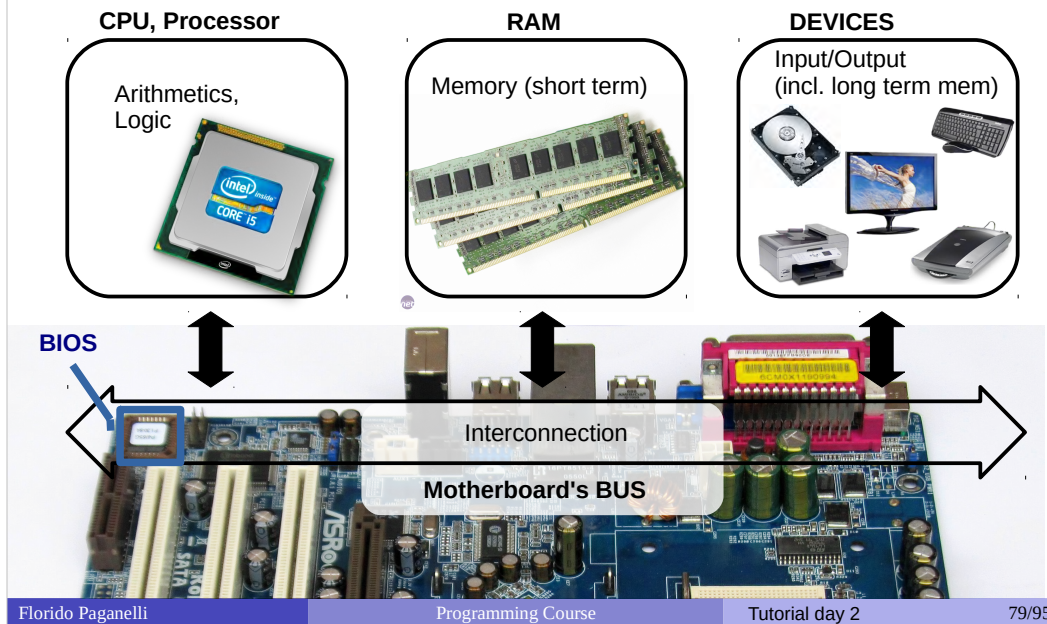
- Arithmetics and Logic
- Memory:
 - Short term
 - Long term
- Interconnection
- Interaction with external world

Von Neumann-based modern architecture



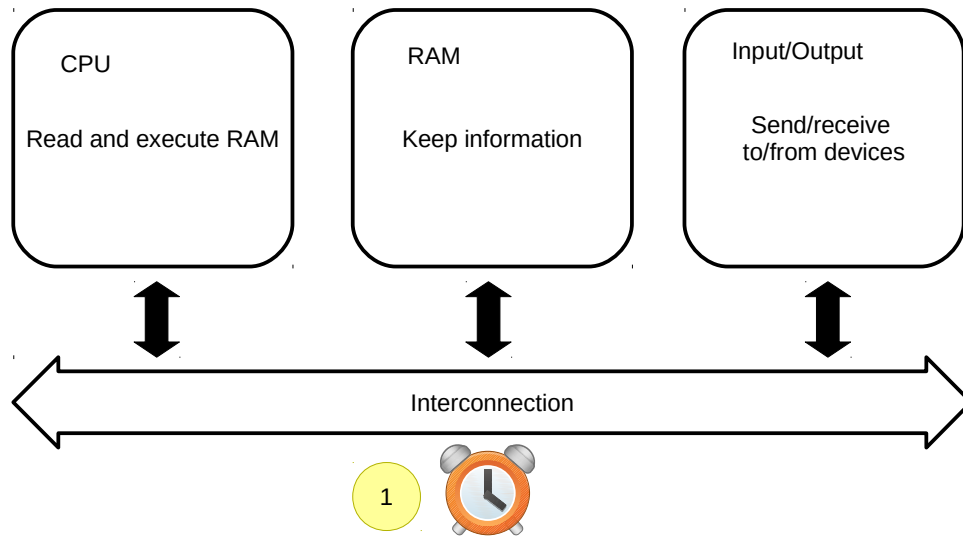
We can make a parallel with the biologic components of a human and senses

Von Neumann-based modern architecture

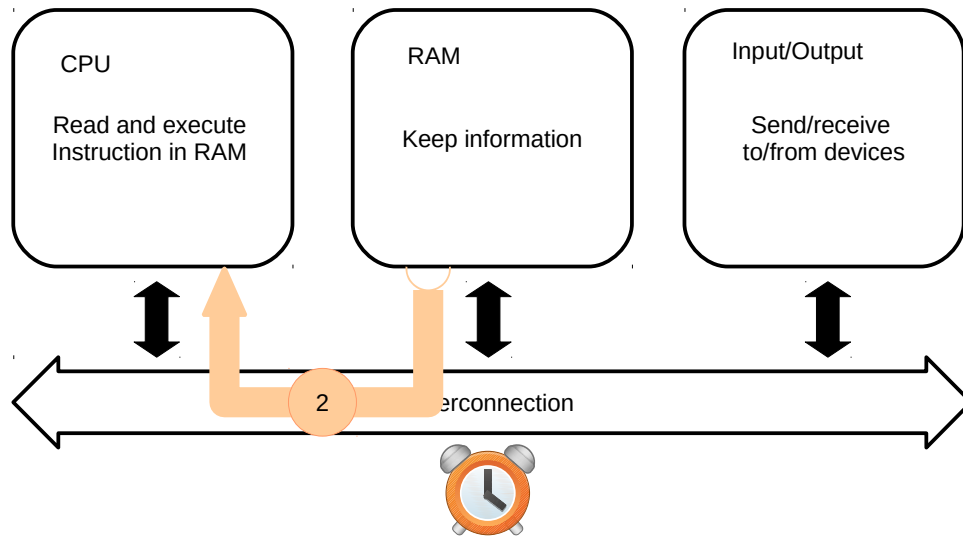


- Arithmetics and Logic: CPU
- Memory:
 - Short term: RAM
 - Long term: discs/USBdongle/cloud
- Interconnection: BUS(PCIE,SATA,USB)
- Interaction with external world: Networking, Screen, Keyboard, Touch screen...

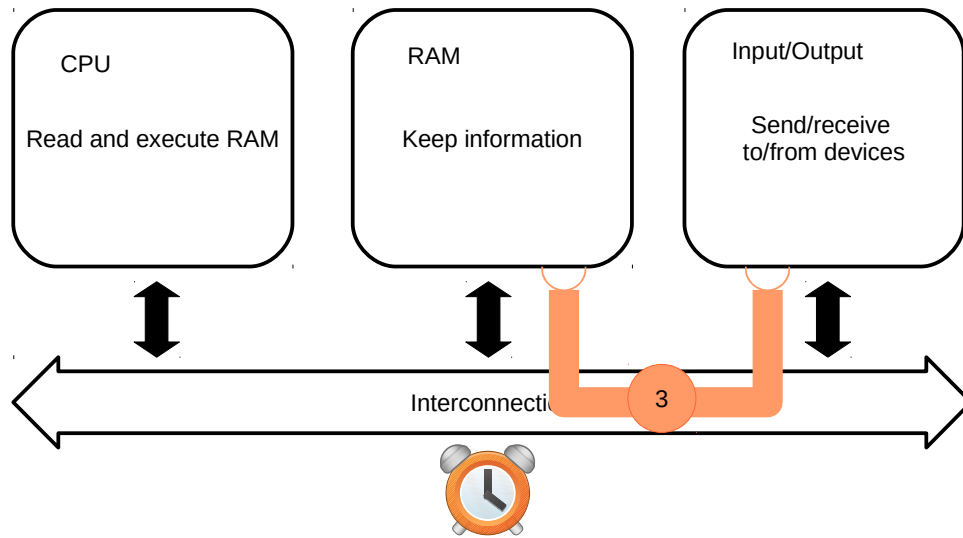
How does it work?



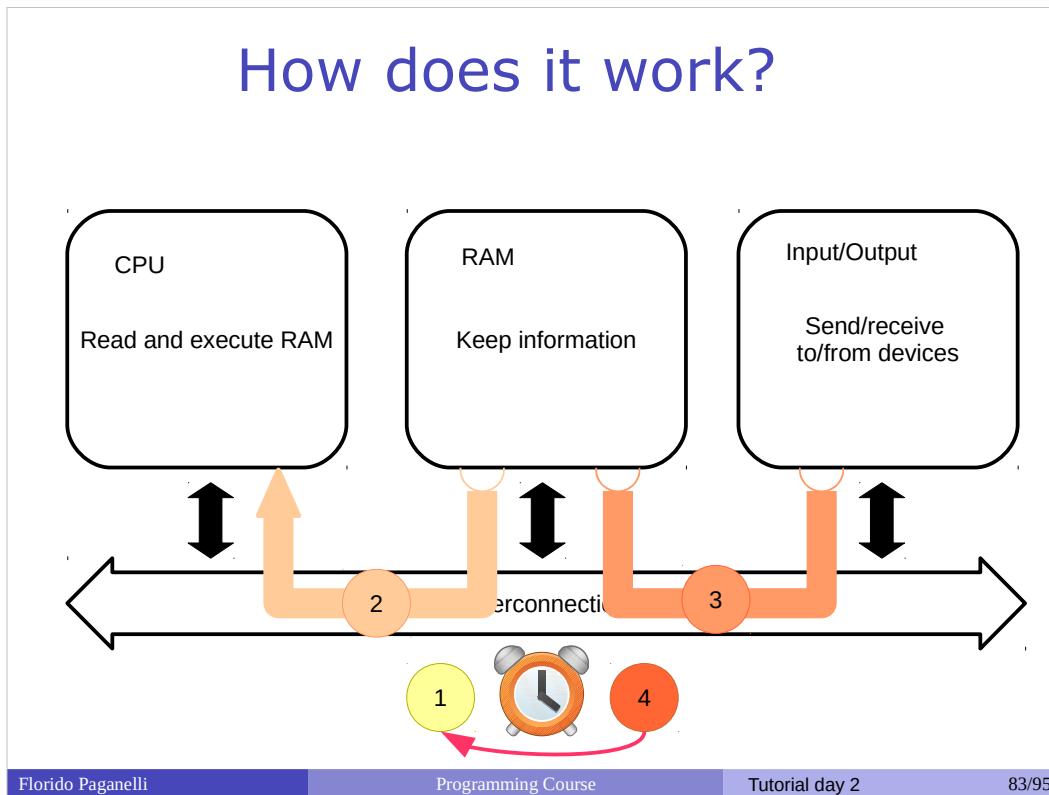
How does it work?



How does it work?



How does it work?



The execution cycle and the clock

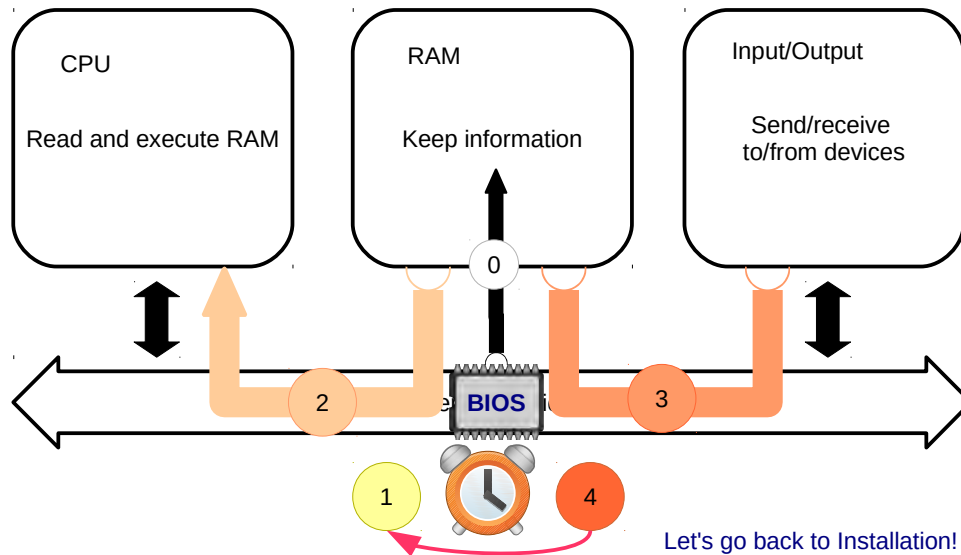
- 1) clock ticks
- 2) CPU reads and executes content of RAM(instructions)
- 3) Execution might dispatch information over the bus
- 4) Wait for next clock cycle

The execution is **always serial**, but gives us a feeling of parallel tasks because of speed.

It might require more than one clock cycle to execute an instruction.

But... How does it start?

But... how does it start?!?



The BIOS

0. The BIOS loads a small program into the RAM. When the clock starts, the CPU will start executing as explained.

References

- Lubuntu installation official documentation
<https://help.ubuntu.com/community/Lubuntu/InstallingLubuntu>
- Lubuntu official documentation / How Tos
<https://help.ubuntu.com/community/Lubuntu/Documentation>
- Virtualbox
<https://www.virtualbox.org/>
- Lubuntu Desktop
- The Sociotechnical Boundaries of Hardware and Software: A Humpty Dumpty History, Brent K. Jesiek,
<http://bst.sagepub.com/content/26/6/497>

Pictures references (not complete)

- <https://openclipart.org/>
- http://en.wikipedia.org/wiki/Eadweard_Muybridge
- http://commons.wikimedia.org/wiki/File:ASRock_P4i65G_motherboard_view.jpg
- <http://perform-360.com/how-to-fix-chronic-lower-back-sorenes/>
-

Homework

Read about the different installation strategies.

Install your own Linux.

Linux installation: three strategies

1 Single operating system: Linux deletes everything on the disk and it becomes the one and only operating system for that computer

- Difficulty: easy
- Costraint: user must own the computer (administrator)
- Pros:
 - Performance: Linux can take all the resources available on the machine
- Cons:
 - The machine can only run Linux programs.

Linux installation: three strategies

2 Virtual Machine: Run Linux in a virtual machine where it is the one and only system. The virtual machine is run on an hypervisor that is running in the existing OS.

- Difficulty: Medium
- Constraint: User must be able to install software, i.e. VirtualBox hypervisor
- Pros:
 - The machine can run programs for the existing OS **and** linux, **at the same time**
- Cons:
 - No need to touch the existing OS boot: all is taken care by the Virtualization Hypervisor
 - Performance loss due to virtualized hardware.

Linux installation: three strategies

3 Dual boot: Linux shares the disk with other operating systems.

Requires an alternative boot loader (e.g. GRUB2), a small program that is loaded **BEFORE** any other installed system. This program must be installed at the beginning of the disk.

- Difficulty: Hard
- Constraint: user must own the computer (administrator)
- Pros:
 - Performance: Linux can take all the resources available on the machine
 - The machine can run programs for windows and linux, at the cost of a reboot
- Cons:
 - Rewriting the initial part of the disk usually prevents windows to boot properly. One needs to be very aware of what he is doing.
 - Only one operating system can control the machine at time: requires reboot to switch from one operating system to the other.

Linux installation: Recommendations

During this course we will use **method 2 (Virtual Machine)**, as it is supposed to work on every platform.

Methods 1 and 3 are only recommended to those who want to learn about the boot process or plan to use Linux for intense computation, or simply to switch to Linux as their main operating system (It's fun! :D).

Method 2 prerequisite: Download and install Virtualbox

<https://www.virtualbox.org/wiki/Downloads>

- 1.1. Download
VirtualBox 4.3.18 for Windows hosts
- 1.2. double click on the Virtualbox file just downloaded and follow the installation procedure on screen.
- 1.3. install linux as seen during the tutorial.

Method 1 prerequisites

- 1.1. Make sure you are using an hard disk that you can **completely wipe. No data but Lubuntu will be left on the disk after installation!!**
- 1.2. Download the Lubuntu ISO as seen in the tutorial.
- 1.3. Burn the ISO on a CD/DVD or on a USB stick. There are several ways to do this, search Ubuntu mainpage or the internet to find how. Start here:
<http://www.ubuntu.com/download/desktop/install-ubuntu-desktop>
- 1.4. Boot the machine from the ISO. Usually it is enough to insert the burned CD or the USB pen and start the machine.
- 1.4. Install Linux as we've seen in the course. Remember, **everything on the hard disk will be removed!!**

Method 3 prerequisites

DISCLAIMER: USE THIS METHOD ONLY IF YOU UNDERSTAND WHAT YOU'RE DOING. YOU RISK DESTROYING YOUR EXISTING DATA.

- 1.1. Download the Lubuntu ISO as seen in the tutorial.
- 1.2. Burn the ISO on a CD/DVD or on a USB stick. There are several ways to do this, search Ubuntu mainpage or the internet to find how. Start here:
<http://www.ubuntu.com/download/desktop/install-ubuntu-desktop>
- 1.3. Make space on the harddisk to fit linux. You usually need at least 40GB. To do that on windows, read about it here:
<http://technet.microsoft.com/en-us/magazine/gg309169.aspx>
- 1.4. Decide how to install linux: using GRUB, using Windows bootloader... and follow the procedures. More information here:
<https://help.ubuntu.com/community/WindowsDualBoot>
For Windows, I suggest to use windows boot loader and a tool called EasyBCD. A guide here:
<http://askubuntu.com/questions/325402/booting-win7-12-04-what-do-i-need-for-easybcd>
- 1.5. Install linux as described in the course, but targetting the correct partition you made space for on the disk. **Do not choose "erase entire disk" when prompted.**