M01 Sistemes Informàtics

CFGS Desenvolupament d’Aplicacions Multiplataforma
UF1. Instal·lació, configuració i explotació del sistema informàtic.

NF1.2

Anàlisi i instal·lació de sistemes informàtics

Estructura i components físics del sistema informàtic

Introducció

En un sentit genèric, un ORDINADOR és aquell aparell dissenyat per processar informació.

- Un procés es correspon a les fases o transformacions que soporta la informació per tal de resoldre un problema determinat.
- L'arquitectura d'un computador defineix el seu comportament funcional.

Un ordinador es pot definir com una màquina electrònica capaç de fer les següents tasques:

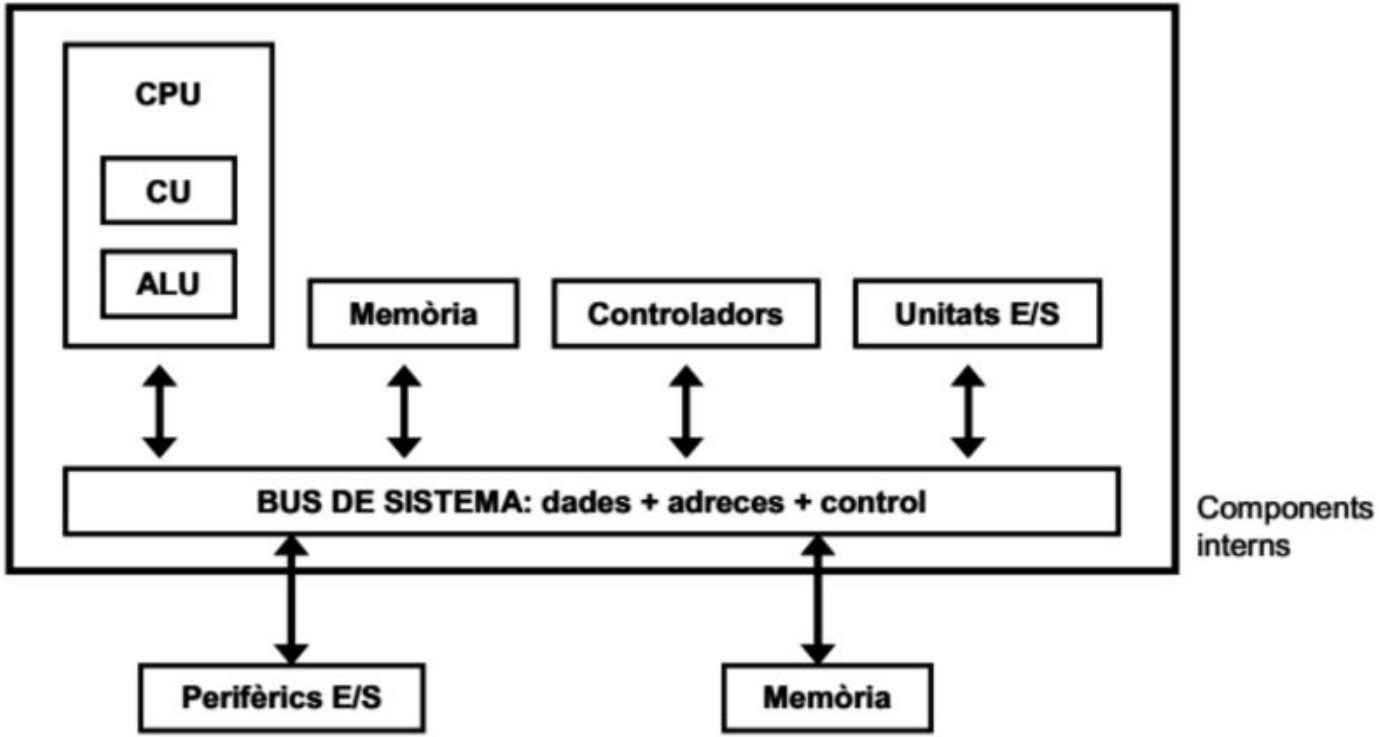
- Acceptar informació
- Emmagatzemar informació
- Processar la informació
- Produir i proporcionar resultats

Introducció

Un ordinador disposa de quatre components principals per a fer les tasques descrites:

- Unitats E/S. Mouen dades entre el computador i l’entorn extern
- Processador (CPU. Central Processing Unit). Controla l’operació del computador i fa el processat de dades
- Memòria per emmagatzemar la informació i les instruccions
- Sistemes de comunicació (BUS). Mecanisme que comunica els components del sistema

Introducció



CPU

La CPU és l’encarregada de controlar tots els dispositius interns i externs, i executa totes les instruccions i operacions de l’ordinador.

Circuit integrat que executa les instruccions de programa que estan emmagatzemats a l’ordinador. És el “cervell” d’un ordinador

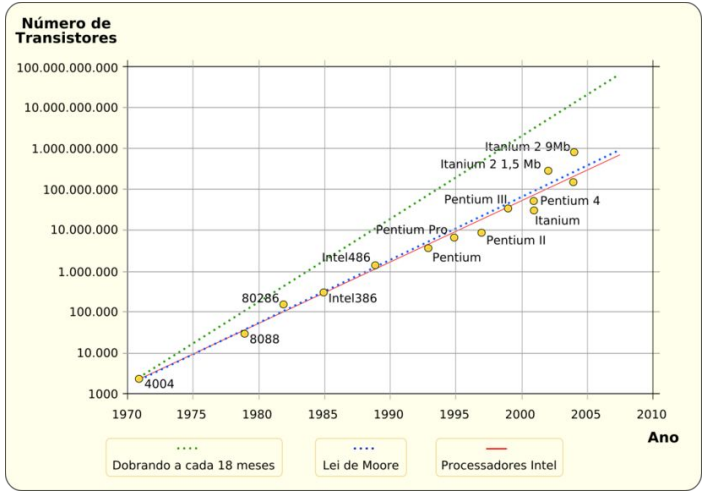
Es divideix en 3 components principals

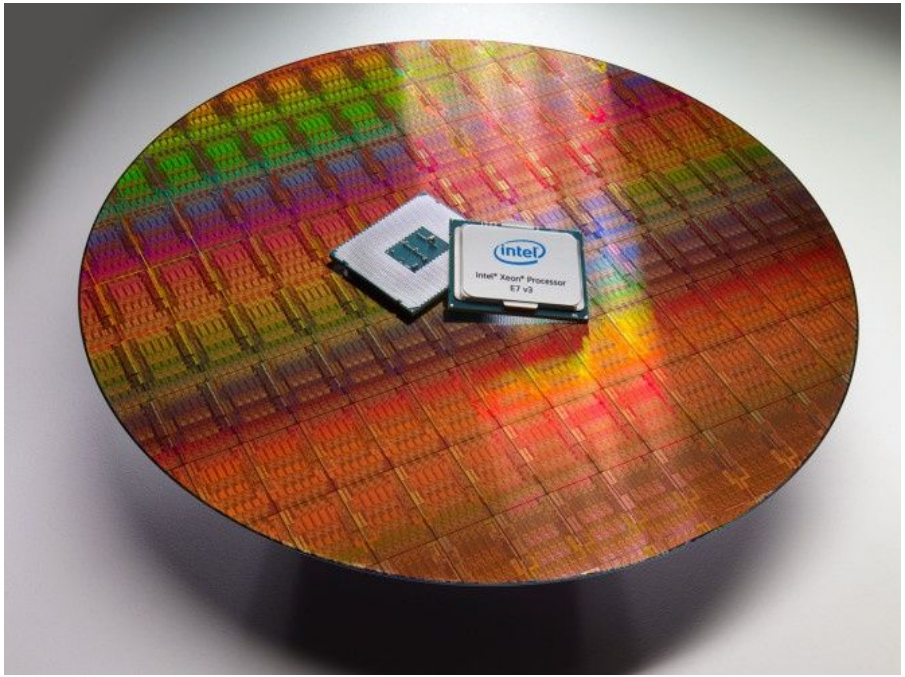
- Registres. Registre d’instrucció (IR), Comptador de programa (PC), Rellotge (1Mhz → 106 impulsos/segon)
- Unitat Aritmètica-Lògica (ALU). Realitza les operacions aritmètiques i lògiques.
- Unitat de control (UC). Interpreta les instruccions del programa i genera els senyals de control

Mesurem el seu rendiment mitjançant el nombre d’instruccions i d’operacions per segon que pot fer (Hz, MHz, GHz) i mitjançant el nombre de bits que pot gestionar en cada instrucció i operació (32-bits, 64-bits)

CPU

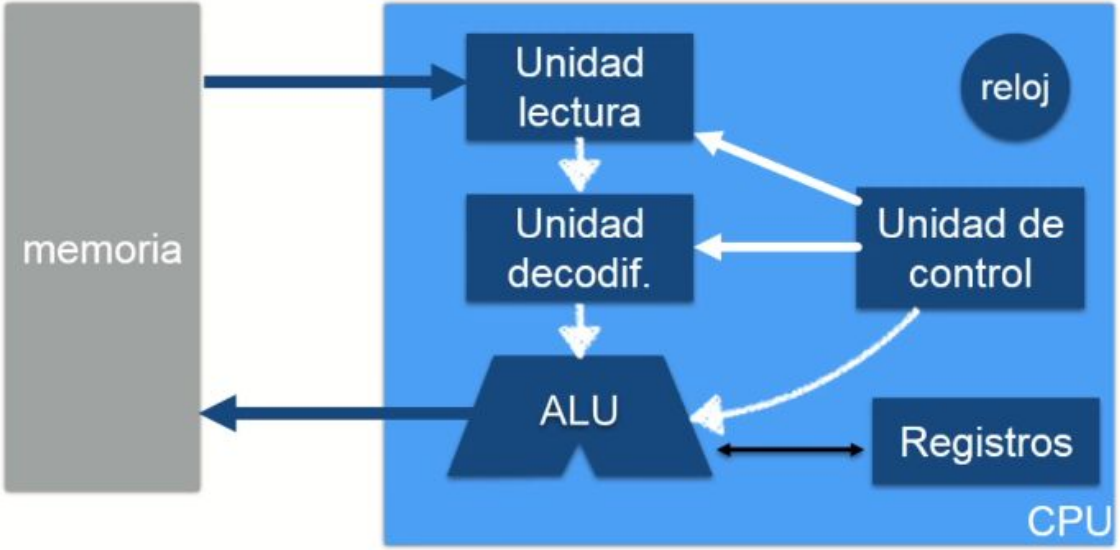
- Conté entre 500 i 1000 processadors.
- [Llei de Moore (cofundador d'Intell)]





CPU

- La unitat de lectura llegeix la següent instrucció de la memòria principal i la decodifica (unitat de decodificació)
- Agafa les dades de la memòria que necessiti per executar la instrucció i ho guarda als registres per poder operar-los dins l'ALU (unitat Aritmètica lògica) i torna les dades a la memòria.
- Mecanisme d'interrupció



CPU

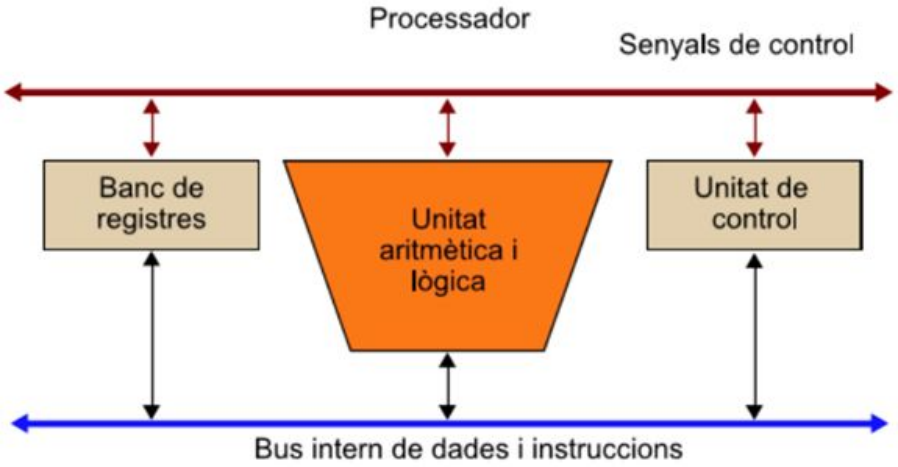
És necessari disposar d'un sistema per interconnectar els components.

- Línies de control → governar el processador
- Línies de dades → transferir dades i instruccions

Vídeos:

[El processador]

[Què hi ha a dins de la CPU?]



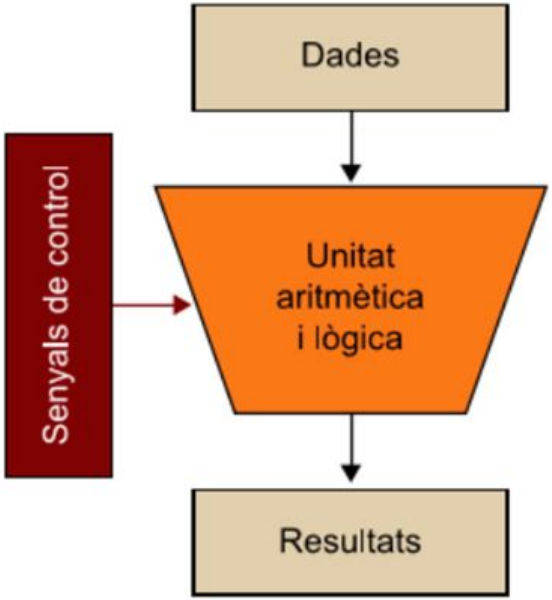
CPU. ALU

Unitat Aritmètica Lògica (Arithmetic Logic Unit)

És un circuit digital que calcula operacions aritmètiques (com suma, producte etc.) i operacions lògiques (com OR, NOT, XOR, etc.), entre dos nombres.

La Unitat Aritmètica Lògica està feta d'un conjunt de circuits combinacionals dissenyats per obtenir resultats de càlculs.

Les operacions poden ser aritmètiques (suma, subtracció, producte...) o lògiques (and not... bit a bit), i desplaçaments del contingut de registres.



Registres informàtics

Els registres són elements de memòria SRAM d'accés ràpid. Són imprescindibles: l'ALU treballa amb registres interns del processador.

Registres d'accés a memòria:

- Memory address register (MAR): on posem l'adreça de memòria a què volem accedir.
- Memory buffer register (MBR): on la memòria diposita la dada llegida o la dada que volem escriure.
- Registre d'estat (ST): Els bits d'aquest registre es modifiquen com a resultat de l'execució d'algunes instruccions. P.e.: bit de zero, bit de transport, bit de signe, …
- Registre acumulador (AC): Guarda el resultat de les operacions.
- Altres registres de propòsit general: dades i/o adreçament.

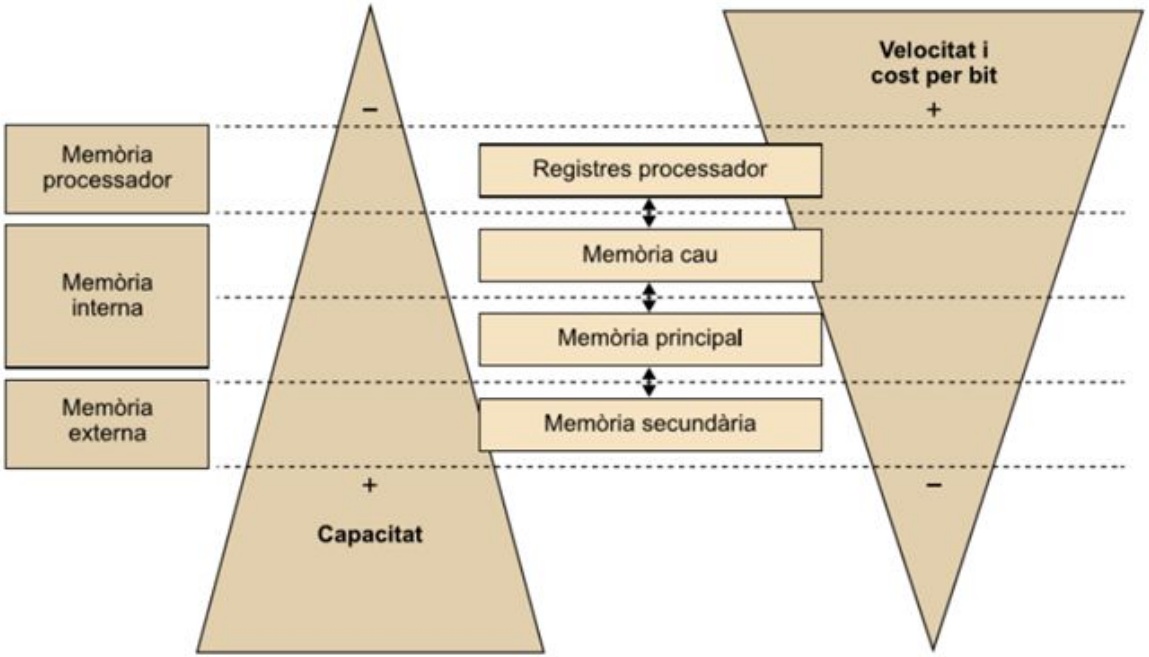
Registres informàtics

- Exemple lectura d’una instrucció
    1. MAR←PC: es posa el contingut del registre PC al registre MAR.
    2. MBR←Memòria: es llegeix la instrucció.
    3. PC←PC+Δ: s'incrementa el PC tantes posicions de memòria com s'han llegit (Δ posicions).
    4. IR←MBR: es carrega la instrucció en el registre IR.
- La informació emmagatzemada en el registre IR es descodifica per a identificar les diferents parts de la instrucció i determinar les operacions necessàries que cal fer en les fases següents.

La memòria

En una jerarquia de memòries s'utilitzen diferents tipus de memòria amb característiques diferents de capacitat, velocitat i cost, que dividirem en nivells diferents: memòria del processador, memòria interna i memòria externa.

Cada nivell de la jerarquia es caracteritza també per la distància a la qual es troba del processador. Els nivells més propers al processador són els primers en ser utilitzats; això és així perquè també són els nivells amb una velocitat més elevada.



La memòria. Primària

La memòria primària és la memòria principal d’un ordinador. S’encarrega d’emmagatzemar les dades i les instruccions necessàries per executar-se a la CPU.

Hi ha dos tipus de memòria primària

- RAM - memòria volàtil
- ROM - Memòria no volàtil i només de lectura, conté la BIOS

El rendiment d’un ordinador té molta dependència de la quantitat de memòria RAM disponible. Es mesura habitualment en GB

La memòria. RAM

Memòria volàtil (que no manté l’estat sense corrent elèctrica), de més capacitat que la memòria cau.

1) SRAM

- Cada cel·la 1 bit
- Major cost
- Més ràpid
- Ús: registres CPU i memòria CAU

2) DRAM

- Cada cel·la formada per un transistor i un condensador
- Circuit de refresc → temps d’accés més gran
- gran escala d’integració
- Ús: construcció de la memòria principal

3) SDRAM (Synchronous DRAM) → DDRA

La memòria. Caché i secundària

La memòria caché s’utilitza per accelerar l’accés a les dades per part de la CPU, s’ubica entre la RAM i la CPU i no utilitza cap tipus de bus entre ella i la CPU. Té poca capacitat i és molt ràpida.

La memòria secundària s’utilitza per emmagatzemar dades que necessitem que tinguin una persistència un cop apaguem l’ordinador. La velocitat d’accés a aquestes memòries és molt més baixa que per accedir a la caché o a la RAM, però permet emmagatzemar quantitats molt més grans d’informació (TB)

Un cop la CPU satura tota la capacitat de la RAM comença a utilitzar també la memòria secundàira per emmagatzemar les dades i les instruccions necessàries per continuar treballant. Això implica una baixada molt gran en el rendiment.

La **memòria externa** està formada per dispositius d'emmagatzematge secundari (discos magnètics, CD, DVD, Blu-ray, etc.). Aquests dispositius es poden trobar físicament dins del computador o fora del computador. És de tipus no volàtil.

El mètode d'accés varia segons el dispositiu: generalment els dispositius basats en disc utilitzen un mètode d'accés directe.

Memòria. La memòria CAU

- Les memòries cau són memòries de capacitat reduïda, però més ràpides que la memòria principal.
- Es poden trobar dins del xip del processador o a prop, estan dissenyades per a reduir el temps d'accés a la memòria.
- A la memòria cau s'emmagatzemen les dades que es preveu que seran utilitzades més habitualment, de manera que sigui possible reduir el nombre d'accessos que ha de fer el processador a la memòria principal (ja que el temps d'accés a la memòria principal sempre és superior al temps d'accés a la memòria cau).
- S'implementa utilitzant tecnologia SRAM

Memòria. Memòria externa

La memòria externa està formada per dispositius d'emmagatzematge secundari (discos magnètics, CD, DVD, Blu-ray, etc.). Aquests dispositius es poden trobar físicament dins del computador o fora del computador.

La memòria externa és de tipus no volàtil.

El mètode d'accés varia segons el dispositiu: generalment els dispositius basats en disc utilitzen un mètode d'accés directe, mentre que altres dispositius com les cintes magnètiques poden utilitzar accés seqüencial.

Dispositius d’E/S

Els dispositius d’entrada són els responsables d’obtenir la informació del món exterior. Els dispositius de sortida, de comunicar els resultats obtinguts.

També els coneixem amb el nom de perifèrics i es connecten a l’ordinador mitjançant ports o adaptadors.

Tota operació d'E/S que es du a terme entre l’ordinador i un perifèric és sol·licitada i governada des del processador. La gestió global d’un sistema d’ E/S la fa el sistema operatiu.

Quan el SO vol fer una operació d'E/S amb un perifèric crida a una d'aquestes rutines.

- Necessiten d’un Hardware addicional anomenat controlador per guardar l’estat i transferir les dades del dispositiu
- Necessiten d’un Software addicional anomenat drivers per possibilitar la comunicació del dispositiu amb la CPU

Interconnexió entre elements

La CPU i la memòria principal es troben a la placa mare de l'ordinador. Aquesta placa mare és un circuit imprès que a part d'aquests elements també pot incorporar més elements HW (controladors, targetes d'expansió…) Aquesta interconnexió entre els diferents elements i la CPU es realitza mitjançant l'ús de busos. Entenem com a bus els cables conductors o pistes de circuit que proporcionen un camí als impulsos elèctrics

Conjunt de fils físics utilitzats per la transmissió de dades entre els components d'un sistema informàtic.

- **Bus de dades:** Transmet informació entre la CPU i els perifèrics.
- **Bus d'adreces:** Transmet l'adreça de memòria.
- **Bus de control:** Gestiona l'ús i l'accés a bus de dades i d'adreces.

La BIOS

BIOS

La BIOS (Basic Input Output System) és un firmware (és a dir, una porció de codi emmagatzemada en una memòria de només lectura) encarregat d’iniciar els components de maquinari, habilitar funcions d’energia i temperatura i carregar el sistema operatiu d’un ordinador quan l’engeguem.

És un programa incorporat en un xip de la placa base que s'encarrega d'inicialitzar i identificar tots els dispositius connectats a l'ordinador com són el teclat, targeta gràfica, disc durs... i la resta del maquinari.

Primer programa que s’executa en encendre l’ordinador. Inicialment memòria ROM

Firmware: software que té interacció directa amb el hardware

Per què es diu que és només lectura si es pot actualitzar?

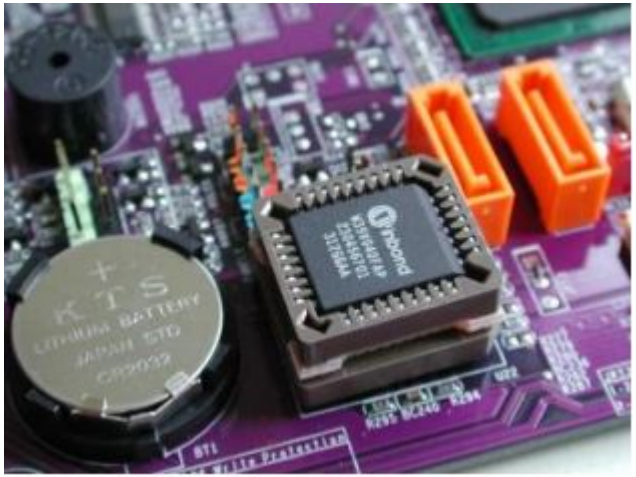
En primer lloc, les primeres BIOS no es podien modificar. Si volies afegir qualsevol instrucció havies de canviar el Xip. Després, van aparèixer les primeres memòries **PROM** (Programmable ROM, tot i que només es podien programar un cop), després les **EPROM** (Erasable PROM, programables moltes vegades mitjançant raig ultraviolat), després les **EEPROM** (Electrical EPROM) programables elèctricament.

Actualment la majoria de BIOS són memòries **Flash** que poden actualitzar-se via software. Tot i que poden actualitzar-se, el 99% de les vegades que fem servir una BIOS serà per arrencar un ordinador i, per tant, per LLEGIR el seu contingut.

Si la BIOS és una memòria no volàtil per a què necessita la pila?

La Bios no requereix la pila per mantenir les seves instruccions. Si apaguem l'ordinador el codi de la BIOS es manté, sense cap problema. La pila serveix per alimentar la memòria no volàtil CMOS (Complementary Metal-Oxide-Semiconductor) que emmagatzema les variables que utilitzarà la BIOS durant l'arrencada (Data, hora, freqüència de bus, seqüència d'inici de dispositius, contrasenya de BIOS, etc.)

Si traiem la pila, perdrem el contingut de la memòria CMOS, no el de la BIOS, Per tant, encara que és molt freqüent, no és correcte dir que la pila manté la informació que guarda la BIOS. La BIOS, no obstant això, guarda uns valors per assignar per defecte als paràmetres de la CMOS, en cas de pèrdua per falta d'energia.

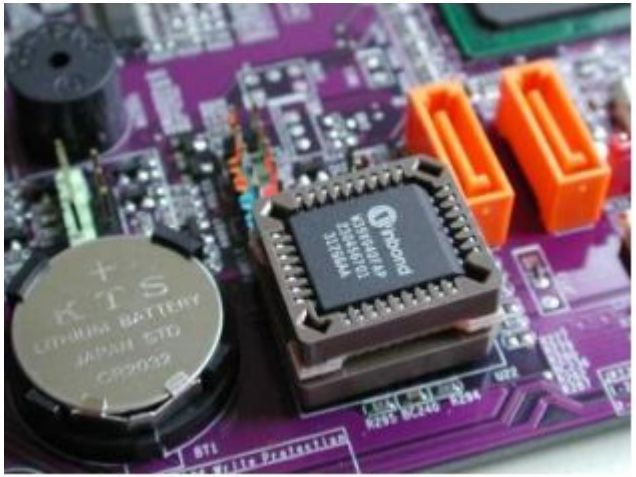


Es poden canviar els valors de la memòria CMOS? i esborrar-la?

Es pot fer, mitjançant el menú de la BIOS, o més ben dit, el CMOS Setup.

Per esborrar el seu contingut de la CMOS (el que porta com a resultat omplir amb els valors per defecte els seus paràmetres) tenim a la nostra disposició tres formes de fer-ho:

1. Traient la pila (com hem comentar ja abans)
2. Seleccionant al menú l' opció LOAD SETUP DEFAULTS
3. Posant el jumper “CLEAR CMOS” a la posició d'esborrat (per fer això hem de consultar el nostre manual de la placa base).



BIOS. Com intervé la BIOS durant l'arrancada del PC?

1. La placa Base s'alimenta elèctricament. Quan la tensió elèctrica s'estabilitza el processador busca la primera instrucció a una determinada direcció de la memòria BIOS.
2. El processador executa una sèrie d'instruccions denominades POST (Power on Self Test) emmagatzemades a la BIOS que permeten comprovar els components indispensables per a que el sistema informàtic pugui funcionar (aquestes instruccions utilitzen diferents valors que són carregats a partir de la memòria CMOS):

    a) Es verifica una primera part de la memòria RAM (RAM Baixa).

    b) S'inicia el dispositiu de vídeo. Aquest dispositiu té la seva pròpia BIOS.

    El processador processarà totes les instruccions d'inicialització del dispositiu de vídeo contingut a la BIOS vídeo. A partir d'aquest moment ja es poden visualitzar missatges al monitor.

BIOS. Com intervé la BIOS durant l'arrancada del PC?

c) S'inicia el controlador de teclat.

d) Es verifica la resta de la memòria RAM.

e) Es verifiquen els dispositius presents (CDROM, Discs Durs, etc...)

f) Si la Bios i els dispositius suporten Plug and Play (Pnp → “enchufar, conectar y usar”) es configuren automàticament (interrupcions, Memòria, Entrades/Sortides, velocitat, etc.).

g) Es mostra un informe mitjançant el monitor.

h) S'espera durant un breu interval de temps per veure si l'usuari vol entrar o no al menú de CMOS setup o bé varia l'ordre de la “boot sequence”.

i) S'inicia el dispositiu prioritari (USB->CDROM->HD1->HD2->USB LAN) i busca el MBR (Master Boot Record, el primer sector del disc, que conté els paràmetres per arrencar el sistema) o bé el GPT (si es tracta d’una UEFI en lloc d’una BIOS). Un cop troba el MBR (o el GPT) Ha acabat la feina de la BIOS

BIOS. Com intervé la BIOS durant l'arrancada del PC?

3. El processador executa el codi contingut al MBR (o GPT). Aquest codi, bàsicament, mira com està dividit el disc dur (particions) i busca la partició activa. Dins aquesta partició, busca el primer sector que és on es troba el codi que iniciarà el Sistema Operatiu instal·lat a la partició.

Si troba cap problema retorna un o més sons i mostra un missatge per pantalla (aquest missatge només es podrà veure si s'ha pogut carregar la targeta gràfica)

AMI (American Megatrends International): AMI fa servir sons de la mateixa longitud.

Award Bios Beep Codes. Award Bios: Fa servir sons de diferent duració.

Phoenix Bios Beep Codes.

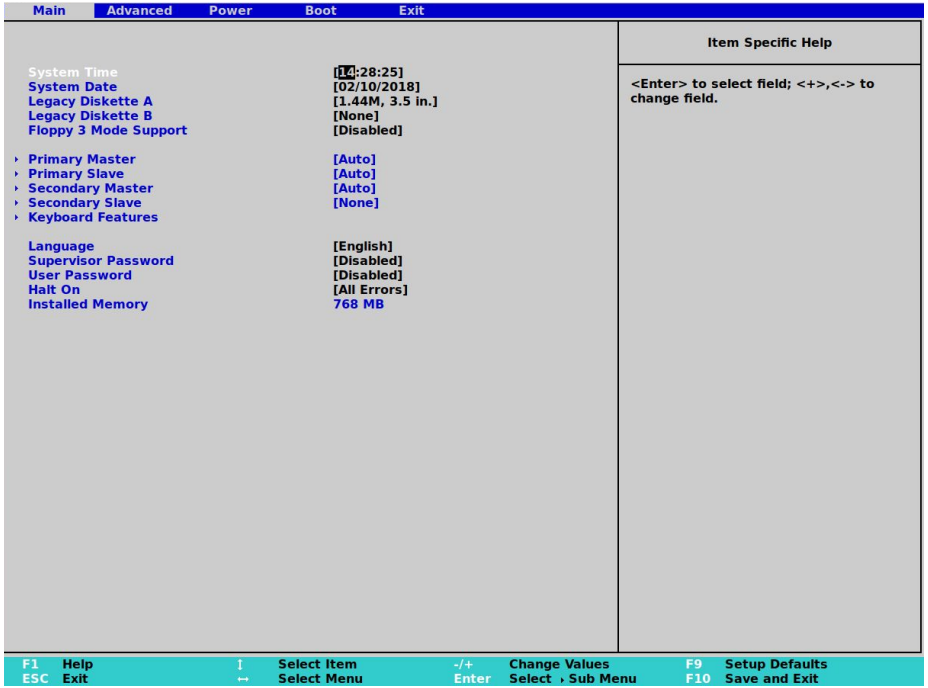
UEFI

UEFI Unified Extensible Firmware Interface (UEFI), és una especificació que substitueix l'anterior BIOS. És a dir, realitza la mateixa funció que aquesta però aportant una sèrie de millores molt remarcables.

- Compatibilitat per a sistemes operatius només compatibles amb BIOS.
- Substitueix les taules de particions MBR per les taules de particions GPT. Això suposa que es poden crear fins a 128 particions per disc (només 4 a MBR) cadascuna amb una capacitat total de fins a 8ZB (2,2TB a MBR).
- Permet arrancar amb unitats d'emmagatzematge més grans.
- Millores a la interfície de modificació de la BIOS/UEFI (en moltes ocasions disposen d'interfície gràfica).
- Disseny d'arquitectura compatible amb 32 i 64 bits (MBR disposa d'arquitectura 16 bits) Això suposa que UEFI accepti incorporació de nous mòduls que milloren les característiques de configuració del sistema.
- Incorpora característiques avançades de BIOS, com ACPI (Interfaz Avanzada de Configuración y Energía) y el SMBIOS (Sistema de Gestión de BIOS).

Com hi accedim?

Per accedir a la BIOS/UEFI hem de reiniciar l'ordinador i pulsar una de les tecles següents (pot variar en funció de la màquina): **F2, F10, F12, ESC, Supr**, o qualsevol altra indicada al manual del sistema informàtic en qüestió.



Què hi trobem? Main

Secció ‘Main’ de la UEFI BIOS

En aquesta secció podem trobar la següent informació:

- Fabricant, versió i data de construcció de la BIOS/UEFI.
- Quantitat total de memòria RAM del dispositiu.
- Idioma per defecte de la BIOS/UEFI.
- Data i hora del sistema.
- Seqüència de tecles per tal d’interactuar amb la interfície.

Què hi trobem? Advanced

Secció ‘Advanced’ de la UEFI BIOS

En aquesta secció podem trobar la següent informació:

- Launch PXE OpRom: Habilita o inhabilita els controladors necessaris per tal d’arrancar l’ordinador o instal·lar un sistema operatiu a la màquina mitjançant una xarxa.
- Launch Storage OpRom: Habilita o inhabilita controladors necessaris per tal d’accedir a dispositius d’emmagatzematge.
- PCI Subsystem Settings: Permet configurar paràmetres dels busos PCIe de la placa.
- ACPI Settings: Permet configurar paràmetres de la interfície Avançada de configuració d’energia.
- CPU Configurations: Permet configurar paràmetres relacionats amb la CPU del PC (per exemple, habilitar característiques de virtualització de la CPU).
- SATA Configuration: Permet configurar paràmetres dels busos SATA de la placa.
- USB Configuration: Permet configurar paràmetres dels ports USB del PC.
- Serial Port Console Redirection: Permet configurar el port sèrie del PC.

Què hi trobem? Boot

Secció ‘Boot’ de la UEFI BIOS

En aquesta secció podem trobar la següent informació:

- Quiet Boot: Si està habilitat, oculta els resultats de tests del POST a la pantalla d’inici de l’ordinador.
- Fast Boot: Si està habilitat, evita realitzar determinats tests avançats de l’estat de la memòria durant el POST de la màquina.
- Setup Prompt Timeout: Pausa l’inici un número determinat de segons per a que es puguin veure els missatges del POST.
- BootUp NumLock State: Especifica l’estat de la tecla ‘numlock’ durant l’inici.
- Boot Option Priorities: Conté una llista jeràrquica d’arxius o perifèrics per tal de tractar d’iniciar el sistema.

Què hi trobem? Security

Secció ‘Security’ de la UEFI BIOS:

En aquesta secció podem configurar dos nivells d’accés a la BIOS, l’administrador (amb accés total) i l’usuari (amb algunes característiques restringides) podem trobar la següent informació:

- Administrator Password: Configura la clau de pas d’administrador.
- User Password: Configura la clau de pas d’usuari.

Què hi trobem? Save and exit

Secció ‘Save & Exit’ de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

- Save changes and Exit: Surt de la configuració de la BIOS/UEFI guardant els canvis i tancant la màquina.
- Discard changes and Exit: Surt de la configuració de la BIOS/UEFI sense guardar els canvis i tancant la màquina.
- Save Changes and Reset: Surt de la configuració de la BIOS/UEFI guardant els canvis i reiniciant la màquina.
- Discard Changes and Reset: Surt de la configuració de la BIOS/UEFI sense guardar els canvis i reiniciant la màquina.
- Restore Defaults: Restaura els valors de la BIOS/UEFI als valors de fàbrica.
- Boot Override: Mostra i configura els diferents arxius d’inici que poden utilitzar-se per a inicialitzar el sistema mitjançant la llista present a ‘Boot Option Priorities’.

Més informació

Document amb més informació: aquí

Tutorial youtube. En aquesta serie de set vídeos explica molt bé els diferents apartats de la BIOS. No està actualitzat a l'última tecnologia però està ben explicat.

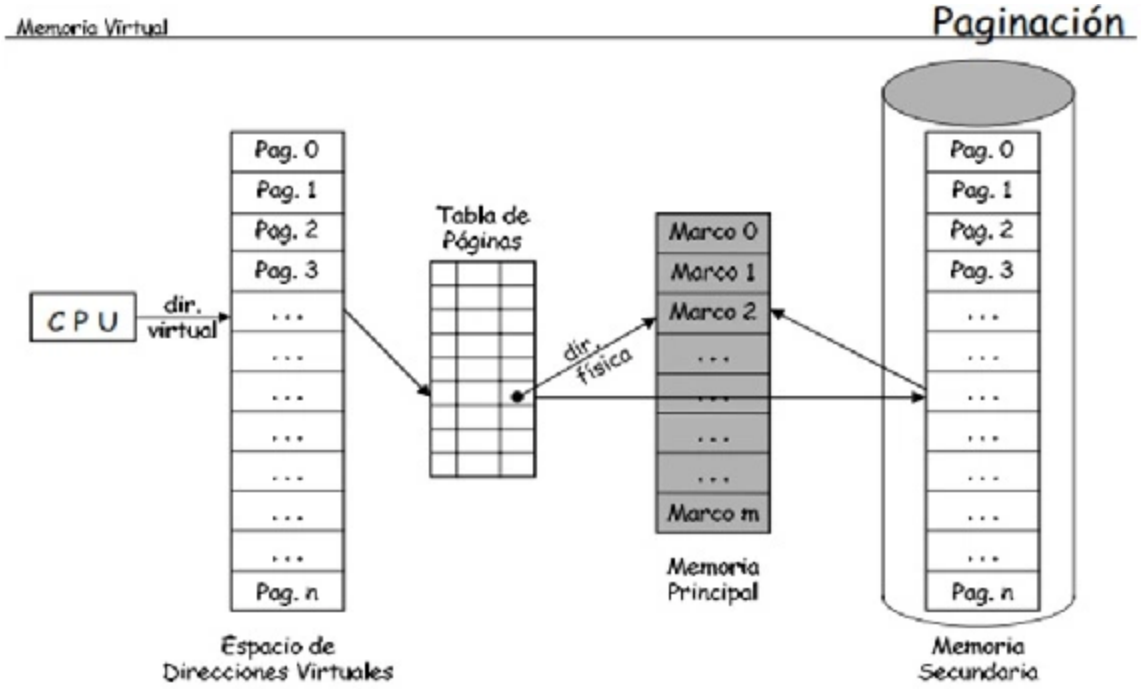
Hiperpaginació (thrashing)

Paginació

La memòria es divideix en pàgines de mida fixa, típicament poden ser de 4KB. Això permet que la memòria d'un programa hi sigui paginada, fraccionada en pàgines, permetent la intercalació amb altres programes.

Compte! Reservem la paraula "*fragmentada*" per indicar els casos en què queden fragments de memòria desaprofitats, sovint quan tenim necessitat de més memòria, però els "*fragments*" no serveixen, perquè ens fa falta un espai de memòria contigu.

La memòria RAM real del sistema es divideix en marcs (*frames*) de la mida de les pàgines; cada marc de memòria allotjarà una pàgina. La taula de pàgines proporciona l'adreça real de la pàgina, i cada adreça que referència un programa s'ha de traduir fent servir aquesta taula de pàgines.



Segmentació

La segmentació permet la diferenciació entre diferents programes, però també entre dades i codi d'un mateix programa, i fins i tot entre diferents dades (la pila -*stack*-, el muntó -*heap*-...). La característica principal de la segmentació és que **ofereix**

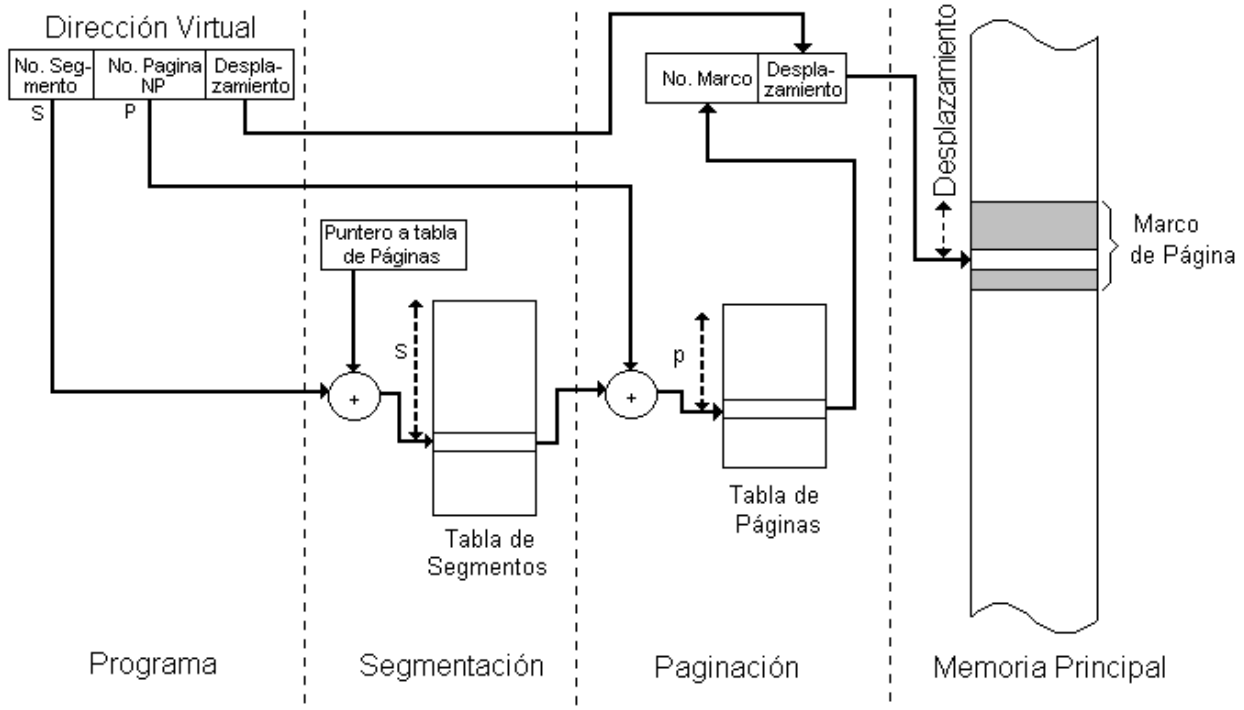
protecció, impedint que uns programes puguin tenir afectació sobre altres, tant intencionadament com per mal funcionament.

També ofereix altres avantatges, com ara la **modularitat** del codi font dels programes o la possibilitat de compartir segments de dades entre diferents processos, preservant la resta de segments d'afectacions indesitjades, i altres avantatges d'una gestió modular de la memòria…

La característica de segmentació també requereix el càlcul de l'adreça real fent servir una taula, de manera similar a com es fa en la paginació.

Segmentació paginada

La segmentació ofereix uns avantatges diferents de la paginació, però poden combinar-se totes dues, obtenint doncs els beneficis d'ambdues. Naturalment la segmentació paginada requereix una taula de segments i una taula de pàgines.



Memòria virtual: memòria d'intercanvi o de swap

La paginació permet direccionar algunes pàgines fora de la memòria RAM del sistema, pàgines que no es troben en cap marc de memòria, sinó en memòria secundària (disc). Quan en la taula de pàgines no es troba l'adreça es produeix una ***fallida de pàgina***. La

taula de pàgines tindrà l'adreça de memòria secundària on localitzar la pàgina, però serà en una memòria secundària, i per tant, més lenta; el procés quedarà bloquejat esperant, i altres processos s'executaran mentre la pàgina fallida és carregada en un marc de memòria.

Configuració de la memòria virtual

La memòria virtual els sistemes permeten que sigui configurada per l'usuari, indicant la seva mida.

Cal considerar que com més memòria virtual configurem, més memòria ens cal per mantenir la taula de pàgines, que sempre ha de ser a memòria física.

Windows

La memòria virtual en Windows és un gran desconegut. Com que la configuració per defecte en el moment de la instal·lació de Windows és "*Administrar automàticament la mida de l'arxiu de paginació per a totes les unitats*", sovint els usuaris deixen aquesta característica sense configurar, i fins i tot desconeixen la opció de canviar la configuració.

El problema apareix quan fem servir molta memòria, i el sistema queda bloquejat: apareix el fenomen de la **hiperpaginació** (thrashing).

La configuració que ofereix Windows per defecte és una configuració que evita que l'usuari quedi sense memòria, rebent l'incòmode missatge de "memòria insuficient" en algun dels programes que s'executen. Desgraciadament l'usuari, sense el missatge d'error, sovint queda ignorant de la problemàtica, amb un sistema degradat, que arriba a l'extrem de no respondre les ordres de l'usuari.

La memòria virtual pot configurar-se en diversos discos alhora. Podem considerar que es produeixen moltes escriptures, però avui dia això ja no és un impediment per a considerar en primer lloc els ràpids discos SSD (els discos SSD tenen un límit en el nombre d'escriptures), millorant notòriament el rendiment del sistema.

És cert que els discos SSD més antics era convenient evitar fer-los servir per a la memòria virtual, perquè si anem justos de memòria es produeixen moltes escriptures. Així, fa pocs anys, tot i tenir un ràpid disc SSD es preferían els discos mecànics per a una funcionalitat tan crítica com la memòria virtual, on la velocitat és molt important.

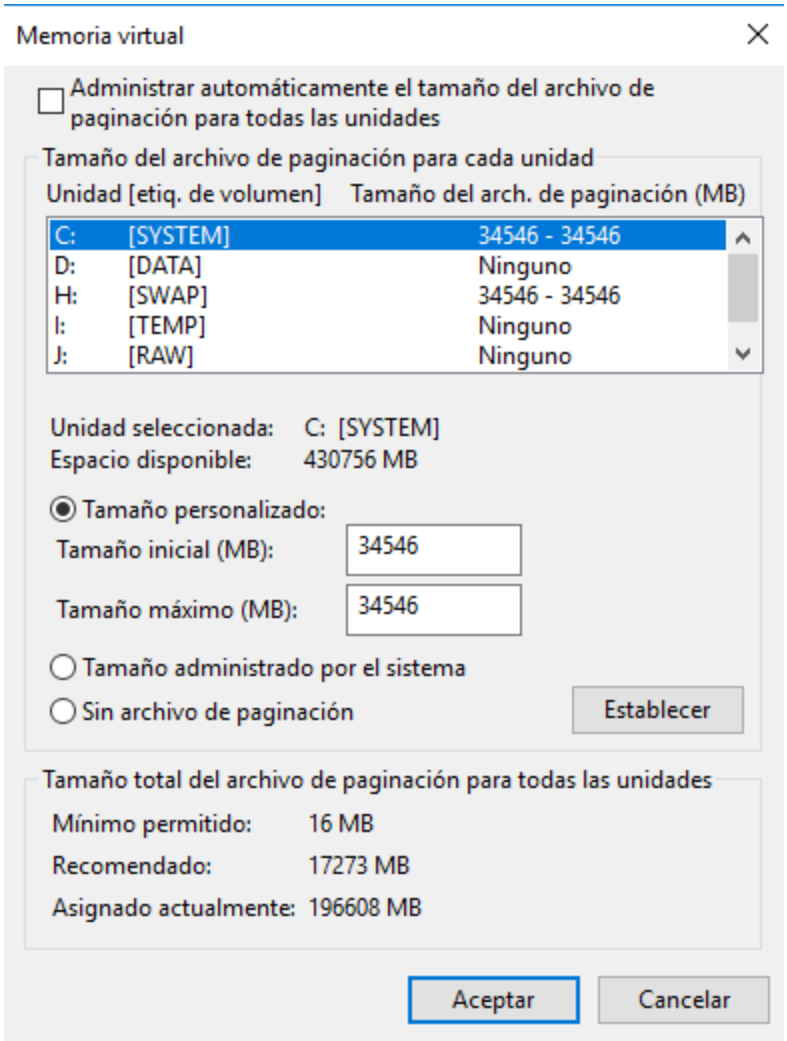
La divisió de la memòria virtual en diferents discos també pot proporcionar una millora del rendiment, sempre que siguin discos de velocitat similar.

Configuració:

És important desactivar la gestió automàtica, i fixar una mida. Microsoft recomana:

1. Fer servir la mida recomanada que es mostra en el mateix diàleg com a **mida mínima**.
2. Prendre el doble d'aquesta quantitat com a **mida màxima**, però **sempre sense excedir el doble de la memòria física de l'equip**. Si el doble de la memòria virtual recomanada excedeix el doble de la memòria física, és a dir, quan la memòria virtual recomanada excedeix la memòria física, clarament tenim un indicador de mancança de memòria.

Naturalment podem tenir les nostres pròpies consideracions, perquè sempre existeix el factor del programari que s'està fent servir, i és difícil que una estimació automàtica pugui considerar alguns factors. Cal ser pessimista amb els requeriments de memòria, però mirar de no sobredimensionar-la gaire, perquè, com ja s'ha comentat, les estructures necessàries per tenir memòria virtual han de ser a memòria, i quan s'incrementa la memòria virtual n'hi ha una pèrdua de memòria física.



Assignació de memòria virtual en Windows (arxiu d'intercanvi). Exemple en un servidor amb 128MB de RAM; la memòria virtual recomanada és només de 17273MB. En aquest cas s'estableix una mida més gran de la recomanada per fer servir aplicacions amb ús intensiu de memòria; la configuració anterior havia estat de 196608MB, encara per sota del valor crític de 256GB, el doble de la memòria física. En aquest cas la memòria virtual es divideix en dos diferents discos: el disc del sistema (C:) i una partició d'un altre disc que s'ha reservat per aquesta funcionalitat (H:)

Linux

Linux pot fer servir un fitxer del sistema de fitxers o una partició separada per intercanviar espai. Una partició d'intercanvi és més ràpida, però és més fàcil canviar la mida d'un fitxer d'intercanvi.

Configuració:

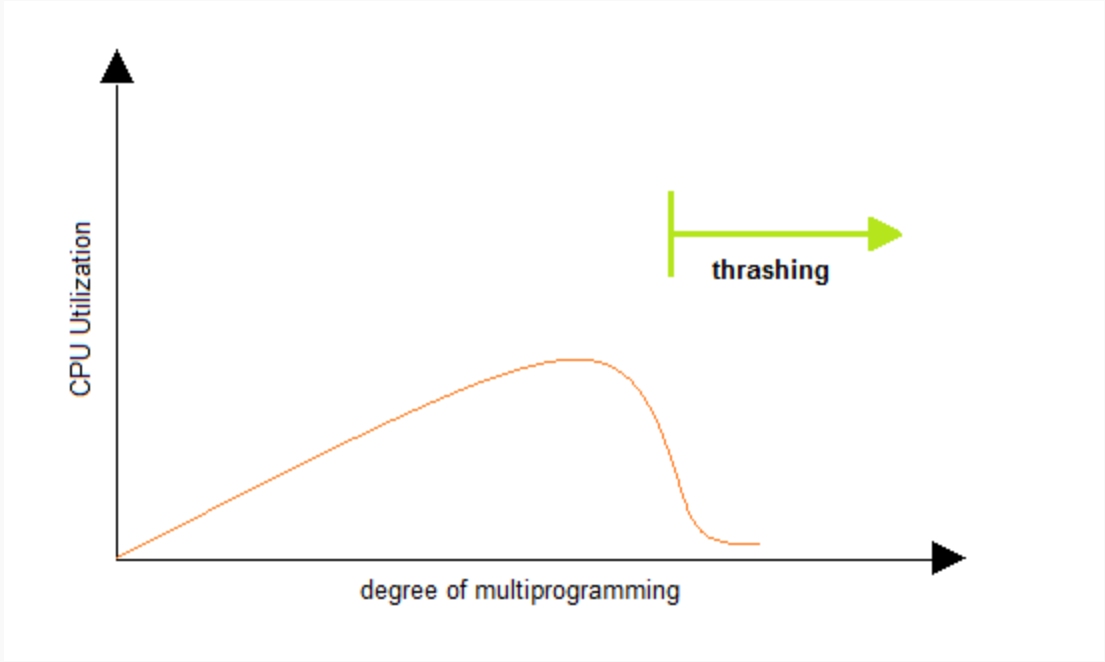
Configuració de la memòria virtual (de swap) en Linux

https://myhsts.org/tutorial-how-linux-virtual-memory-and-disk-buffer-cache-work.php

Swappiness.

https://ubunlog.com/swappiness-como-ajustar-el-uso-de-la-memoria-virtual/

Hiperpaginació (Thrashing)



La hiperpaginació produeix una caiguda dràstica del rendiment, perquè el sistema està paginant. El problema es dona perquè els programes no tenen el conjunt de treball a memòria (les pàgines necessàries) i es produeix una paginació, que per atendre-la el sistema ha de retirar un marc de memòria, cosa que sovint comporta l'escriptura en disc, si les dades de la pàgina han estat modificades.

El procés queda en espera que la seva pàgina sigui carregada en un marc de memòria, i de mentre s'executa un altre procés, que, si el sistema està molt curt de memòria és fàcil que també tingui una fallida de pàgina.

Contra més processos hagin en execució més probable que tots tinguin fallides de pàgina perquè cap procés té el seu conjunt de treball disponible.

Aquesta situació encara pot empitjorar més, perquè el sistema detecta que l'ús de CPU és baix, i generalment incorpora més processos per aprofitar el recurs de CPU disponible, provocant la caiguda definitiva del sistema. Els sistemes més moderns tenen algunes proteccions contra aquesta última decisió suicida del sistema, però en un sistema amb poca memòria la hiperpaginació és una situació que apareixerà si es sobrecarrega y la memòria virtual es troba sobredimensionada.

Els sistemes en hiperpaginació no són capaços de respondre a les peticions dels programes o dels dispositius, donant-se la típica situació de no acceptar ordres per la consola.

Accions per evitar la hiperpaginació

- La hiperpaginació avui dia no és freqüent, degut a la gran quantitat de memòria que instal·len els ordinadors.
- La millor forma d'evitar-la és augmentant la memòria física.
- Si la memòria física és escasa i no és posible augmentar-la:
    a. És imprescindible mantenir la memòria virtual per sota del doble de la memòria física.
    b. S'ha de disminuir en tot el posible el nombre de processos i threads actius, per decrementar el nivell de multiprogramació; en un nivell de multiprogramació baix és més difícil que es doni la situació d'hiperpaginació.

Consorci d'Educació de Barcelona
Generalitat de Catalunya
Ajuntament de Barcelona

M1- Sistemes informàtics

UF1-Tema 10

Bios, Uefi i gestors d'arrencada

Índex de continguts

1 BIOS 2

Per què es diu que és només lectura si es pot actualitzar? 2
Si la BIOS és una memòria no volàtil Per a què necessita la pila? 2
Es poden canviar els valors de la memòria CMOS? i esborrar-la? 3
Com intervé la BIOS durant l'arrancada del PC? 4
Post BEEP codes 5

2 UEFI 9

Quines millores presenta UEFI respecte de BIOS? 9
Com puc saber si el meu ordinador disposa de BIOS o UEFI? 9

3 Accedir i configurar la BIOS/UEFI 10

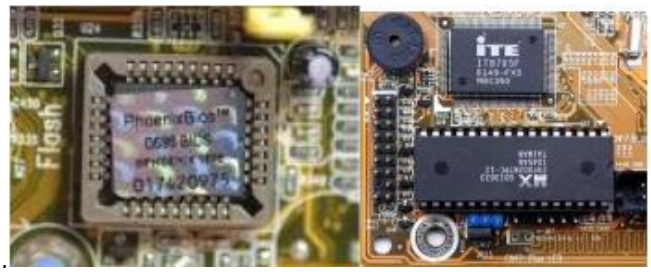
Secció 'Main' de la UEFI BIOS: 10
Secció 'Advanced' de la UEFI BIOS: 11
Secció 'Chipset' de la UEFI BIOS: 12
Secció 'Boot' de la UEFI BIOS: 13
Secció 'Security' de la UEFI BIOS: 14
Secció 'Save & Exit' de la UEFI BIOS: 15

4 Gestors d'arrencada. 16

Quins tipus de gestors d'arrencada existeixen? 16
Com es pot configurar Boot Configuration Data? 18
Com es pot configurar Grub? 21

1 BIOS

La BIOS (Basic Input Output System) és un firmware (és a dir, una porció de codi emmagatzemada en una memòria de només lectura) encarregat de iniciar els components de maquinari, habilitar funcions d'energia i temperatura i carregar el sistema operatiu d'un ordinador quan l'engeguem.

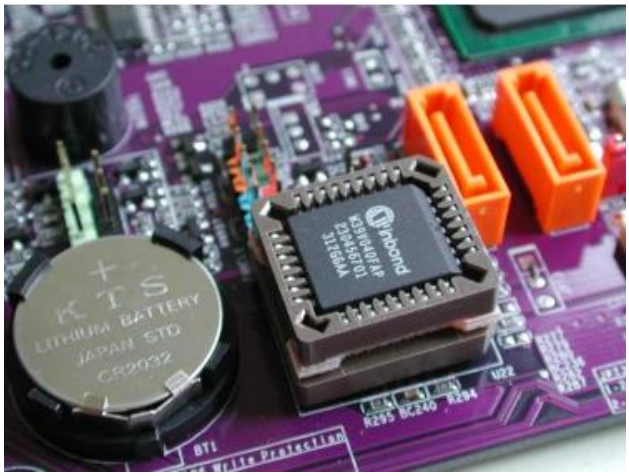


Per què es diu que és només lectura si es pot actualitzar?

En primer lloc, les primeres BIOS no es podien modificar. Si volies afegir qualsevol instrucció havies de canviar el Xip. Després, van aparèixer les primeres memòries PROM (Programmable ROM, tot i que només es podien programar un cop), després les EPROM (Erasable PROM, programables moltes vegades mitjançant raig ultraviolat), després les EEPROM (Electrical EPROM, programables elèctricament).

Actualment la majoria de BIOS són memòries Flash que poden actualitzar-se via software. Tot i que poden actualitzar-se, el 99% de les vegades que fem servir una BIOS serà per arrencar un ordinador i, per tant, per LLEGIR el seu contingut.

Si la BIOS és una memòria no volàtil Per a què necessita la pila?

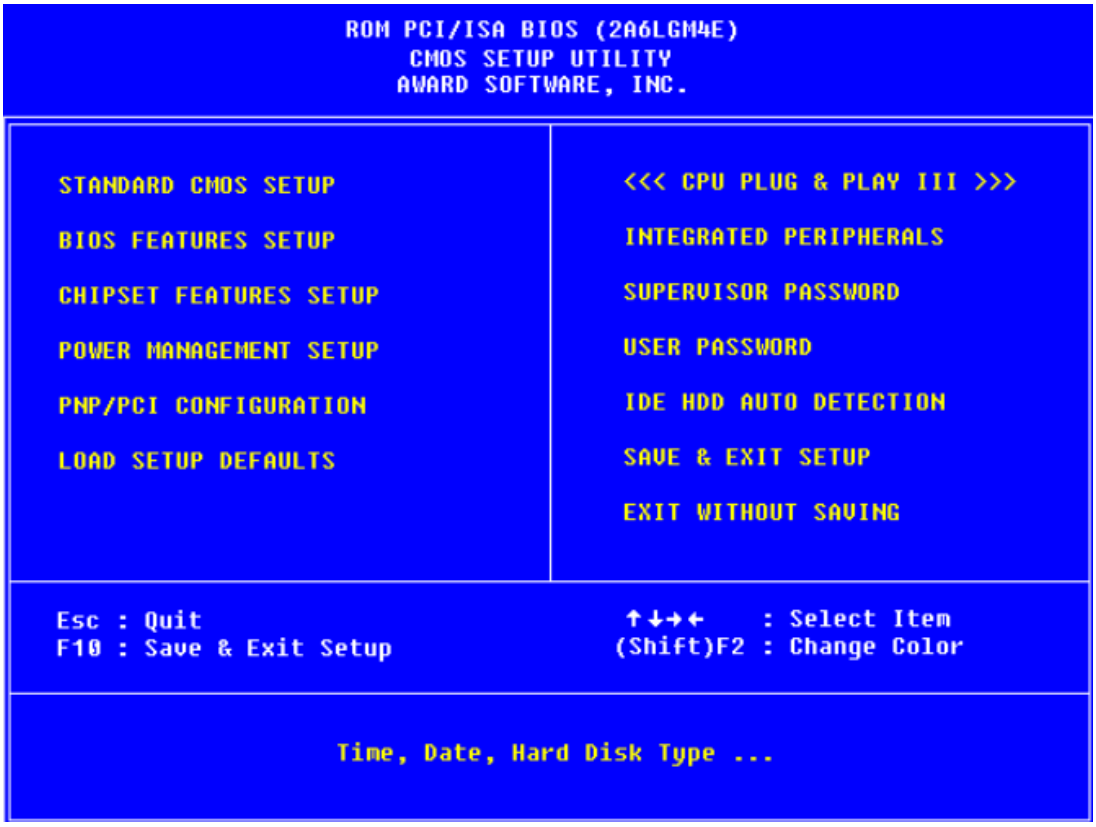


La Bios no requereix la pila per mantenir les seves instruccions. Si apaguem l'ordinador el codi de la BIOS es manté, sense cap problema. La pila serveix per alimentar la memòria no volàtil CMOS que emmagatzema les variables que utilitzarà la BIOS durant l'arrencada (Data, hora, freqüència de bus, seqüència d'inici de dispositius, contrasenya de BIOS, etc.) Si traiem la pila, perdrem el contingut de

la memòria CMOS, no el de la BIOS, Per tant, encara que és mol freqüent, no és correcte dir que la pila manté la informació que guarda la BIOS. La BIOS, no obstant això, guarda uns valors per assignar per defecte als paràmetres de la CMOS, en cas de pèrdua per falta d'energia.

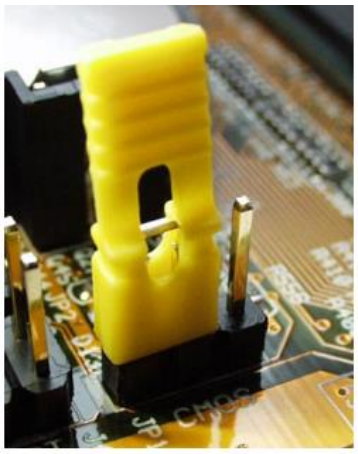
Es poden canviar els valors de la memòria CMOS? i esborrar-la?

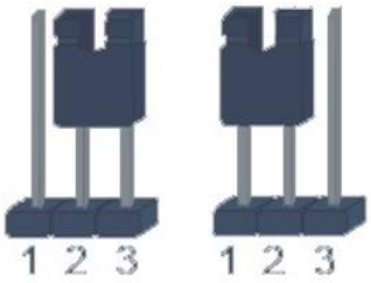
Es pot fer, mitjançant el menú de la BIOS, o més ben dit, el CMOS Setup.



Per esborrar el seu contingut de la CMOS (el que porta com a resultat omplir amb els valors per defecte els seus paràmetres) tenim a la nostra disposició tres formes de fer-ho:

1. Traient la pila (com hem comentar ja abans)
2. Seleccionant al menú l' opció LOAD SETUP DEFAULTS
3. Posant el jumper “CLEAR CMOS” a la posició d'esborrat (per fer això hem de consultar el nostre manual de la placa base).





Com intervé la BIOS durant l'arrancada del PC?

1. La placa Base s'alimenta elèctricament Quan la tensió elèctrica s'estabilitza el processador busca la primera instrucció a una determinada direcció de la memòria BIOS.

---CODI BIOS---

2. El processador executa una sèrie d'instruccions denominades POST (Power on Self Test) emmagatzemades a la BIOS que permeten comprovar els components indispensables per a que el sistema informàtic pugui funcionar (aquestes instruccions utilitzen diferents valors que són carregats a partir de la memòria CMOS):

a) Es verifica una primera part de la memòria RAM (RAM Baixa).

b) S'inicia el dispositiu de vídeo Aquest dispositiu té la seva pròpia BIOS.

--CODI BIOS VIDEO--

El processador processarà totes les instruccions d'inicialització del dispositiu de vídeo contingut a la BIOS vídeo A partir d'aquest moment ja es poden visualitzar missatges al monitor.

--FINAL CODI BIOS VIDEO--

c) S'inicia el controlador de teclat.

d) Es verifica la resta de la memòria RAM.

e) Es verifiquen els dispositius presents (CDROM, Discs Durs, etc...)

f) Si la Bios i els dispositius suporten Plug and Play (Pnp) es configuren automàticament (interrupcions, Memòria, Entrades/Sortides, velocitat, etc.).

g) Es mostra un informe mitjançant el monitor.

h) S'espera durant un breu interval de temps per veure si l'usuari vol entrar o no al menú de CMOS setup o be varia l'ordre de la "boot sequence".

i) S'inicia el dispositiu prioritari (USB->CDROM->HD1->HD2->USB →LAN) i busca el MBR (Master Boot Record, el primer sector del disc, que conté els paràmetres per arrencar el sistema) o bé el GPT (si es tracta d'una UEFI en lloc d'una BIOS). Un cop troba el MBR (o el GPT) Ha acabat la feina de la BIOS.

--- FINAL CODI BIOS---

3. El processador executa el codi contingut al MBR (o GPT). Aquest codi, bàsicament, mira com està dividit el disc dur (particions) i busca la partició activa. Dins aquesta partició, busca el primer sector que és on es troba el codi que iniciarà el Sistema Operatiu instal·lat a la partició.

Post BEEP codes

Com s'ha dit abans, La BIOS fa un anàlisi de tots els dispositius imprescindibles per poder fer servir el sistema informàtic. Si troba cap problema retorna un o més sons i mostra un missatge per pantalla (aquest missatge només es podrà veure si s'ha pogut carregar la targeta gràfica)

AMI (American Megatrends International): AMI fa servir sons de la mateixa longitud.

BEEP CODE	EXPLICACIÓ	POSIBLE CAUSA	QUE FEM?
1 Beep	Error en el refresc de la memòria	Memòria RAM defectuosa	
2 Beep	Error en la paritat de memòria	Memòria RAM defectuosa	Canviarem la memòria RAM de posició als sócols que ocupa ja que existeix un problema de paritat, o als primers 64Kbytes de memòria Si el problema persisteix canviarem la placa.
3 Beep	Base 64K mem error	Memòria RAM defectuosa	
4 Beep	Contador no operatiu	Placa base defectuosa	
5 Beep	Error de processador	Processador defectuós	
6 Beep	8042 Gate A20 error	Processador / Placa base defectuosos	
7 Beep	Excepció del processador	Processador defectuós	Canviem el processador o revisam la configuració
8 Beep	Error a la memòria de vídeo	Tarja gràfica o memòria gràfica defectuosa	Canviem de slot la tarja o revisem la memòria de vídeo
9 Beep	ROM checksum error	BIOS defectuosa	Procedim a 'flaseJarla' si podem, o a canviar el xip.
10 Beep	CMOS checksum error	CMOS/Placa base defectuosa	Intentem esborrar aquestes dades fent un pont amb el jumper "Clean CMOS" o traient la pila, i intentem salvar els valors de nou en la CMOS. Si el problema persisteix tindrem que canviar la placa ja que aquest xip està sempre soldat.
11 Beep	caxé errònia	Placa base/processador defectuós	Podes reactivar la caxé mitjançant la combinació Control + Alt + Shift + I

Award Bios Beep Codes. Award Bios: Fa servir sons de diferent duració.

BEEP CODE	EXPLICACIÓ	POSIBLE CAUSA	QUE FEM?
To ininterromput	Fallada en el subministre elèctric	Font d'alimentació defectuosa	Revisem connexions i font d'alimentació
1 llarg	Error en la memòria RAM	Error en la memòria RAM o en la placa	Canviem de banc la memòria i comprovar jumper de bus.
1 llarg 1 curt			Provarem a 'flasejar' o reemplacem el xip de la BIOS sinó podem canviar de placa.
1 llarg 2 curts	Error en la tarja gràfica	Tarja gràfica defectuosa	Si apareix per pantalla aquest missatge:"No vídeo card found", aquest error és només aplicable a plaques base amb tarja de vídeo
			integrada. Inhabilitarem la placa (mitjançant el CMOS Setup) i introduirem una nova tarja de vídeo en un slot.
Bucle infinit de sons	Error a la memòria RAM	Memòria defectuosa o malament connectada	
1 llarg 3 curts	Error a la tarja gràfica	Error a la tarja gràfica o a la memòria de la tarja	
sons d'alta freqüència	CPU s'està sobreescalfant	Error al ventilador de la CPU	
Bucle de sons aguts i baixos	Error al processador	Processador defectuós	

Phoenix Bios Beep Codes. Phoenix Bios fa servir patrons de sons. El guió representa una petita pausa entre sons.

BEEP CODE	EXPLICACIÓ	POSIBLE CAUSA
1 - 1 - 2	CPU / error de placa base	CPU / placa base defectuosa
1 – 1 - 3	CMOS error de lectura/escriptura	placa base defectuosa
1 - 1 – 4	BIOS ROM error	BIOS defectuosa
1 – 2 - 1	Error de comptador	placa base defectuosa
1 – 2 - 2	DMA error	placa base defectuosa
1 – 2 - 3	DMA error	placa base defectuosa
1 – 3 - 1	Error en el refresc de la memòria	Memòria RAM defectuosa
1 – 3 - 2	Error a la memòria baixa (64K)	Memòria RAM defectuosa
1 – 3 - 3	Error a la memòria baixa (64K)	Memòria RAM defectuosa
1 – 3 - 4	Error a la memòria baixa (64K)	Memòria RAM defectuosa
1 – 4 - 1	Error en el direccionament	Memòria RAM defectuosa
1 – 4 - 2	Error de paritat	Memòria RAM defectuosa
1 – 4 - 3	Error de comptador	placa base defectuosa
2 - * - *	Error a la memòria baixa (64K)	Memòria RAM defectuosa
3 – 1 - 1	Error en DMA esclau	placa base defectuosa
3 – 1 - 2	Error en DMA mestre	placa base defectuosa
3 – 1 - 3	Error al controlador d'Interrupcions	placa base defectuosa
3 – 1 - 4	Error al controlador d'interrupcions esclau	placa base defectuosa
3 – 2 - 2	Error al controlador de interrupcions	placa base defectuosa
3 – 2 - 4	Error al controlador del teclat	placa base defectuosa
3 – 3 - 1	Pila de la CMOS defectuosa	Pila de la CMOS defectuosa
3 – 3 - 2	CMOS error configuració	error configuració
3 – 3 - 4	Error a la memòria de vídeo	Tarja o memòria gràfica defectuosa
3 – 4 - 1	Error a la inicialització de la Tarja gràfica	Tarja o memòria gràfica defectuosa
4 – 2 - 1	Error de comptador	placa base defectuosa
4 – 2 - 2	Error durant el reseteig de CMOS	placa base defectuosa
4 – 2 - 3	Gate A20 error	placa base defectuosa
4 – 2 - 4	Interrupció inesperada	Processador defectuós
4 – 3 - 1	Error a la comprovació de la RAM	Memòria RAM defectuosa
4 – 3 - 3	Error al comptador	placa base defectuosa
4 – 3 - 4	RTC error	placa base defectuosa
4 – 4 - 1	Error al port serie	placa base defectuosa
4 – 4 - 2	Error al port paral·lel	placa base defectuosa
4 – 4 - 3	Error al coprocessador	placa base defectuosa o CPU
9 – 2 - 1	Tarja gràfica no compatible	Tarja gràfica no compatible

Altres vegades no escoltem cap so i la pantalla mostra qualsevol d'aquest missatges. són comuns a totes les BIOS.

***BIOS ROM checksun error – system halted:** el codi de control de la BIOS és incorrecte, això pot significar que està corrupta. En cas de reiniciar i repetir aquest missatge tindrem que canviar la BIOS.

***CMOS battery failed:** La pila de la placa base que alimenta la memòria CMOS ha deixat de subministrar corrent. és necessari canviar la pila immediatament

***CMOS checksum error – Defaults loaded**: El codi de control de la CMOS no és correcte, es procedeix a carregar els paràmetres per defecte. Aquest error és produeix perquè la informació emmagatzemada en la CMOS és incorrecta, pot indicar que la pila comença a fallar.

***Display switch is set incorrectly:** El tipus de pantalla especificada a la BIOS és incorrecta. Això pot passar si hem seleccionat l' existència d'un adaptador blanc i negre quan tenim un a color, o be al contrari.

***Floppy disk(s) Fail (code 40/38/48 dependent de l' antiguitat de la bios):** Disquetera mal connectada, verifiquem tots els cables de connexió.

***Hard disk install failure:** La BIOS no pot inicialitzar o trobar el disc dur. Tenim que estar segurs de que tots els discs han sigut degudament connectats.

***Keyboard error or no keyboard present:** no és possible inicialitzar el teclat. Pot ser no es troba connectat o be pot estar trencat. També podria ser degut a que haguessin mantingut pitjada una tecla durant el procés d'arrancada.

***Keyboard error is locked out – Unlock the key:** Es mostra a poques bios. Apareix quan una tecla queda bloquejada.

***Memory Test Fail:** La verificació de memòria RAM ha fallat degut probablement a errors als mòduls de memòria. Si apareix aquest missatge hem de tenir molta precaució amb l'equip, el qual es pot tornar inestable i causar pèrdues de dades. La solució és verificar els bancs de memòria i substituir aquests bancs de ser necessari.

***Override enabled – Defaults loaded:** Si el sistema no pot iniciar-se amb els valors emmagatzemats a la CMOS, la BIOS pot creure convenient substituir aquest valors per una sèrie de paràmetres per defecte dissenyats per que l'equip funcioni de forma estable, tot i que no funcionarà de forma optima.

***Primary master hard diskfail:** el procés d'arrancada ha detectat una fallada durant l'inici del disc mestre al controlador IDE primari. Per solucionar aquest problema, comprovarem, les connexions del disc i la configuració de la BIOS.

2 UEFI

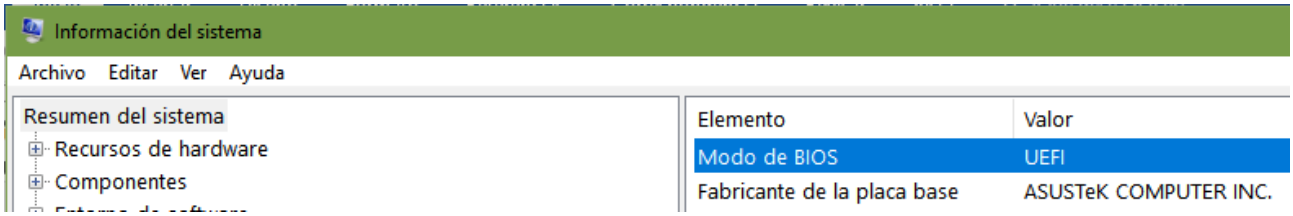
UEFI Unified Extensible Firmware Interface (UEFI), és una especificació que substitueix la anterior BIOS. És a dir, realitza la mateixa funció que aquesta però aportant una sèrie de millores molt remarcables.

Quines millores presenta UEFI respecte de BIOS?

- Compatibilitat per a sistemes operatius només compatibles amb BIOS.
- Substitueix les taules de particions MBR per les taules de particions GPT. Això suposa que es poden crear fins a 128 particions per disc (només 4 a MBR) cadascuna amb una capacitat total de fins a 8ZB (2,2TB a MBR).
- Permet arrancar amb unitats d'emmagatzematge més grans.
- Millores a la interfície de modificació de la BIOS/UEFI (en moltes ocasions disposen d'interfície gràfica).
- Disseny d'arquitectura compatible amb 32 i 64 bits (MBR disposa d'arquitectura 16 bits) Això suposa que UEFI accepti incorporació de nous mòduls que milloren les característiques de configuració del sistema.
- Incorpora característiques avançades de BIOS, com ACPI (Interfaz Avanzada de Configuración y Energía) y el SMBIOS (Sistema de Gestión de BIOS).

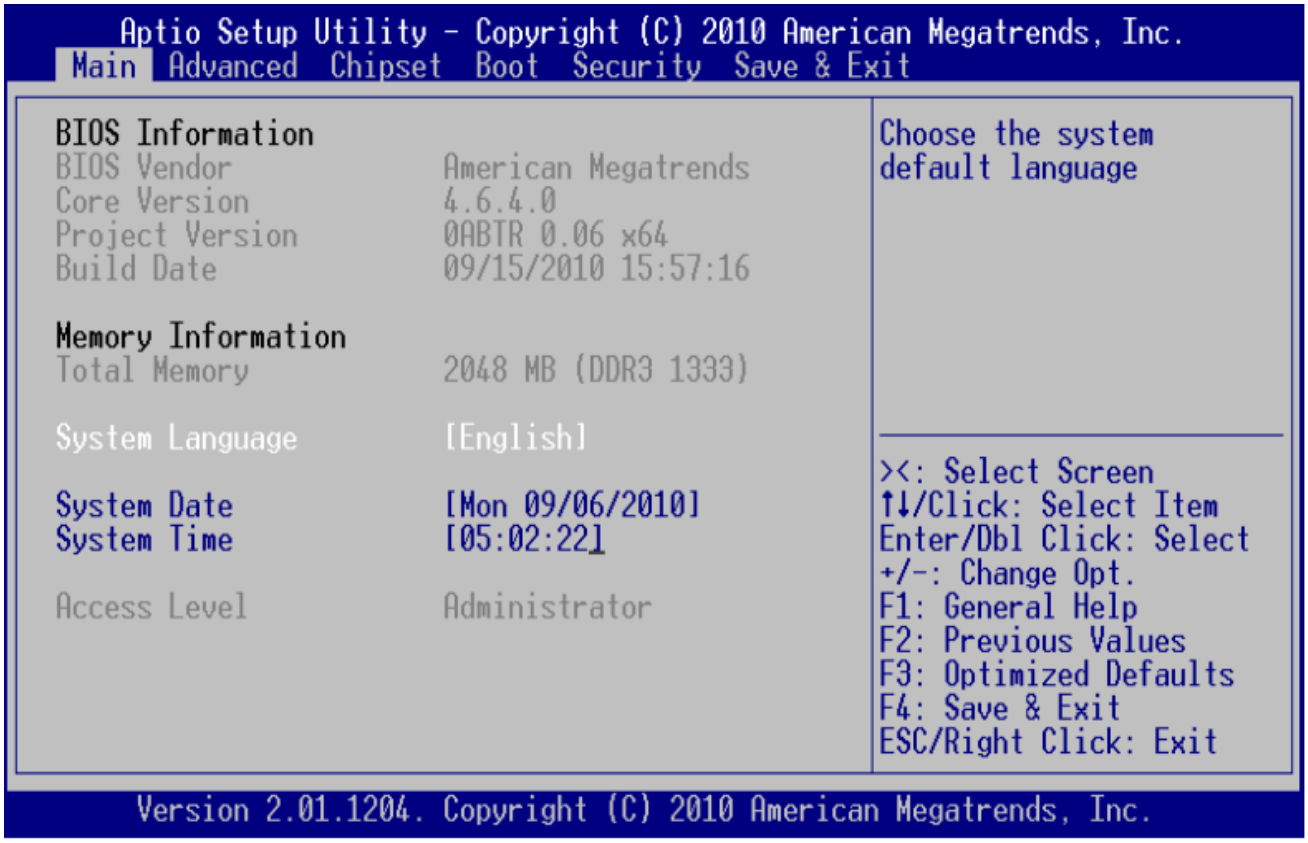
Com puc saber si el meu ordinador disposa de BIOS o UEFI?

Es pot consultar l'eina 'Configuració del sistema', msinfo32. A la secció 'Recursos del sistema' → 'Modo de BIOS'



3 Accedir i configurar la BIOS/UEFI

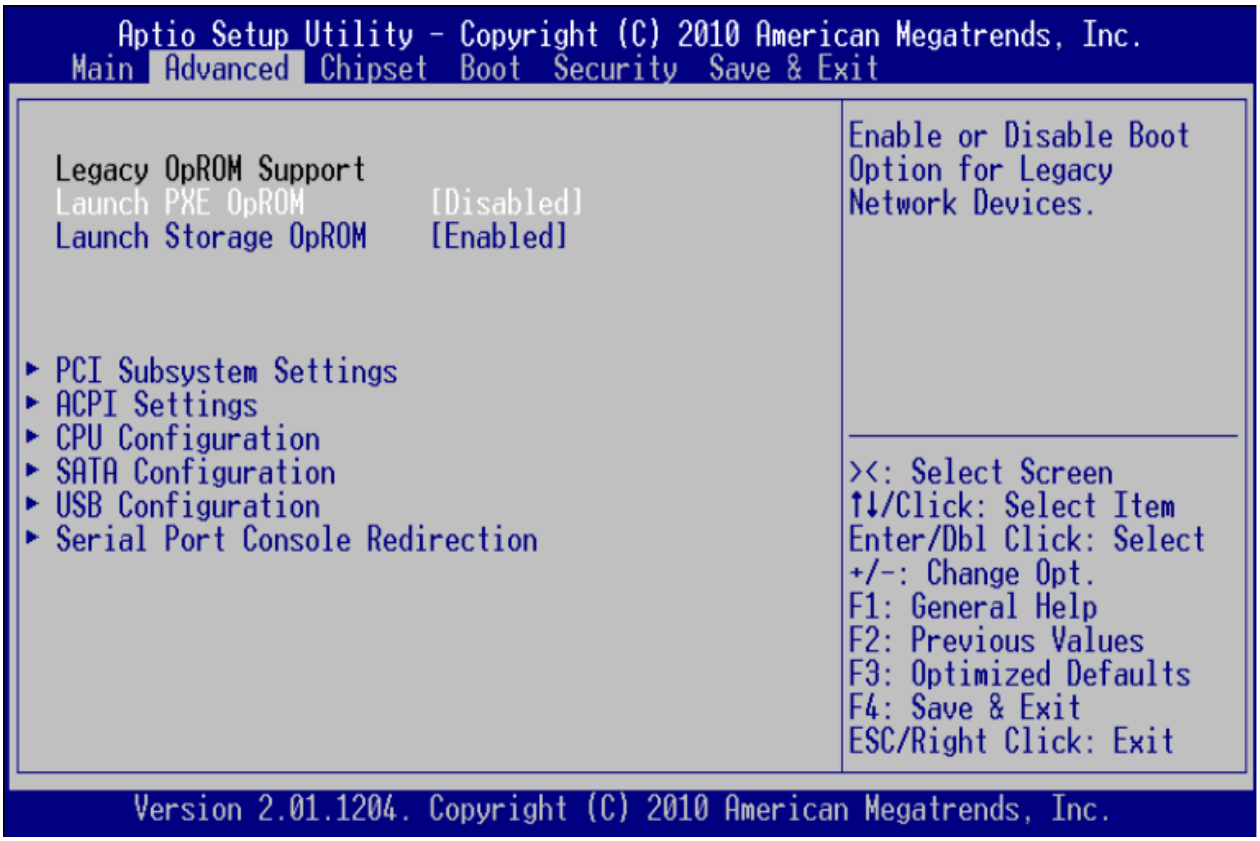
Per accedir a la BIOS/UEFI hem de reiniciar l'ordinador i polsar una de les tecles següents (pot variar en funció de la màquina): F2, F10, Supr, o qualsevol altra indicada al manual del sistema informàtic en qüestió. Apareixerà una interfície com la següent:



Secció 'Main' de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

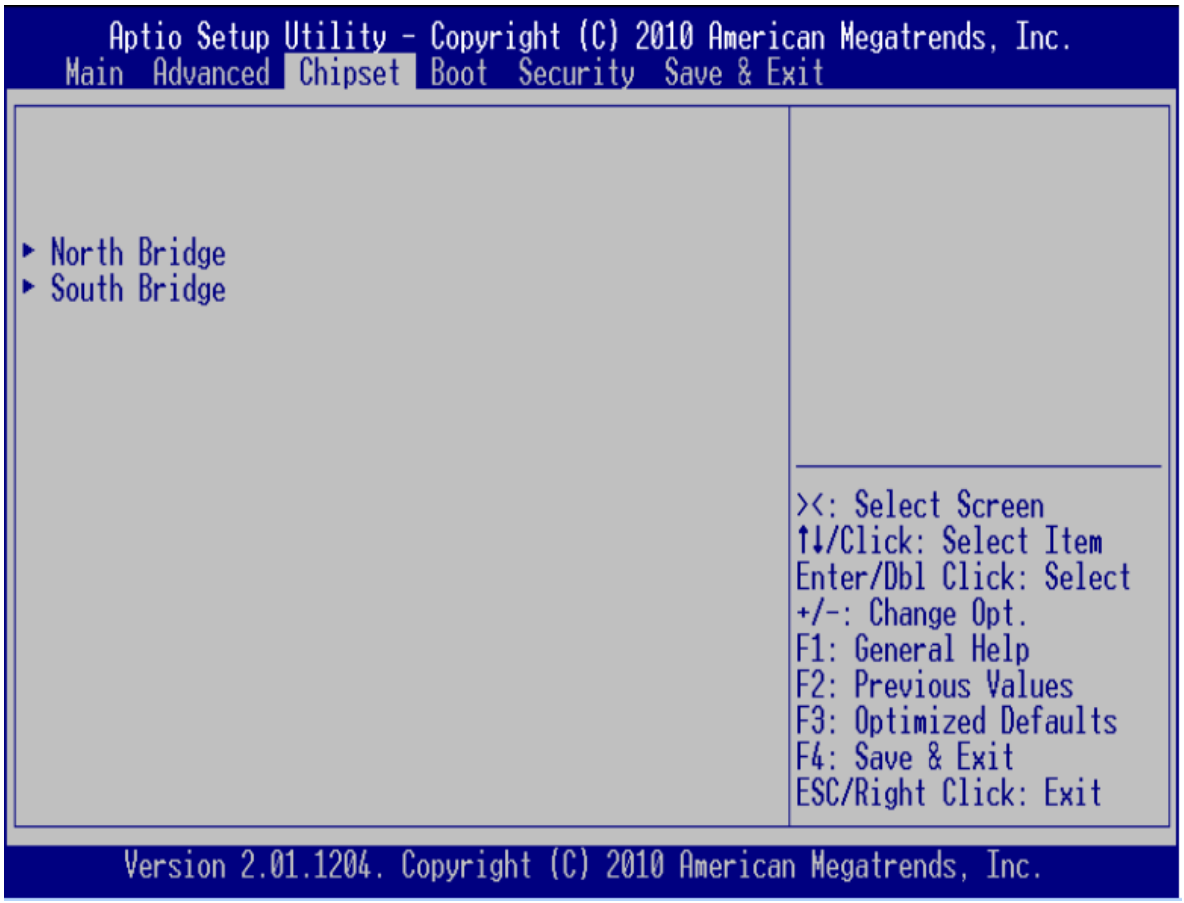
- Fabricant, versió i data de construcció de la BIOS/UEFI.
- Quantitat total de memòria RAM del dispositiu.
- Idioma per defecte de la BIOS/UEFI.
- Data i hora del sistema.
- Seqüència de tecles per tal d'interactuar amb la interfície.



Secció 'Advanced' de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

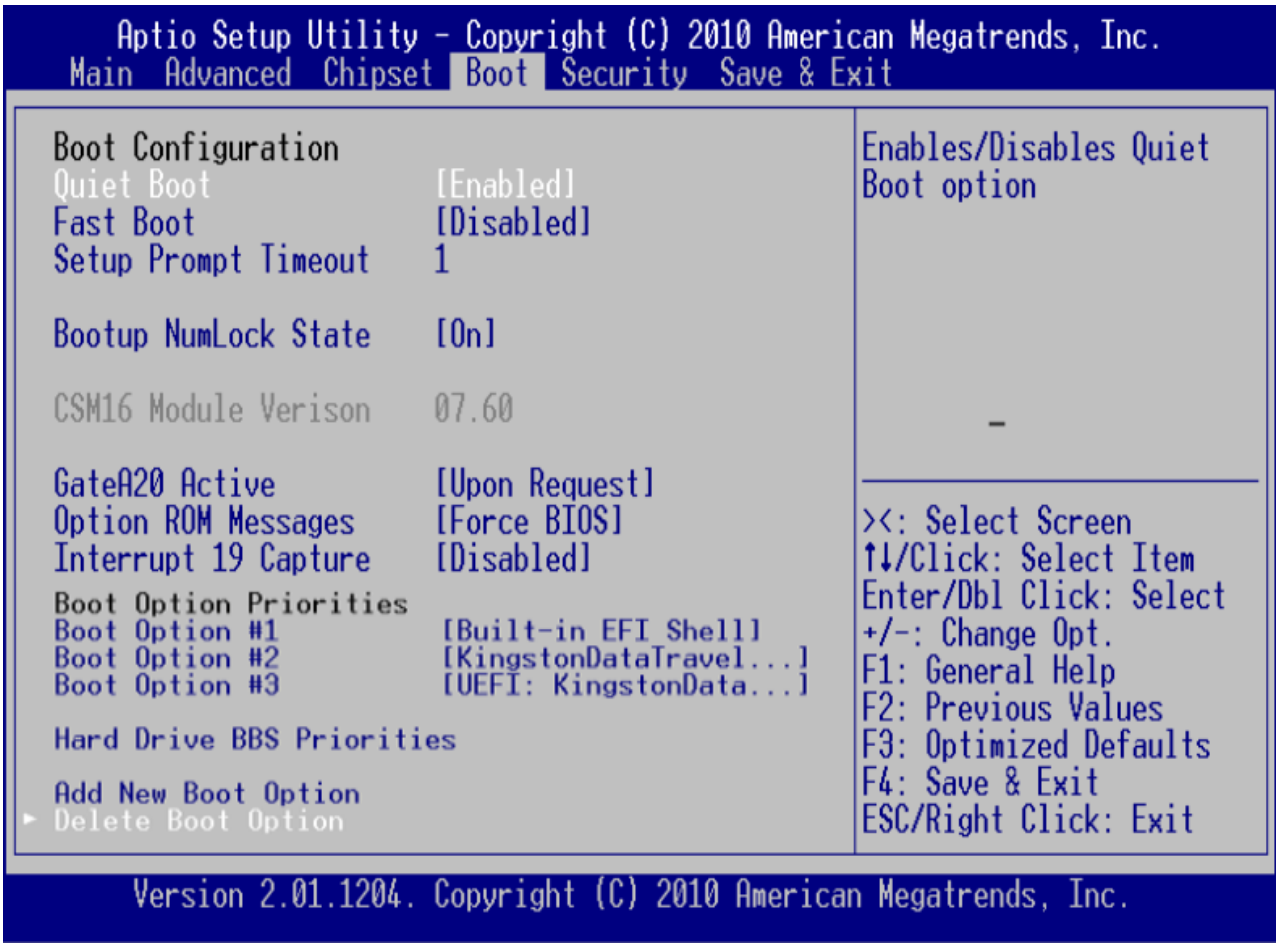
- **Launch PXE OpRom:** Habilita o inhabilita els controladors necessaris per tal d'arrancar l'ordinador o instal·lar un sistema operatiu a la màquina mitjançant una xarxa.
- **Launch Storage OpRom:** Habilita o inhabilita controladors necessaris per tal d'accedir a dispositius d'emmagatzematge.
- **PCI Subsystem Settings:** Permet configurar paràmetres dels bussos PCIe de la placa.
- **ACPI Settings:** Permet configurar paràmetres de la interfície Avançada de configuració d'energia.
- **CPU Configurations:** Permet configurar paràmetres relacionats amb la CPU del PC (per exemple, habilitar característiques de virtualització de la CPU).
- **SATA Configuration:** Permet configurar paràmetres dels bussos SATA de la placa.
- **USB Configuration:** Permet configurar paràmetres dels ports USB del PC.
- **Serial Port Console Redirection:** Permet configurar el port sèrie del PC.



Secció 'Chipset' de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

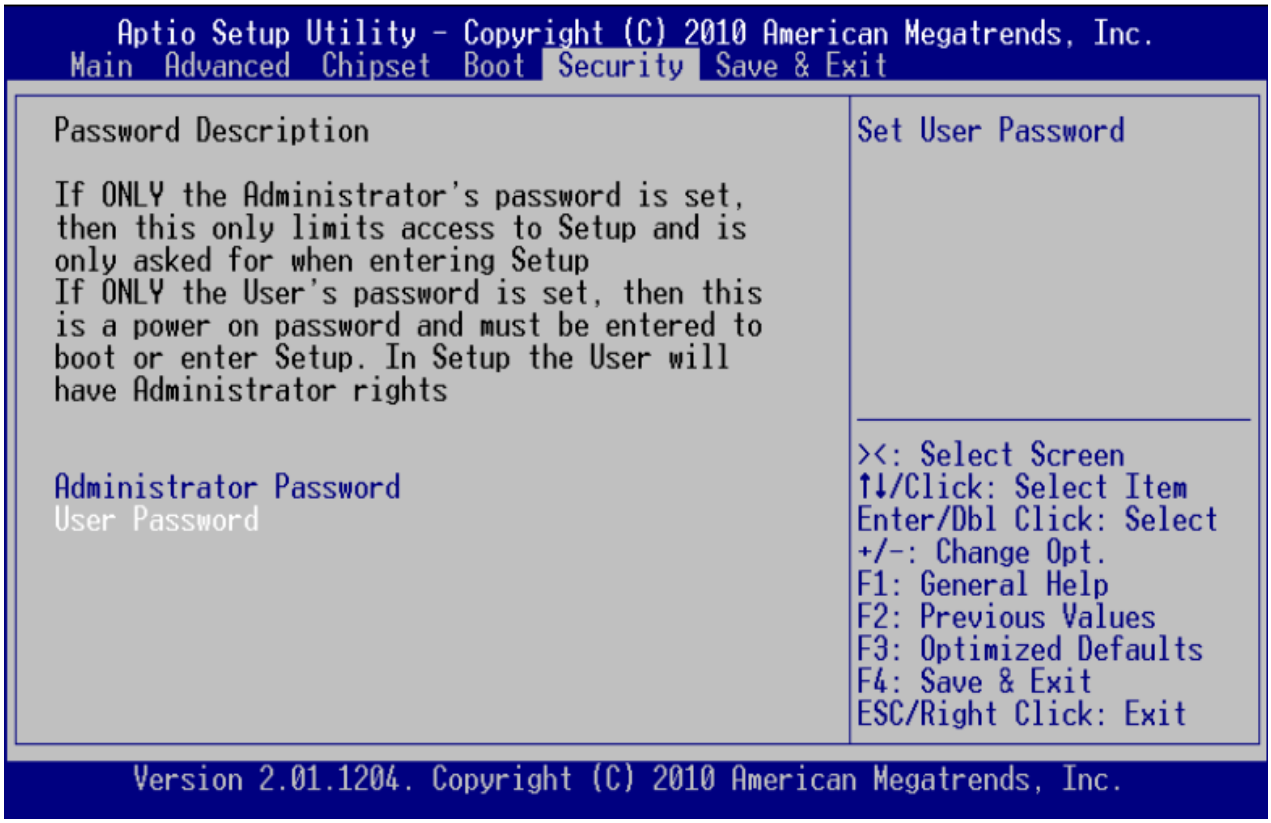
- North Bridge: Accedeix als paràmetres propis del xipset nord de que disposa la màquina, alguns d'aquest són:
    - Tipus de CPU: Mostra el nom clau (Lynnfield, Clarkdale, Wolfdale,...)
    - Mostra la quantitat de memòria assignada a cada slot de la placa base.
    - Mostra dades de la latència de les memòries assignades a cada slot.
    - Mostra dades del controlador gràfic integrat a la placa base.
    - Permet seleccionar el controlador gràfic: PEG (PCI Express Graphics) o IGD (Integrated Graphics Device).
- South Bridge: Accedeix als paràmetres propis del xipset sud de que disposa la màquina, alguns d'aquest són:
    - Habilitar/inhabilitar divers dels busos PCI Express o SMBus (per a poder carregar automàticament la bateria del PC).
    - Habilitar/inhabilitat wake on lan (iniciar l'ordinador a través de xarxa).
    - Habilitar/Inhabilitar ports PCIe i USB.
    - Permet seleccionar el controlador d'Àudio intern o extern.



Secció 'Boot' de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

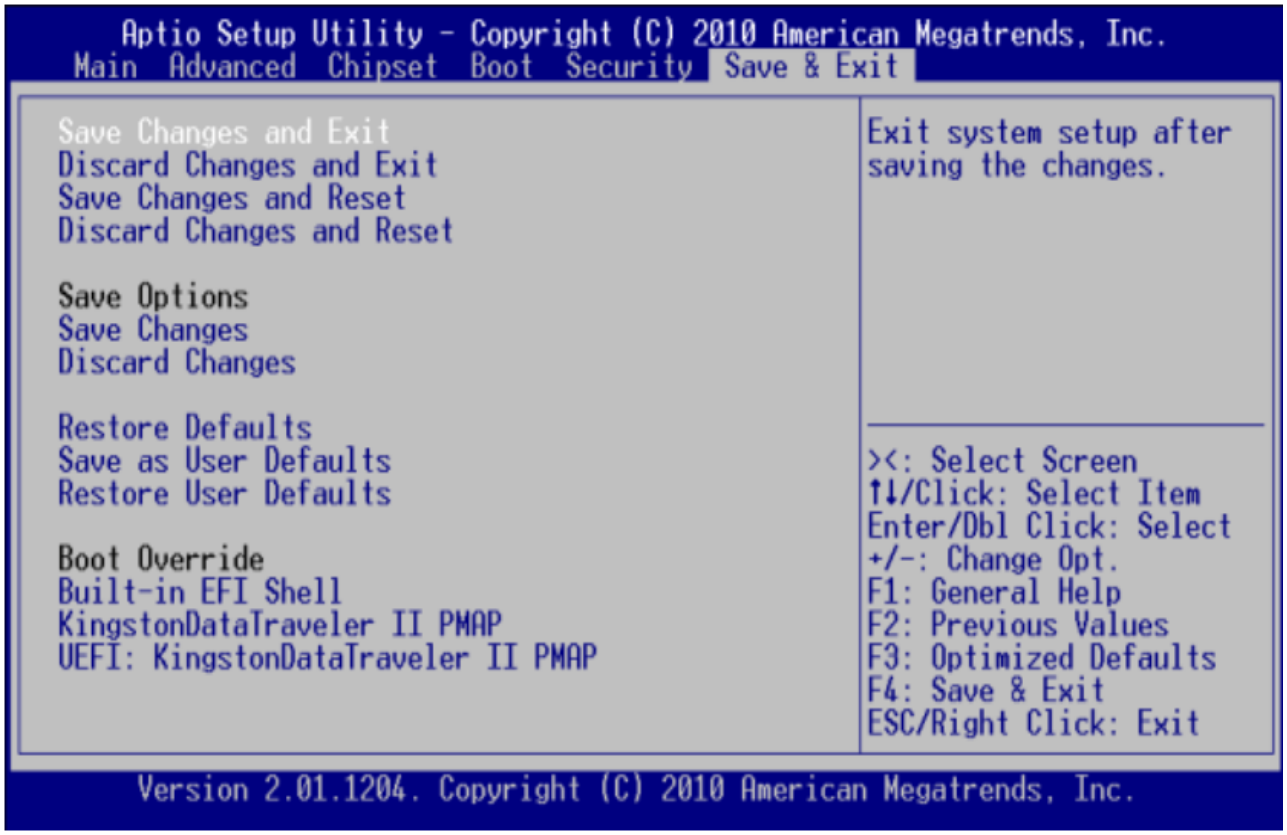
- Quiet Boot: Si està habilitat, oculta els resultats de tests del POST a la pantalla d'inici de l'ordinador.
- Fast Boot: Si està habilitat, evita realitzar determinats tests avançats de l'estat de la memòria durant el POST de la màquina.
- Setup Prompt Timeout: Pausa l'inici un número determinat de segons per a que es puguin veure els missatges del POST.
- BootUp NumLock State: Especifica l'estat de la tecla 'numlock' durant l'inici.
- Boot Option Priorities: Conté una llista jeràrquica d'arxius o perifèrics per tal de tractar d'iniciar el sistema.



Secció 'Security' de la UEFI BIOS:

En aquesta secció podem configurar dos nivells d'accés a la BIOS, l'administrador (amb accés total) i l'usuari (amb algunes característiques restringides) podem trobar la següent informació:

- Administrator Password: Configura la clau de pas d'administrador.
- User Password: Configura la clau de pas d'usuari.



Secció 'Save & Exit' de la UEFI BIOS:

En aquesta secció podem trobar la següent informació:

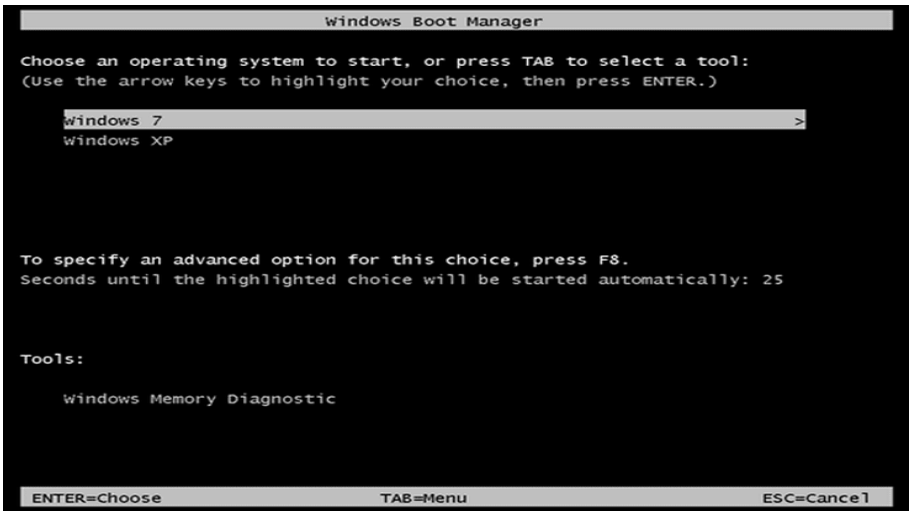
- **Save changes and Exit:** Surt de la configuració de la BIOS/UEFI guardant els canvis i tancant la màquina.
- **Discard changes and Exit:** Surt de la configuració de la BIOS/UEFI sense guardar els canvis i tancant la màquina.
- **Save Changes and Reset:** Surt de la configuració de la BIOS/UEFI guardant els canvis i reiniciant la màquina.
- **Discard Changes and Reset:** Surt de la configuració de la BIOS/UEFI sense guardar els canvis i reiniciant la màquina.
- **Restore Defaults:** Restaura els valors de la BIOS/UEFI als valors de fàbrica.
- **Boot Override:** Mostra i configura els diferents arxius d'inici que poden utilitzar-se per a inicialitzar el sistema mitjançant la llista present a 'Boot Option Priorities'.

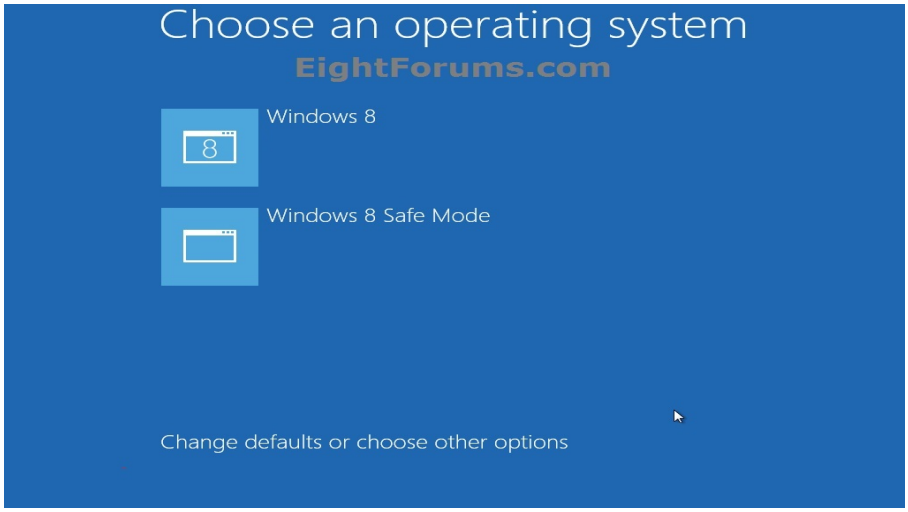
4 Gestors d'arrencada.

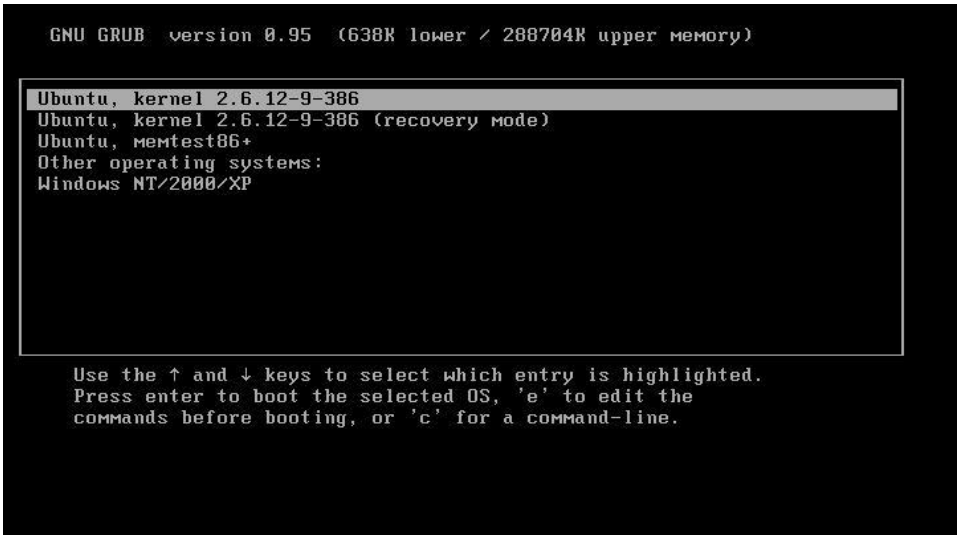
Els gestors d'arrencada (també anomenats bootloaders, boot programs o bootstrap loaders) són programes dissenyats per tal de gestionar l'inici d'un o varis sistemes operatius presents a un sistema informàtic. Normalment aquests programes s'ocupen de l'inici del sistema un cop la BIOS/UEFI ha executat el POST i ha obtingut la ubicació del MBR (o GPT en cas de ser una UEFI).

Quins tipus de gestors d'arrencada existeixen?

Podem trobar diferents tipus de gestors d'arrencada, però els que estan directament relacionats amb els sistemes informàtics són els anomenats **Gestors d'arrencada de segona etapa.** Aquest són un tipus de programari enfocat a la cerca al sistema informàtic d'unitats d'emmagatzematge que puguin arrencar el sistema, fent servir una petita aplicació emmagatzemada en aquesta. Els gestors d'arrencada més coneguts d'aquest tipus són: LILO (ja obsolet), GRUB (orientats a l'arrencada de sistemes operatius lliures) i Windows Boot Manager (orientat a l'arrencada de sistemes operatius privatius Windows).







Com es pot veure als gràfics anterior, els gestors d'arrencada permeten mantenir diversos sistemes operatius en un ordinador i alternar el procés d'arrencada. L'usuari escull aquest procés per mitjà d'un menú que apareix per pantalla. Una vegada iniciat un sistema operatiu, si volem fer un canvi, haurem de reiniciar la màquina.

Una de les condicions principals és que el gestor d'arrencada escollit resulti adequat per als sistemes operatius que vulguem gestionar. Malgrat que els gestors d'arrencada tenen les mateixes funcions, n'hi ha de diferents tipus:

1. Gestors d'arrencada que s'instal·len obligatòriament en l'MBR. Aquest tipus instal·la, en primer lloc, els arxius propis del programa en una partició del disc dur i el gestor s'instal·la en l'MBR. És interessant poder crear disquets d'inici, per recuperar el gestor en cas que s'esborri l'MBR. Dins aquest grup de programari, trobaríem el gestor Grub.

2. Gestor d'arrencada Windows (fins a Windows XP). En instal·lar un sistema operatiu Windows, es desen els arxius necessaris per a l'arrencada: ntldr, ntdectect.com i boot.ini (és l'arxiu de configuració del selector d'arrencada). En la partició primària, a mesura que hi instal·lem més sistemes operatius, també s'hi desen els fitxers dels sistemes operatius implicats, per la qual cosa no es podran arrencar directament, sinó que, de primer, caldrà arrencar la partició que conté els arxius d'arrencada i, des d'allà, executar el selector d'arrencada de Windows.

   En fer la instal·lació, es crea un arxiu boot.ini, en la partició activa. El gestor d'arrencada (NTLDR, llegeix l'arxiu boot.ini . Si té més d'un sistema operatiu, ens en permetrà seleccionar un. En cas que no fem cap selecció, de manera predeterminada, se'ns carrega l'últim que es va instal·lar.

3. Gestor d'arrencada Windows (Actual). A partir de Windows Vista, l'arxiu ntldr i boot.ini desapareixen i són reemplaçats per winload i Boot Configuration Data (o BCD, que en realitat és una base de dades en lloc d'un arxiu). Aquest nou sistema és més versàtil permetent:
   - Arrancada en sistemes UEFI.
   - La informació de BCD es guarda xifrada a la carpeta \boot ubicada al directori arrel de la partició "booteable".
   - Conté el menú Windows Boot Manager que permet:
     - Opcions per a solucionar problemes d'inici de Windows.
     - Triar el sistema operatiu Windows amb el que es vol iniciar.
     - Restaurar Windows a una versió prèvia.

Com es pot configurar Boot Configuration Data?

A aquesta part explicarem com realitzar les tasques més habituals en l'arrencada de l'equip fent servir bcdedit.exe (L'aplicació per defecte per a configurar BCD). Realitzarem les tasques més bàsiques per editar la llista de sistemes d'arrencada de Windows Vista i posteriors.

A partir d'una instal·lació dual de Windows haurem d'obrir una consola amb privilegis administratius. Per fer-ho a Windows 7, cal fer clic sobre el botó inici, tots els programes, seleccionar símbol del sistema amb el botó dret del ratolí i fer clic sobre executar com administrador. A partir de Windows 8 cercarem l'aplicació 'cmd' i un cop aparegui la seva icona amb el nom 'Indicador d'ordres' cal fer clic amb el botó dret i seleccionar executar com administrador.

Crear una còpia de seguretat

És imprescindible que abans de manipular el magatzem BCD de l'equip es faci una còpia de seguretat del mateix. Per fer-ho, des de la consola d'ordres amb privilegis administratius caldrà

executar:

```
bcdedit /export <NomFitxer>
```

(on <NomFitxer> és el nom que se li vol donar a la còpia de seguretat).

Eliminar una entrada del inici

Imaginem que hem eliminat físicament un sistema operatiu de l'equip, per exemple, esborrant la partició on estava instal·lat, però encara tenim la seva entrada corresponent al menú d'inici. Per eliminar l'entrada és convenient fer servir l'eina gràfica msconfig. Cal executar msconfig des del quadre de diàleg 'executar' (tecla Windows + 'R') o des de la consola amb privilegis d'administració.

Nota: Si Msconfig no llista l'entrada que es vol eliminar, és degut a que MSConfig només mostra els objectes del magatzem BCD del tipus "Cargador de arranque de Windows". Pots visualitzar el tipus de cada entrada amb l'ordre:

```
bcdedit /enum
```

Si és el cas, caldrà recórrer a l'eliminació "manual", fent servir Bcdedit.exe.

Tot i que aparegui l'entrada que es vol eliminar, és possible que el botó "Eliminar" estigui desactivat (en gris). Aquest botó es desactiva quant l'entrada que es vol eliminar és el sistema operatiu que està iniciat en aquest moment. Tècnicament Bcdedit.exe identifica aquesta entrada amb l'identificador {current}

Si l'entrada té simultàniament les propietats de ser el sistema operatiu iniciat en aquest moment i a més és el sistema operatiu per defecte {default} el seu identificador serà {current}.

Com s'ha comentat anteriorment, l'eina msconfig és possible que no ens mostri l'entrada que volem treure. L'escenari més comú és tenir instal·lat simultàniament un sistema operatiu anterior a Windows 7, com pot ser XP. L'entrada d'aquests objectes al magatzem BCD té un tipus diferent de "Carregador d'arrencada de Windows", concretament tenen tipus "Carregador del SO Windows heretat" i habitualment fan servir l'etiqueta {ntldr}. En aquests casos no queda més remei que fer servir Bcdedit.exe, de la següent manera:

Per eliminar l'entrada amb l'etiqueta {ntldr}

```
bcdedit /delete {ntldr} /f
```

(El modificador /f cal fer-lo servir sempre que s'utilitzi un GUID conegut, com {ntldr} i similars).

Per eliminar qualsevol altra entrada:

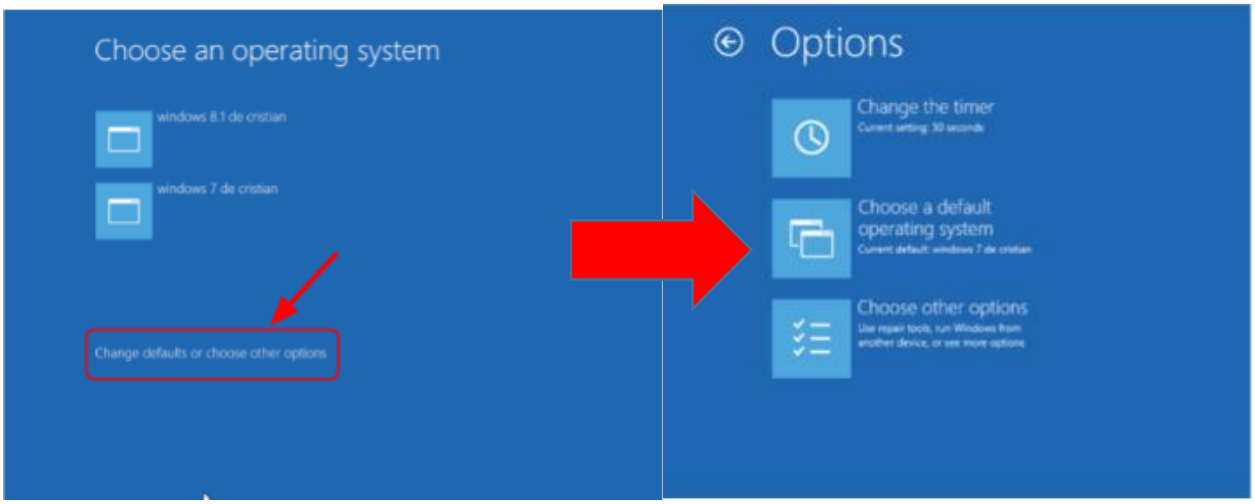
En primer lloc cal enumerar en primer lloc tots els objectes que formin part del magatzem BCD:

bcdedit /enum Ens fixarem en l'atribut Descripción, que és el text que mostra el menú d'arrencada, per identificar l'entrada que volem eliminar. A partir d'aquí, anotem el valor corresponent al seu atribut Identificador. Per exemple, si l'atribut Identificador contingués {afb0516d-f5bd-11dd-8879-98b0f6447174}, per eliminar-la només caldria executar la comanda:

```
bcdedit /delete {afb0516d-f5bd-11dd-8879-98b0f6447174}
```

Canviar el sistema que inicia per defecte

Com sap, en un entorn amb múltiples sistemes operatius instal·lats, si l'usuari no tria cap durant un temps definit, s'inicia un per defecte. Per canviar el sistema operatiu per defecte, podem utilitzar msconfig. Seleccionem el sistema operatiu en qüestió i premem sobre Selecciona per defecte.



Canviar el text de l'entrada del menú d'inici

Canviar el text descriptiu d'un sistema operatiu al magatzem BCD és una tasca comuna, especialment per a la gent que instal·la diverses edicions d'un mateix sistema operatiu (per defecte tindran totes el mateix nom al menú).

Per aconseguir això, cal esbrinar l'identificador de l'objecte amb el qual volem treballar, i per a això cal executar aquest altre ordre:

```
bcdedit /set {GUID} description "La meva nova descripció del SO"
```

(On {GUID} és l'identificador obtingut en el pas anterior, seguint amb l'exemple anterior seria {afb0516d-f5bd-11dd-8879-98b0f6447174}).

Canviar l'ordre de la llista de sistemes operatius

Per realitzar aquest pas hem de recórrer a Bcdedit.exe. En primer lloc novament cal esbrinar l'identificador dels objectes que volem posicionar a la llista. Un cop obtinguts, podem executar aquesta ordre:

```
bcdedit /displayorder {802d5e32-0784-11da-bd33-000476eba25f}
{cbd971bf-b7b8-4885-951-fa03044f5d71} {ntldr}
```

Aquesta ordre posicionaria a la llista (en aquest ordre), els sistemes que el seu identificador són {802d5e32-0784-11da-bd33-000476eba25f}, {cbd971bf-b7b8-4885-951-fa03044f5d71} i {ntldr}, que representaria un sistema Windows XP, per exemple.

També podem afegir sistemes al principi o al final. Per exemple, per afegir el sistema amb identificador sigui {802d5e32-0784-11da-bd33-000476eba25f} al final de la llista, n'hi hauria prou amb executar:

```
bcdedit /displayorder {802d5e32-0784-11da-bd33-000476eba25f} /addlast
```

I per col·locar al principi:

```
bcdedit /displayorder {802d5e32-0784-11da-bd33-000476eba25f} /addfirst
```

Aquest article ha explicat com realitzar les tasques no reparatives més comuns amb l'arrencada dels sistemes operatius Vista, 7, 8, i 10, fent servir l'eina gràfica Msconfig.exe, i la de línia d'ordres Bcdedit.exe, ambdues incloses amb el sistema operatiu.

Certament hi ha multitud d'utilitats de terceres persones que editen de manera gràfica el magatzem BCD de l'equip, potser la més popular sigui EasyBCD (http://neosmart.net/dl.php?id=1). En qualsevol cas, utilitzeu l'eina que esteu utilitzant és altament important que hagi fet una còpia de seguretat de l'arrencada si res anés malament.

Com es pot configurar Grub?

Ara explicarem la configuració del GRUB2, per ser un dels gestors d'arrencada més utilitzats en els sistemes operatius lliures, i és el gestor d'arrencada per defecte en Linkat, Ubuntu i Debian.

Per saber la versió del grub que tenim es pot executar la següent ordre:

```
user@host:~$ grub-install -V
stagrub-inll (GRUB) 1.99-21ubuntu3.10
```

Nota: Qualsevol versió per sobre de la 1.98 és considerada GRUB2.

Estructura de fitxers del GRUB2.

Tota la configuració del GRUB2 està situada al directori /etc/grub.d/ on estan situats una sèrie d'scripts que permeten la configuració automàtica del gestor d'arrancada. El directori /boot/grub/ conté el fitxer de configuració grub.cfg que generen aquests scripts i els mòduls del gestor d'arrancada. El fitxer de configuració mai el tocarem directament, sinò que la personalització la realitzarem a partir del fitxer /etc/default/grub

Podem mostrar aquest fitxer de configuració:

```
user@host:~$ cat /etc/default/grub
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
#GRUB_HIDDEN_TIMEOUT=0
```

```
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
GRUB_CMDLINE_LINUX=""

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that
obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD
...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal (grub-pc only)
#GRUB_TERMINAL=console

# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports
via VBE
# you can see them in real GRUB with the command `vbeinfo'
#GRUB_GFXMODE=640x480

# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter
to Linux
#GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

# Uncomment to get a beep at grub start
#GRUB_INIT_TUNE="480 440 1"
```

Qualsevol línia que comenci per # és un comentari i no serà interpretada.

Tal i com diu al fitxer, la documentació de les opcions està disponible mitjançant una ordre de la consola:

```
user@host:~$ info -f grub -n 'Simple configuration'
```

Exemple de configuració, el temps d'espera

Com a exemple de configuració canviarem el temps d'espera del menú de selecció del sistema operatiu al arrancar el sistema. Per defecte són 10 segons, i ho augmentarem a 20 segons. Només cal editar el fitxer (com a administradors del sistema):

user@host:~$ sudo gedit /etc/default/grub

i modificar una línia:

```
...
GRUB_TIMEOUT=20
...
```

Ara, per aplicar la nova configuració s'executa la següent ordre (sempre que siguem administradors del sistema, com a root directament o amb sudo):

user@host:~$ sudo update-grub

A partir d'ara, el nou temps d'espera serà de 20 segons.

M01 Sistemes Informàtics

CFGS Desenvolupament d’Aplicacions Multiplataforma
UF1. Instal·lació, configuració i explotació del sistema informàtic.

NF1.3

Anàlisi i instal·lació de sistemes informàtics

Sistemes operatius

Sistemes operatius

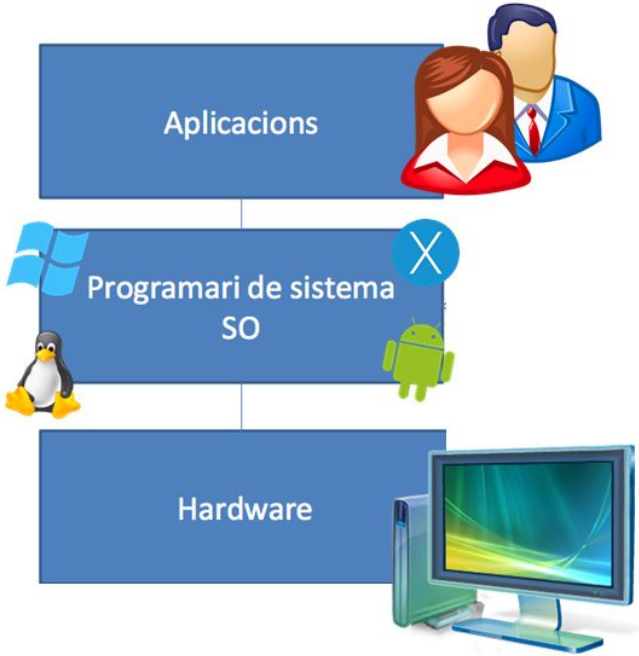
Els sistemes operatius són un conjunt de programes que creen la interfície del maquinari amb l'usuari i que tenen dues funcions primordials:

- Gestionar el maquinari. Administrar d'una manera més eficient els recursos de la màquina.
- Facilitar la feina a l'usuari. Permet una comunicació amb els dispositius de la màquina.

Sense programari, el computador és pràcticament un conjunt d'objectes units entre sí sense cap utilitat.

El programari permet a l'usuari emmagatzemar informació, processar-la i recuperar-la, explotant així tot el potencial que el maquinari per si sol no pot explotar.

Sistemes operatius



Gestiona les diferents parts del computador, i presenta una interfície d'ús agradable a l'usuari.

Per tant podem definir els Sistemes Operatius com un programa o conjunt de programes que gestionen i optimitzen el funcionament del maquinari i del programari, actuant com a intermediari entre l’usuari i els components de la màquina.

Sistemes operatius. Evolució històrica

Evolució Històrica (més info)

L'evolució dels sistemes operatius ha anat paral·lela a la dels ordinadors.

Les causes principals de l'evolució dels sistemes operatius han estat les següents:

- Les actualitzacions i els nous tipus de maquinari.
- La necessitat de resoldre diferents tipus d'errors.
- Les demandes de nous serveis.

Time Line Sistemes Operatius

La primera generació (1945-1955)

Els ordinadors d'aquesta època eren de grans dimensions, difícils d'utilitzar, i a més, presentaven una manca de recursos i tenien uns preus molt alts. **ENIAC i UNIVAC** Aquestes màquines eren gestionades per l'usuari des d'un tauler d'endolls, on no hi havia sistema operatiu.

L'usuari interaccionava directament amb l'ordinador i l'únic llenguatge de programació possible era el llenguatge de màquina (ensamblador).

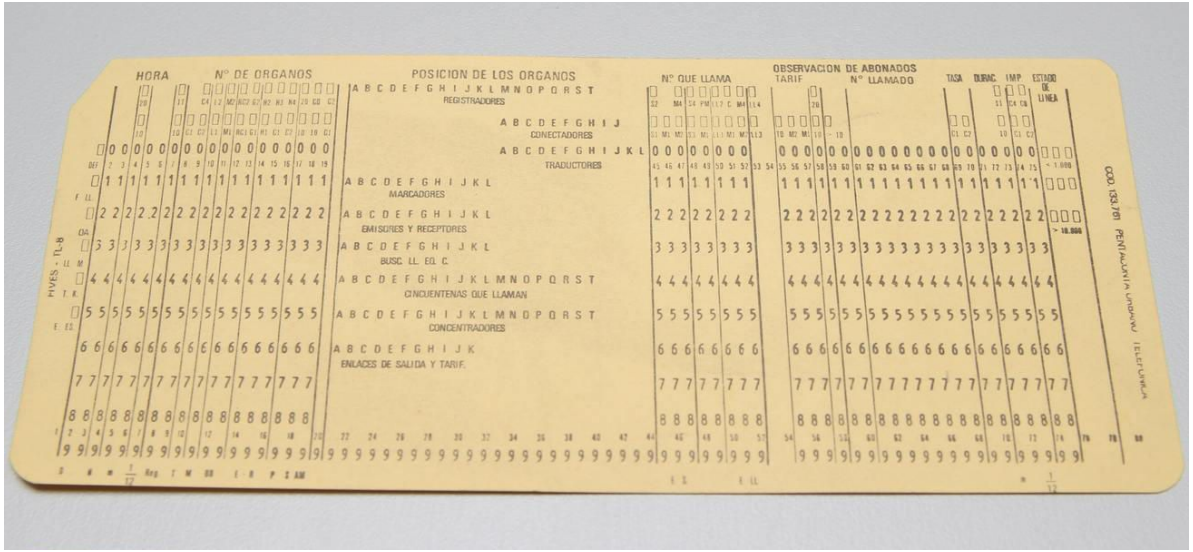
```
-u 100 1a
0CFD:0100 BA0B01        MOV     DX,010B
0CFD:0103 B409          MOV     AH,09
0CFD:0105 CD21          INT     21
0CFD:0107 B400          MOV     AH,00
0CFD:0109 CD21          INT     21
-d 10b 13f
0CFD:0100                                   48 6F 6C 61 2C          Hola,
0CFD:0110  20 65 73 74 65 20 65 73-20 75 6E 20 70 72 6F 67  este es un prog
0CFD:0120  72 61 6D 61 20 68 65 63-68 6F 20 65 6E 20 61 73  rama hecho en as
0CFD:0130  73 65 6D 62 6C 65 72 20-70 61 72 61 20 6C 61 20  sembler para la
0CFD:0140  57 69 6B 69 70 65 64 69-61 24                    Wikipedia$
```

La primera generació (1945-1955)

Característiques:

- En aquestes màquines, només era possible la **monoprogramació**, és a dir, l'execució d'un sol programa, que s'introduia generalment per mitjà de màquines lectores de fitxes perforades.
- Gran volum físic i elevat preu.
- S'introdueixen els programes directament a través d'endolls o interruptors.
- No existeixen els llenguatges de programació. Directament en codi màquina.
- Al final de l'etapa s'introdueixen les targetes perforades.
- Principal funció: càlculs matemàtics (sinus, cosinus...)

La primera generació (1945-1955)



La segona generació (1955-1965)

La invenció del transistor va canviar radicalment el món dels ordinadors, van reduir la grandària i la fiabilitat.

Es tractava de màquines que només eren a l'abast de grans corporacions (institucions com l'exèrcit, les universitats i el govern).

Apareixen dispositius d'E/S nous com les impressores.

S'inventen dispositius d'emmagatzematge nous: les unitats de cintes magnètiques.

La segona generació (1955-1965)

Es desenvolupen llenguatges de programació assembladors, carregadors, muntadors, biblioteques de diferents tipus i rutines estàndard d'E/S dels diferents dispositius que es podien utilitzar en els programes.

La manera d'operar dels sistemes continuava essent poc eficaç a causa de les diverses operacions manuals realitzades per l'operador, expert de la màquina al servei dels programadors.

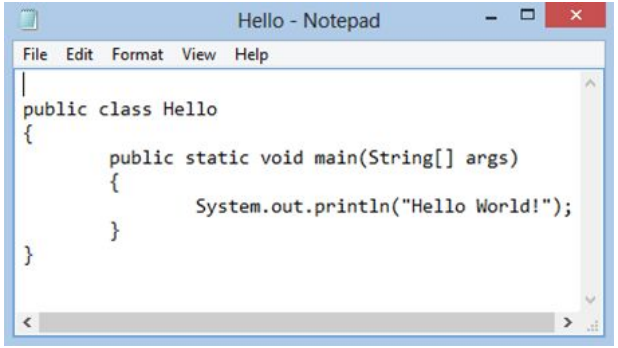
Per això s'intentava aprofitar el temps perdut i augmentar la utilització dels recursos del sistema i apareixen el conceptes de programació per lots, memòria intermèdia (buffering) i gestió de cues (spooling o spool).

Property of Museum of History & Industry, Seattle

Tercera generació (1965-1980)

La tercera generació es fonamenta tecnològicament en els circuits integrats (xips). Els ordinadors d'aquesta època són molt més petits i ràpids.

- Perifèrics més ràpids i efectius.
- Terminals remots per accedir a bancs de dades.
- Programari de base adaptat a l'usuari, la qual cosa permet l' ús de llenguatges universals pensats per ser utilitzats en diferents tipus d'ordinadors (és el que s'anomena portabilitat).



Tercera generació (1965-1980)

- Estandardització dels llenguatges d’alt nivell i aparició d’altres de nous, com el Basic i el Pascal.
- Aparició de la multiprogramació com una de les tècniques més innovadores.

Quarta generació (1980-?)

Apareixen els microprocessadors que és un gran avenç de la microelectrònica , són circuits integrats d'alta densitat i amb una velocitat impressionant . Els microordinadors amb base en aquests circuits són extremadament petits i barats , pel que el seu ús s'estén al mercat industrial . Aquí neixen els ordinadors personals que han adquirit proporcions enormes i que han influït en la societat en general sobre l'anomenada "revolució informàtica"

SO en xarxa i distribuït: fan servir recursos de màquines remotes.

SO en temps real: processar dades (sensors) en temps limitat.

SO encastats (embedded): computadors que controlen aparells.

SO computador butxaca: PDA, mòbils, ...

Quarta generació (1980-?)

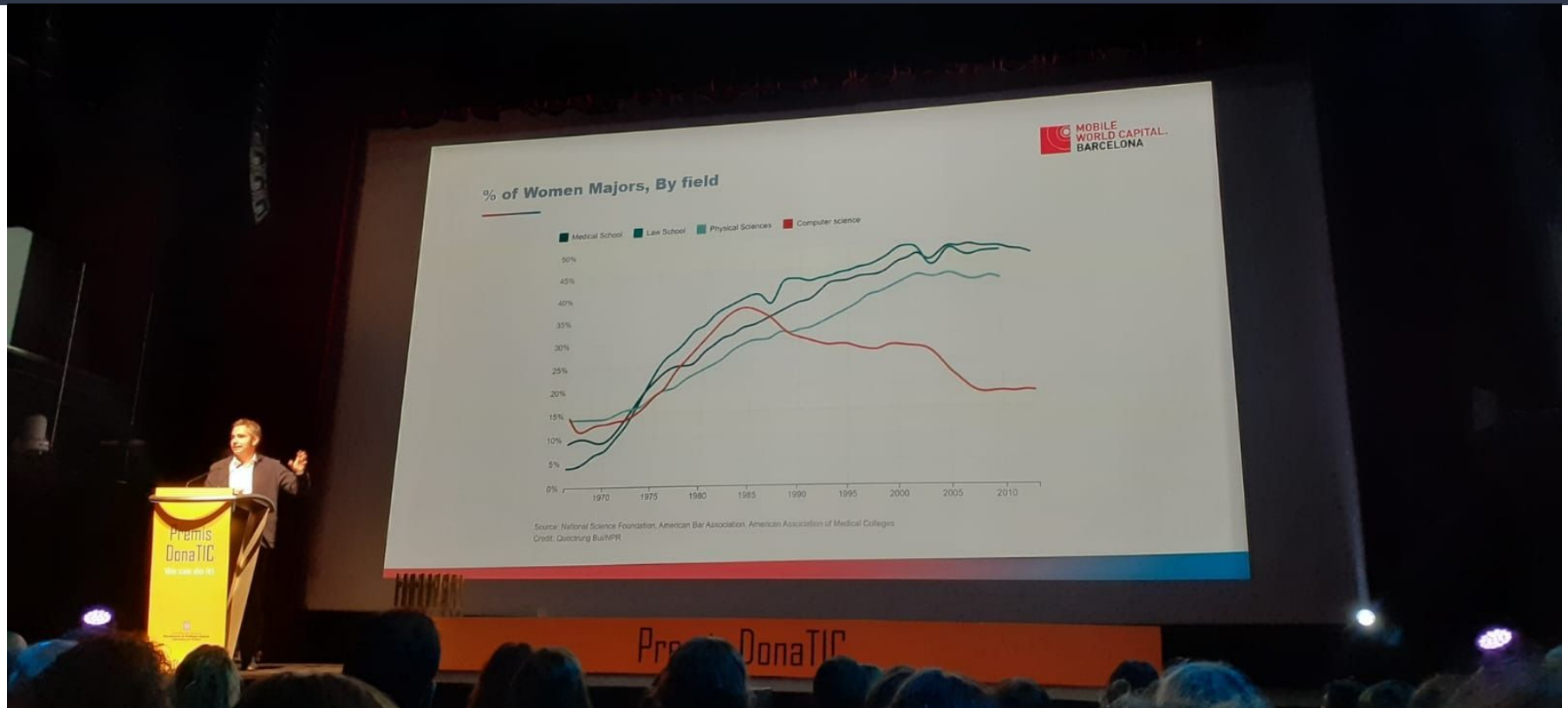
Característiques

- S'introdueix el microprocessador.
- Diferents xips amb diferents funcions.
- Preus més econòmics.
- Apareixen els ordinadors personals(PC)

S’introdueixen els següents conceptes:

- Sistema operatiu en temps real, referit als sistemes que tenen un temps de resposta curt davant de qualsevol petició.
- Una xarxa d’ordinadors és un conjunt d’ordinadors interconnectats podem parlar de LAN, MAN i WAN.
- Un sistema distribuït és un entorn en què diversos ordinadors o nusos es distribueixen les tasques del sistema operatiu.
- El multiprocés és una tècnica en què podem executar més d’una tasca a la vegada (al mateix moment), ja que en un mateix ordinador hi ha més d’una CPU.

Quarta generació (1980-?)



Quins Sistemes operatius coneixem?

Els més utilitzats actualment són els següents

- Ubuntu
- Android
- Mac OS X
- iOS
- Windows

Windows1 1985 | Windows 3.1 1992 | Windows 95 1995 | Windows XP 2001 | Windows Vista 2006 | Windows 7 2009 | Windows 8 2012 | Windows 10 2015

10.0: "Cheetah"
March 24, 2001
$129

10.1: "Puma"
September 25, 2001
Free update

10.2: "Jaguar"
August 23, 2002
$129

10.3: "Panther"
October 24, 2003
$129

10.4: "Tiger"
April 29, 2005
$129

10.5: "Leopard"
October 26, 2007
$129

10.6: "Snow Leopard"
August 28, 2009
$29 (upgrade)

10.7: "Lion"
July 20, 2011
$29 (upgrade)

10.8: "Mountain Lion"
Developer beta February 16, 2012
Release "this summer"

cnet

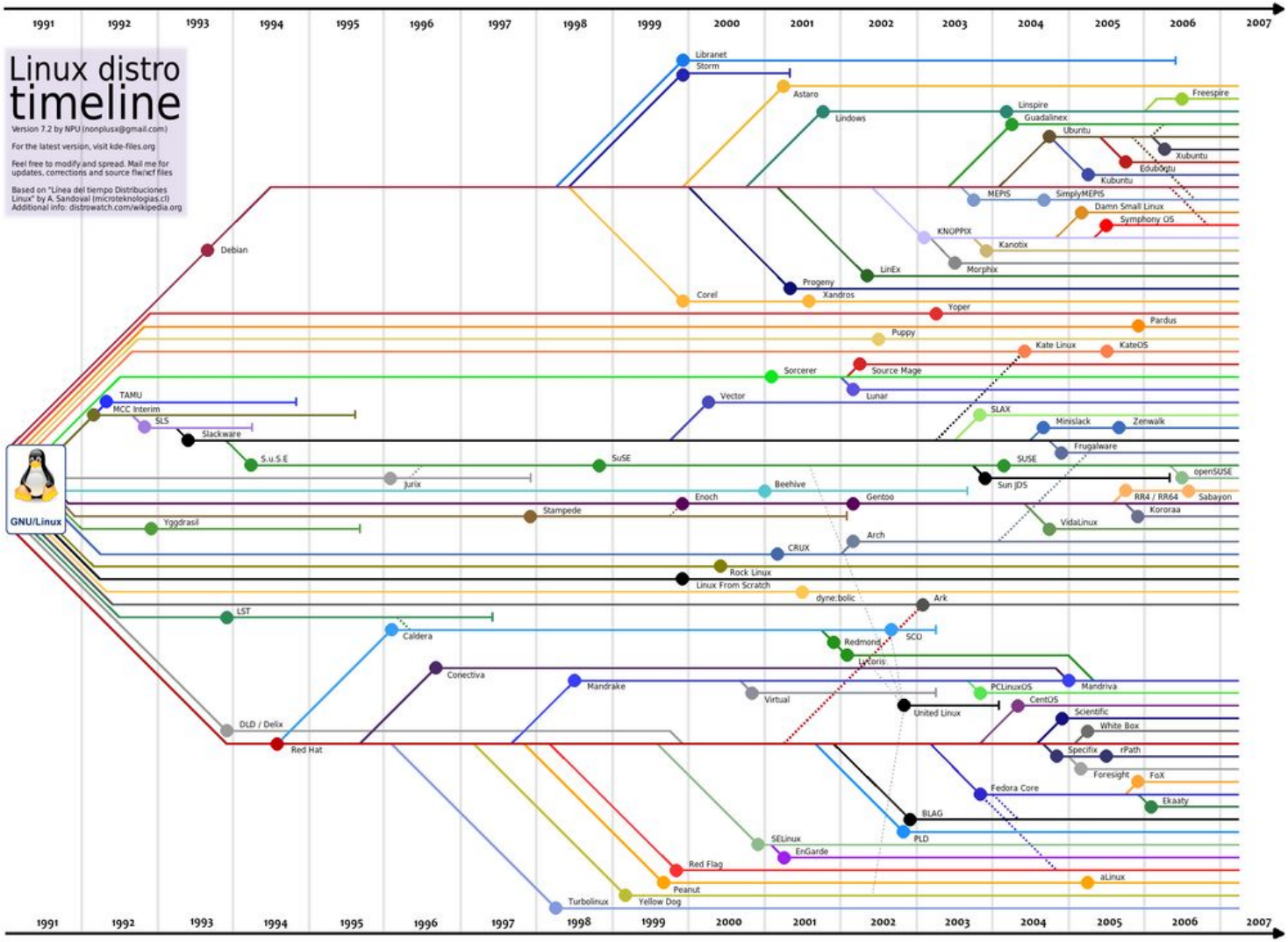
Linux distro timeline

Version 7.2 by NPU (nonplusx@gmail.com)

For the latest version, visit kde-files.org

Feel free to modify and spread. Mail me for updates, corrections and source file/xcf files

Based on "Línea del tiempo Distribuciones Linux" by A. Sandoval (microteknologias.cl)
Additional info: distrowatch.com/wikipedia.org



Android 1.5 Cupcake
Android 1.6 Donut
Android 2.0 Eclair
Android 2.2 Froyo
Android 2.3 Gingerbread
Android 3.0 Honeycomb
Android 4.0 Ice Cream Sandwich
Android 4.1 Jelly Bean
Android 4.4 KitKat
Android 5.0 Lollipop
Android 6.0 Marshmallow
Android 7.0 Nougat

Funcions del SO. Gestió de processos

Un procés és bàsicament un entorn format per tots els recursos necessaris per a poder executar programes. Des del punt de vista del SO un procés és un objecte més que s'ha de gestionar i al qual s'ha de donar servei.

El concepte de procés engloba tots els recursos necessaris per a executar un programa. En aquest apartat englobem els serveis que fan referència a la creació i eliminació de processos. Però també hi podem trobar altres serveis, com la suspensió i la reanimació de processos o l’assignació de prioritats.

Funcions del SO. Gestió dispositius E/S

Els serveis relacionats amb el sistema d'entrada/sortida tenen com a funcions principals crear, obrir i tancar canals d'entrada/sortida, i també escriure i llegir-los

Funcions del SO. Gestió dels recursos del sistema

Aquest grup de serveis és molt específic de cada sistema operatiu, però un exemple comú és la gestió de la memòria. Algunes de les operacions que s'efectuen amb la memòria del sistema són assignar, alliberar, protegir, compartir, etc.

Funcions del SO. Gestió del sistema d’arxius

Juntament amb el bloc de serveis d'entrada/ sortida, també és un grup de serveis molt utilitzat. La gestió del sistema de fitxers inclou les funcionalitats com crear i eliminar fitxers i directoris, canviar de directori, etc.

Com sol·licitar els serveis del SO

Com sol·licitar els serveis al SO

1. Les crides al sistema permeten fer peticions directes al sistema operatiu. Les crides al sistema són la visió que té un programador dels serveis que pot oferir el sistema operatiu. Els llenguatges d'alt nivell en última instància fan una crida al sistema.
2. A un nivell més pròxim a l'usuari, el programari de sistema ofereix un programa, l'anomenat intèrpret d'ordres (shell), gràcies al qual l'usuari pot dialogar amb el sistema operatiu sense necessitat d'escriure cap programa.

Tipus de sistemes operatius

Nombre d’usuaris: en funció dels usuaris que suporti a temps real.

- Monousuari
- Multiusuari

Nombre de tasques: en funció de la quantitat de tasques que es puguin estar desenvolupant en un mateix moment.

- Monotasca
- Multitasca

Nombre de processadors: en funció de la quantitat de processos que es puguin desenvolupar en un mateix moment.

- Monoprocés
- Multiprocés

Més informació

- Resum de l’evolució dels sistemes operatius d’escriptori
- Mapa gràfic Sistemes Operatius
- Time Line Sistemes Operatius
- Documental «La revolución de SO»

Llenguatge assemblador

Simulador

Aquest simulador d’assemblador de 8 bits en JavaScript és una eina educativa per aprendre a practicar les bases del llenguatge màquina. Cada instrucció d’assemblador es tradueix en un codi executable directament pel processador de la simulació.

D’aquesta manera ens podem fer una idea de com funciona, de manera molt simplificada, el llenguatge màquina.

El simulador el va escriure Marco Schweighauser en JavaScript+Angular. A l’esquerra escribim el codi en assemblador i si apretem el botó Ensamblar cada instrucció es converteix en instruccions d’un, dos o tres bytes a la RAM.

Les possibilitats del simulador són limitades, només hi ha 256 bytes de RAM, per tant, el programa i la sortida hi han de poder caber. Hi ha quatre registres (A, B, C, D) i tres flags (cero = Z, acarreo = C i fallo = F), dos punters (IP = instrucciones, pila = SP).

Simulador

SIMULADOR

INSTRUCCIONS

ASCII

Simulador

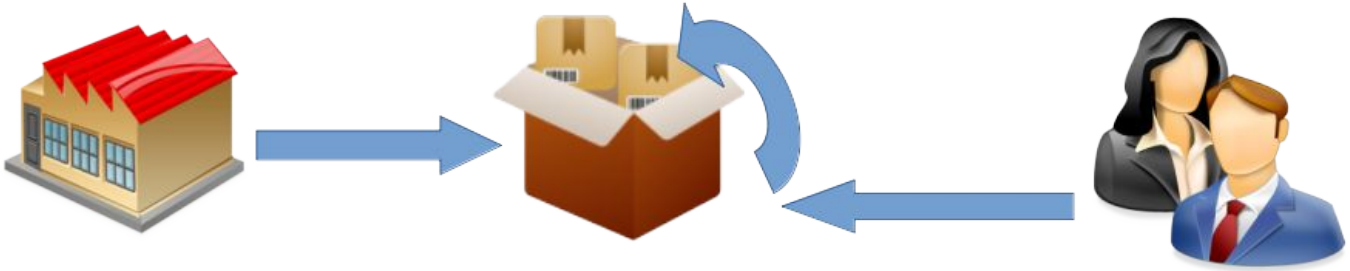
Un altre simulador AQUÍ

Llicències

Llicències i tipus de llicències

Una llicència és un contracte mitjançant el qual una persona rep d'una altra el dret d'ús de diversos dels seus béns, normalment de caràcter no tangible o intel·lectual.

Pot donar-se a canvi del pagament d'una suma determinada per l'ús dels béns. Aquests actius són propietat de l'atorgant, i poden ser béns de propietat intel·lectual com una marca comercial, patents o tecnologies. També poden ser objecte de llicència d'altres béns de caràcter intangible, com la distribució d'obres intel·lectuals.



Llicències i tipus de llicències

Una **llicència de software** és un contracte entre el llicenciant (autor o titular dels drets d'explotació o distribució) i el llicenciatari del programa informàtic, per tal de fer servir el software complint una sèrie de termes i condicions establertes dins de les seves clàusules.

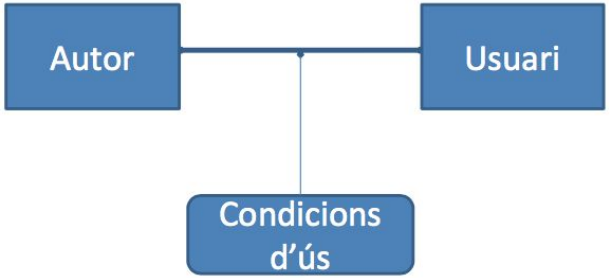
Les llicències de software poden establir entre d'altres coses, la **cessió de determinats drets** del propietari a l'usuari final sobre una o diferents còpies del programa informàtic, els **límits en la responsabilitat per errades**, el **termini de cessió de drets**, **l'àmbit geogràfic de validesa** del contracte o fins i tot poden establir determinats **compromisos de l'usuari final cap al propietari**, com poden ser la **no cessió del programa a tercers** o la **no reinstal·lació del programa** en equips als que no es va instal·lar originàriament.

Hi ha un gran munt de llicències i diferents tipus dels quals caldria destacar: les llicències fonamentals de Microsoft. Les llicències Creative commons i les basades en GNU.

Les llicències d’ús dels Sistemes Operatius

Característiques:

- Cessió de drets
- Termini cessió drets
- Àmbit geogràfic de validesa del contracte
- Compromisos de l’usuari final



Llicències de programari propietari.

Són aquelles en les que els usuaris tenen limitades les possibilitats de fer-lo servir, modificar-lo o redistribuir-lo (amb o sense modificacions). Les més conegudes en aquest sentit són les llicències de Microsoft.

Les llicències a Microsoft poden ser de diferent tipus. Cal destacar només les més habituals.

Software preinstal·lat o en servidor nou (OEM- Original Equipment Manufacturer): Es pot comprar un PC o servidor amb un software preinstal·lat pel fabricant o pel distribuïdor. Aquesta és una de les formes més senzilles i econòmiques d'obtenir el software. Les llicències OEM disposen de les següents característiques:

- Formen part d'un programari instal·lat en un determinat maquinari (tot i que en ocasions pot anar en un cd juntament amb el maquinari sense haver-se arribat a instal·lar). Per tant, quan el maquinari deixa de funcionar, la llicència del programari finalitza.
- Normalment no permet actualitzacions ni desactualitzacions del programari.
- Normalment venen representades a l'equip mitjançant una enganxina (sticker) anomenada COA (Certificate of Autenticity).

Llicències de programari propietari.

Llicències individuals (Retail): Aquest tipus de llicències donen dret a fer una instal·lació en qualsevol ordinador, però no de forma simultània. Aquestes llicències agrupen les dades i la documentació. Aquest tipus és especialment convenient per usuaris finals o petites empreses que tinguin entre 1 y 5 ordinadors com a màxim, però és una mica més car que una llicència OEM.

Llicències de programari propietari.

Llicències per volum: Els acords de llicència per volum de Microsoft són la forma més senzilla de comprar llicències per organitzacions que tenen des de 2 fins milers d'ordinadors. Les opcions de llicències per volum satisfan les diverses necessitats de les empreses, institucions acadèmiques i altres usuaris que facin servir un gran volum de llicències. A més, a nivell individual, cada programari associat a la llicència per volum surt més barat que agafant llicències OEM o Retail.

Es pot triar entre dues modalitats de llicència per volum:

Llicències per dispositiu: Dóna el dret a fer ús d'aquest programari a qualsevol usuari que faci ús del dispositiu que el conté. D'aquesta forma, cada dispositiu rep una llicència diferent, però tot usuari pot fer-ne ús d'aquest (sempre i quan treballin amb el dispositiu que associat a la llicència)

Llicències de programari propietari.

Llicències per usuari: Dóna el dret a fer ús d'aquest programari en qualsevol dispositiu, sempre i quan l'usuari que faci servir el programari sigui el contemplat a la llicència. D'aquesta forma, es llicencia cada usuari per separat, però el programa en qüestió es pot instal·lar a qualsevol dispositiu (sempre i quan només els usuaris associats a la llicència per volum treballin amb aquests dispositius).

Llicències de programari lliure

Les llicències de programari lliure es basen en la distribució del codi font amb el programa. "Programari Lliure" (Free software) és una qüestió de llibertat, no de preu. Per a comprendre el concepte, heu de pensar en "free" com a lliure i no com a gratuït. Els usuaris de programari lliure tenen:

- La llibertat de fer servir el programa amb qualsevol propòsit.
- La llibertat d’estudiar el funcionament del programa i adaptar-lo a les necessitats.
- La llibertat de distribuir còpies.
- La llibertat de millorar el programa i fer públiques les millores.

Un programa és programari lliure si els usuaris tenen totes aquestes llibertats. Hem de tenir clar que el fet de què un programa sigui lliure no implica en cap moment que hagi de ser també gratuït.

Llicències de programari lliure

Normalment s’utilitza la Llicència Pública General de GNU (GNU GPL), aquesta és un tipus de llicència per a programari que permet la còpia, distribució (comercial o no) i modificació del codi, sempre que qualsevol modificació es continuï distribuint amb la mateixa llicència GPL. La llicència GPL no permet la distribució de programes executables sense el codi font corresponent o una oferta de com obtenir-lo gratuïtament.

La documentació pel programari lliure ha de ser documentació lliure, de manera que tothom pugui distribuir-la i millorar-la, igual que el programa que descriu. Per a què sigui lliure cal difondre-la sota els termes d'una llicència de documentació lliure. Normalment s’utilitza la Llicència de Documentació Lliure de GNU (GNU FDL).

La Llicència Pública General (GPL, de l'anglès General Public License) Aquesta llicència va ser dissenyada originalment per Richard Stallman i el grup GNU, principalment Eben Moglen, com a alternativa al model de programari propietari predominant. Actualment, Linux és el programa sota llicència GPL amb més difusió.

Llicències de codi obert (Open Source)

El codi obert és un model de desenvolupament de programari basat en la col·laboració oberta. És similar al concepte de programari lliure, però més enfocat als beneficis pràctics (accés al codi font) que no en qüestions ètiques o de llibertat. Les llicències de codi obert són molt similars a les de codi lliure, entre les més usades estan:

Versió 2 de GPL (GNU GPLv2): La versió 3 de GPL no agrada en entorns OpenSource, ja que segons diuen, s'apropa en excés als ideals de llibertat del programari, deixant de banda d'altres drets més propers al desenvolupador del programari en qüestió.

MIT: Aquesta llicència es va fer a l'institut Tecnològic de Massachusetts. És una licencia molt permisiva que imposa molt poques limitacions a la seva reutilització. Permet reutilitzar programari dintre de programari propietari. Però a la seva vegada és compatible amb GPL.

Les llicències Creative Commons

Creative commons (CC) és una organització no governamental sense ànim de lucre que desenvolupa plans per tal d'ajudar a **reduir les barreres legals de la creativitat**, fent servir una nova legislació i les noves tecnologies. Va ser fundada per Lawrence lessig, professor de dret en la universitat de Stanford i especialista en ciberdret. Creative commons és també el nom donat a les llicències desenvolupades per aquesta organització.

"La filosofia del Creative Commons s'allunya del "Todos los derechos reservados" per derivar a un lema més semblant a "Algunos derechos reservados"

Més informació aquí.

Les llicències Creative Commons

Posar unes obres sota aquest tipus de llicències no significa que no tinguin copyright. Aquest tipus de llicències ofereixen alguns drets a terceres persones sota certes condicions. Hi ha un total de 4 paràmetres Creative commons per combinar entre sí i, així, generar un tipus de llicència Creative commons:

1. Reconeixement: En qualsevol explotació d'una obra autoritzada per la llicència cal reconèixer l'autor.
2. No comercial: L'explotació de l'obra queda limitada a usos no comercials.
3. Sense obres derivades: L'autorització per explotar una obra no inclou la transformació per crear una altra derivada.
4. Compartir igualment: L'explotació inclou la creació d'obres derivades sempre que mantinguin la mateixa llicència al ser divulgades.

Les llicències Creative Commons

Es poden fer aquestes 6 combinacions

[Crea aquí la teva llicència]

Reconocimiento (by): Se permite cualquier explotación de la obra, incluyendo una finalidad comercial, así como la creación de obras derivadas, la distribución de las cuales también está permitida sin ninguna restricción.

Reconocimiento – NoComercial (by-nc): Se permite la generación de obras derivadas siempre que no se haga un uso comercial. Tampoco se puede utilizar la obra original con finalidades comerciales.

Reconocimiento – NoComercial – CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Reconocimiento – NoComercial – SinObraDerivada (by-nc-nd): No se permite un uso comercial de la obra original ni la generación de obras derivadas.

Reconocimiento – CompartirIgual (by-sa): Se permite el uso comercial de la obra y de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Reconocimiento – SinObraDerivada (by-nd): Se permite el uso comercial de la obra pero no la generación de obras derivadas.

Formes de distribució de programari

La forma de distribució del programari esta freqüentment relacionada amb determinades modalitats de llicència. En aquest sentit, podem parlar de les següents formes de distribució:

Freeware: És un tipus de programari que permet la seva distribució de forma gratuïta però amb algunes restriccions: No modificar l'aplicació, no vendre-la, ni apropiar-se de la seva autoria. En ocasions, pot desautoritzar l'ús del programari amb finalitats comercials (o obligar al pagament en cas de treure profit econòmic) o governamentals.

Shareware: És un tipus de programari que permet la seva distribució de forma gratuïta però aquest pot disposar d'alguna limitació al seu ús fins que es fa efectiva la seva compra.

Adware: És un tipus de programari que permet la seva distribució de forma gratuïta, però que, a canvi, mostra publicitat al llarg del seu ús a canvi d'aquesta característica. Normalment el propi programari dona la opció d'adquirir una versió sense publicitat abonant una determinada quantitat de diners. Una aplicació adware pot ser considerada també shareware.

Domini públic: És un tipus de programari que no disposa de cap tipus de llicència. Això significa que qualsevol pot obtenir el seu codi font, modificar-ho, distribuir-ho i inclús publicar modificacions sota una determinada llicència.

M01 Sistemes Informàtics

CFGS Desenvolupament d’Aplicacions Multiplataforma
UF1. Instal·lació, configuració i explotació del sistema informàtic.

NF1.4

Anàlisi i instal·lació de sistemes informàtics

Xarxes

Xarxes

Xarxa informàtica

Una xarxa informàtica és un grup interconnectat de computadores i components diversos. La finalitat principal per a la creació d'una xarxa és **compartir els recursos** i la informació en la distància, assegurar la **fiabilitat** i la **disponibilitat** de la informació, augmentar la velocitat de transmissió de les dades i reduir el cost general d'aquestes accions.

Xarxes

Xarxa d'ordinadors: És un conjunt d'equips, també anomenats Hosts (ordinadors i o altres dispositius finals) connectats mitjançant un medi de cable, ones electromagnètiques o qualsevol altre mètode de transport de dades, amb la finalitat de compartir informació (arxius), recursos (impressores...) i serveis (jocs, Internet).

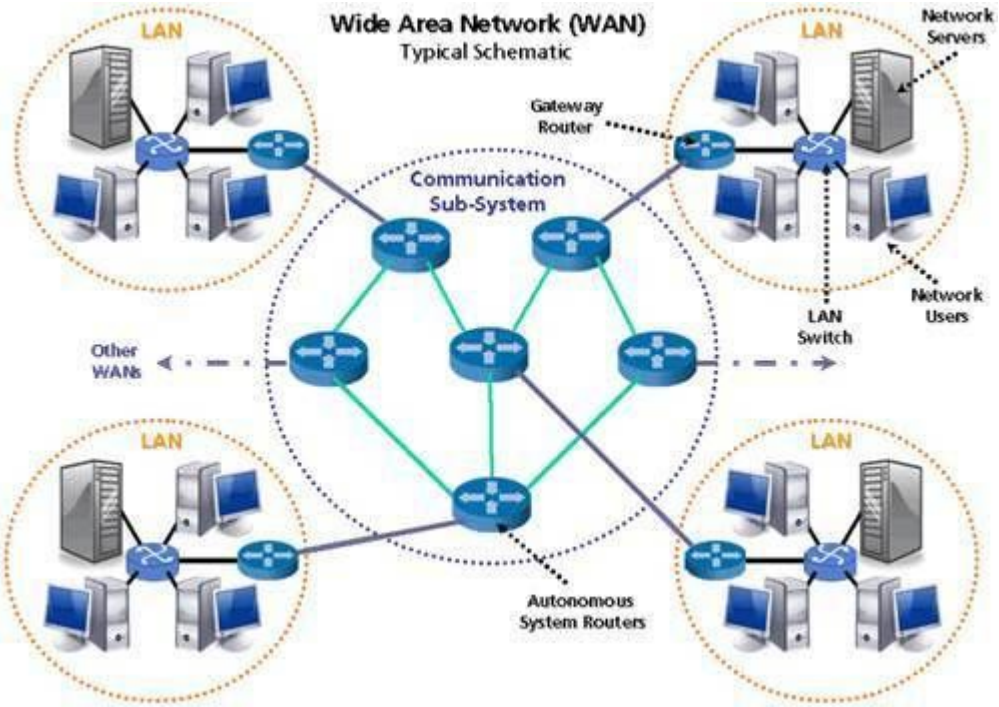
Les xarxes es poden classificar en funció de 3 característiques:

- Segons la seva tecnologia de transmissió.
- Segons la seva topologia (forma en la que s'enllacen els seus nodes).
- Segons la seva escala.

Xarxes. Tipus

Tipus de xarxes:

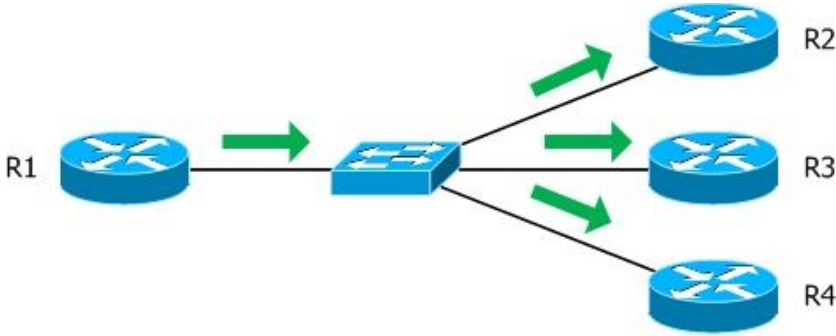
- Segons tecnologia de transmissió
  a. xarxes broadcast
  b. xarxes punt a punt
- Segons la seva topologia
  a. Bus
  b. Estrella
  c. Anell
  d. Arbre
- Segons la seva grandària
  a. LAN: Xarxes d'àrea local
  b. MAN: Xarxes d'àrea metropo
  c. WAN: Xarxes d'àrea estatal



Xarxes. Tipus segons tecnologia transmissió

Broadcast

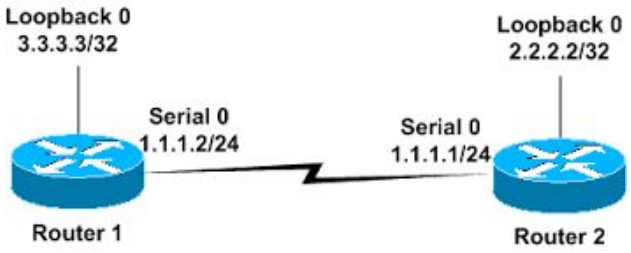
- Tenen un únic canal de comunicació compartit per totes les màquines de la xarxa.
- Els paquets que envia una màquina (ordinador) són rebuts per totes les altres.
- Un camp d'adreça dins del paquet especifica a qui està adreçat.
- La màquina que identifica la seva adreça processa el paquet.



Xarxes. Tipus segons tecnologia transmissió

Punt a punt

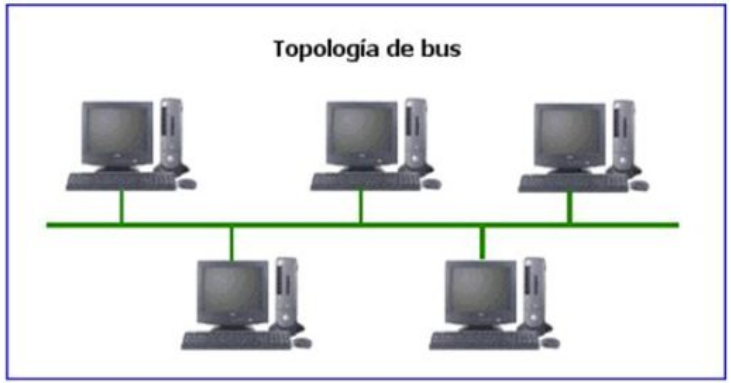
- Només 2 màquines.
- Consisteixen en moltes connexions entre parells individuals de Hosts.
- Cada Host que participa en una xarxa d'enllaços punt a punt és un node de la xarxa.
- Cada node entremig ha de prendre la decisió sobre per on ha de dirigir els paquets que rep.



Xarxes. Topologia Bus

Topologia en bus

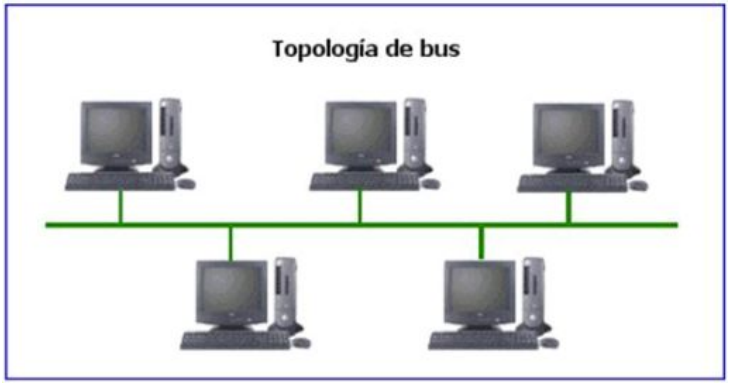
És la topologia de xarxa més simple. Consisteix en un cable al qual es connecten tots els Hosts (nodes, llocs) de la xarxa. A cada Host se li assigna una adreça. Aquesta adreça serveix per identificar l'origen i destí de la informació.



Xarxes. Topologia Bus

Topologia en bus

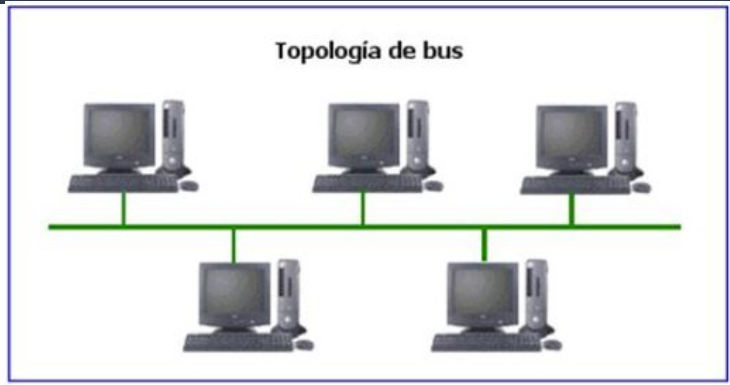
- Quan un Host transmet informació, aquesta viatja en ambdós sentits fins que arriba al destinatari.
- Tots els Hosts estan pendents de si hi ha informació viatjant pel cable.
- Quan es transmet informació, tots els Hosts la capturen i comproven si són els destinataris.
- Si és així, se la queden, si no la descarten.
- El destinatari no elimina la informació del bus, simplement la llegeix.



Xarxes. Topologia Bus

Topologia en bus

- És la topologia més simple i model de les primeres LAN.
- Tots els Hosts d'una xarxa poden veure tots els senyals de tots el altres Hosts.
- Es poden produir problemes de trànsit i col·lisions.
- Si el cable que fa de bus es trenca cau tota la xarxa.



Xarxes. Topologia Anell

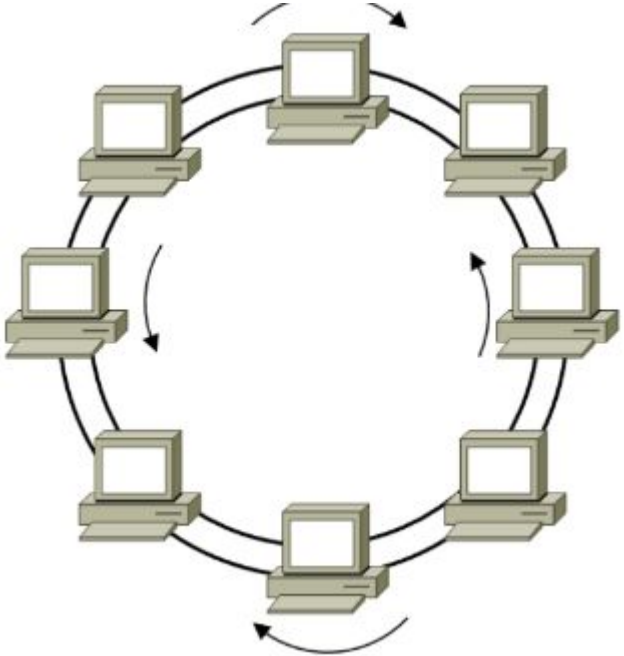
Topologia en anell

- La topologia en anell consisteix a connectar cada ordinador a dos o més, de manera que es formi un anell.
- Quan un ordinador vol enviar una trama a un altre, aquest ha de passar per tots els ordinadors que hi ha entremig: la circulació per l'anell és unidireccional.
- Un node envia informació al següent.
- Si un node rep informació que es per ell se la queda i ja no passa el missatge. En cas contrari passa el missatge al següent node.
- Avui en dia tenen aplicació unint edificis, xarxes de fibra òptica amb gran amplada de banda. (p.e: campus universitari)

Xarxes. Topologia Anell

Topologia en doble anell

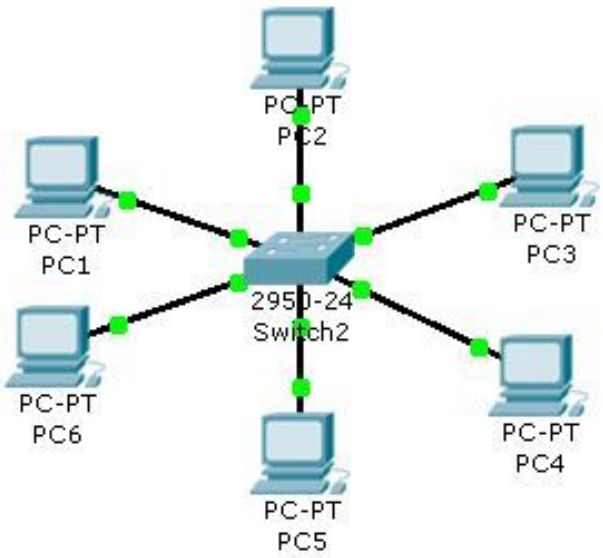
- Consisteix en dos anells concèntrics.
- Cada ordinador de la xarxa està connectat a ambdós anells, però no hi ha connexió entre aquests.
- Aquest tipus de topologia augmenta la confiabilitat i la flexibilitat de la xarxa.



Xarxes. Topologia Estrella

Topologia en estrella

- Tots els ordinadors es connecten a un node central a través de línies de transmissió individuals.
- Aquest node central, que pot ser un concentrador (HUB) o bé un conmutador (Switch), rep els missatges de cada estació i el reenvia al destí que correspongui.
- Cada node (lloc, estació) té el seu propi cable, una errada en aquest cable, només afectarà al trànsit d'aquest node.
- Per altra banda, si falla el node central de la xarxa (que és el que centralitza totes les comunicacions), res funcionarà a la xarxa.



Xarxes. Topologia Estrella

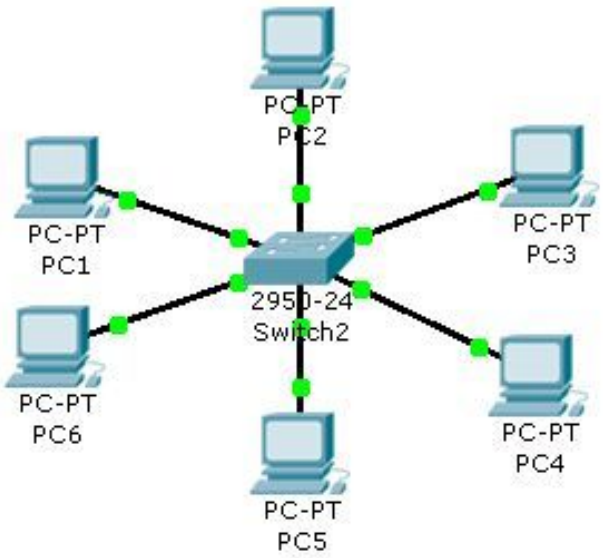
Topologia en estrella

Aspectes positius:

Robustesa davant d’errades.
Molt flexible.
S’adapta a molt bé a connexions sense fils.

Inconvenients:

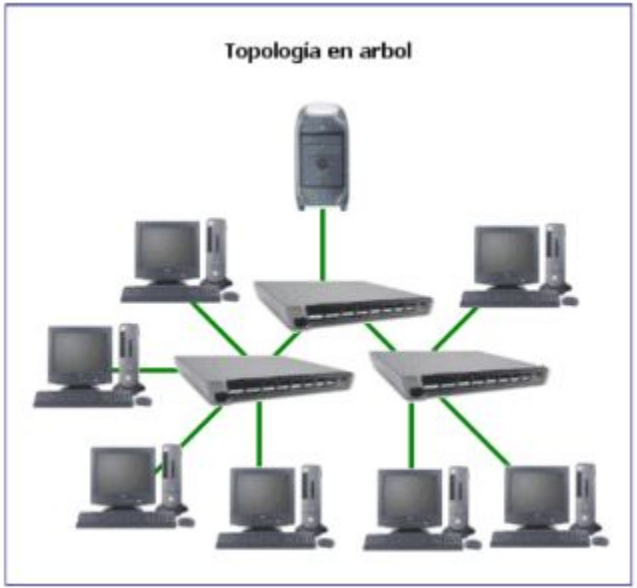
Si falla l'element centralitzador cau tota la xarxa.



Xarxes. Topologia Arbre

Topologia en arbre

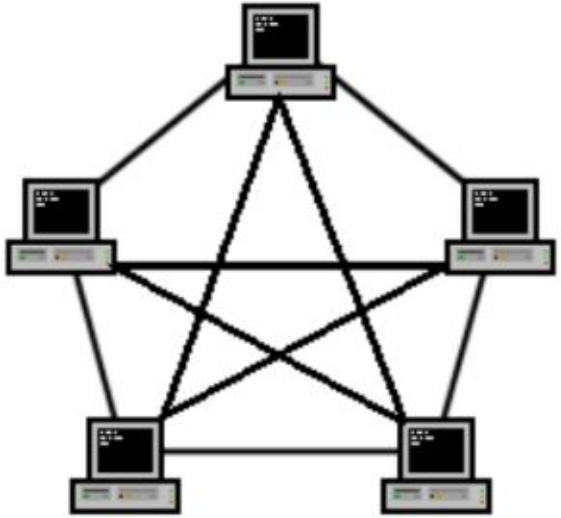
- Similar a la topologia en estrella, llevat que no té un node central.
- Té un node d'enllaç troncal, generalment ocupat per un hub o switch des del que es ramifiquen els altres nodes.
- L'enllaç troncal és un cable amb diverses capes de ramificacions, i el flux de la informació és jeràrquic.
- Connectat en l'altre extrem a l'enllaç troncal generalment es troba un host servidor.



Xarxes. Topologia Malla

Topologia en malla

- Cada node s’enllaça directament amb tots els altres.
- Si algun enllaç deixa de funcionar la informació pot circular a través de qualsevol dels altres enllaços fins arribar a l’ordinador destinatari.
- Permet que la informació circuli per diverses rutes a través de la xarxa.
- Tan sols és recomanable implementar-ho amb una petita quantitat de nodes.

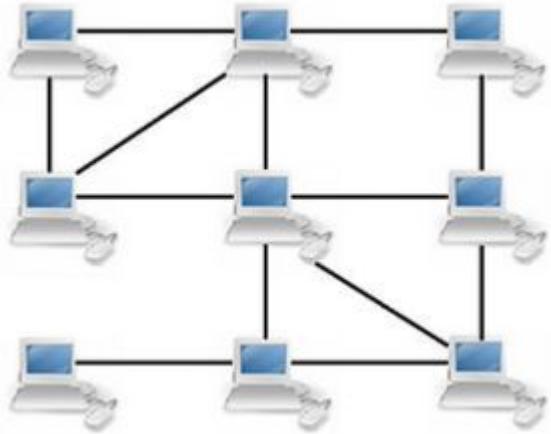


*(Si disposem de N terminals clients el número de cables per interconnectar cada terminal amb un altre és de N * (N-1) /2)

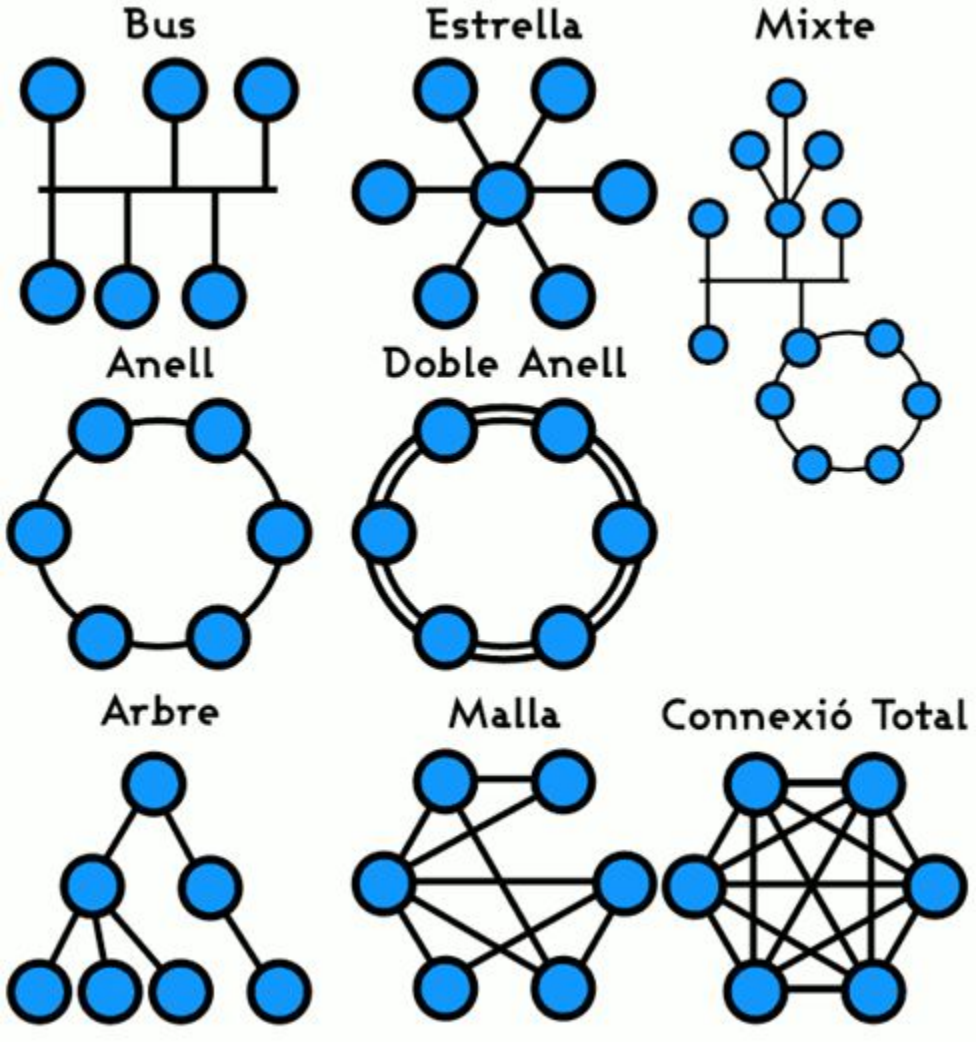
Xarxes. Topologia Irregular

Topologia irregular

No existeix un patró obvi d'enllaços i nodes. Les xarxes que es troben en les primeres etapes de construcció, o es troben mal planificades, molt sovint es connecten d'aquesta manera.



Xarxes. Topologies



Xarxes. Tipus

- Xarxes segons la seva grandària
  - **PAN** (Personal Area Network): Les nostres de casa amb pocs dispositius.
  - **LAN** (Local Area Network): La xarxa de l’aula, oficines…
  - **MAN** (Metropolitan Area Network): Xarxes entre instituts, universitats o altres col·lectius.
  - **WAN** (Wide Area Network): Xarxes de més gran abast entre corporacions internacionals, un exemple podria ser la xarxa eduroam.

Xarxes. Tipus PAN

PAN: Personal Area Network

- Es un tipus de xarxa que permet connectar diferents dispositius personals en un àrea molt petita (pocs metres) i a una velocitat relativament baixa.

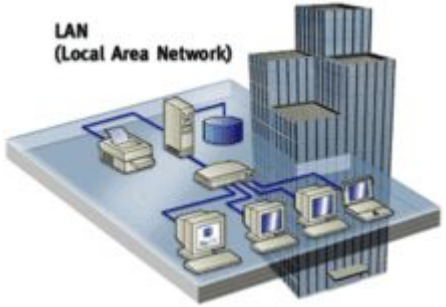
- Normalment els dispositius que s’utilitzen son mòbils, tablets, impressores, escàners ....

Xarxes. Tipus LAN

LAN: Local Area Network.

És un conjunt d'elements físics i lògics que proporcionen interconnexió entre dispositius a una àrea privada. Les seves característiques principals són:

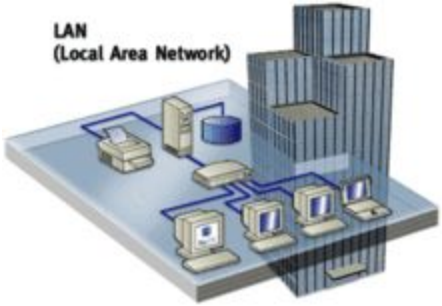
- Restricció geogràfica: àmbit d'una oficina, planta d'un edifici, un edifici sencer i inclús un campus universitari.
- Velocitat de transmissió: relativament elevada (1 Gbps)
- Privacitat: tota la xarxa pertany a la mateixa organització.
- Fiabilitat en les transmissions: la taxa d'error sol ser molt baixa.



Xarxes. Tipus LAN

LAN: Local Area Network.

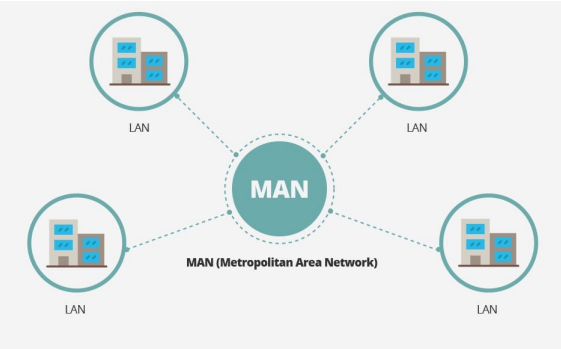
- Les LAN estan orientades a compartir recursos (impressores, documents ...) entre els usuaris de la xarxa.
- Es poden fer servir de dues formes:
  - Compartint recursos (tots els PC són iguals)
  - Fent servir un model client-servidor.
- Un PC servidor proporciona recursos als PC clients. Existeixen diferents tipus de servidors:
  - Servidor de disc (espai).
  - Servidor de comunicacions.
  - Servidor de correu electrònic.
  - Servidor d'aplicacions.
  - Servidor de pàgines web.



Xarxes. Tipus MAN

MAN: Metropolitan area network

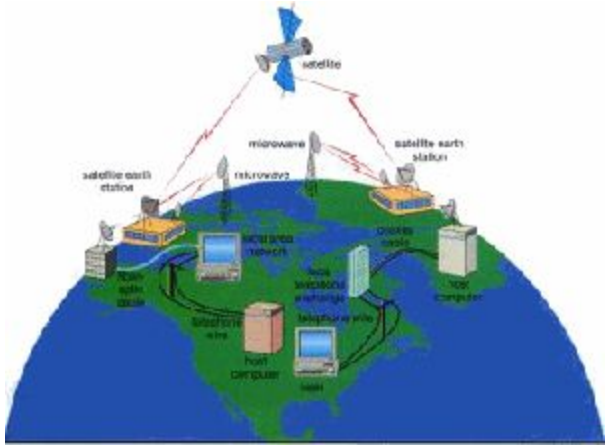
- Una xarxa d'àrea metropolitana és bàsicament una versió més gran d'una xarxa d'àrea local i normalment es basa en una tecnologia similar.
- Pot connectar un grup d'oficines corporatives properes o una ciutat i podria ser privada o pública.
- Té una extensió màxima de poques desenes de quilòmetres
- Tendeix a operar al voltant dels 10 Gbps.



Xarxes. Tipus WAN

WAN: Wide Area Network

- Són xarxes que comuniquen equips a una àrea geogràfica molt àmplia.
  Per exemple països o continents.
- Permeten una velocitat de transmissió molt alta (1 Tbps)
- Es caracteritzen per:
  - Ocupar una gran extensió de terreny.
  - Són línies públiques.
- Es fan servir per un gran número d'Usuaris de forma simultània.
- Capacitat de transmissió major que les LAN.
- Taxes d'error més elevades.



Xarxes. Mètodes de transmissió

Simplex. És unidireccional, només una de les dues estacions pot transmetre i l'altre només pot rebre (teclat, monitor, tv)

Semiduplex: Cada estació pot enviar i rebre però no al mateix temps, la capacitat total del canal és utilitzada per un dels dos dispositius que està o bé enviant o bé rebent (walkie-talkie)

Full-duplex (o duplex): Ambdues estacions poden enviar i rebre simultàniament, les senyals van en qualsevol direcció han de compartir la capacitat de l'enllaç

Dades submarines

Click

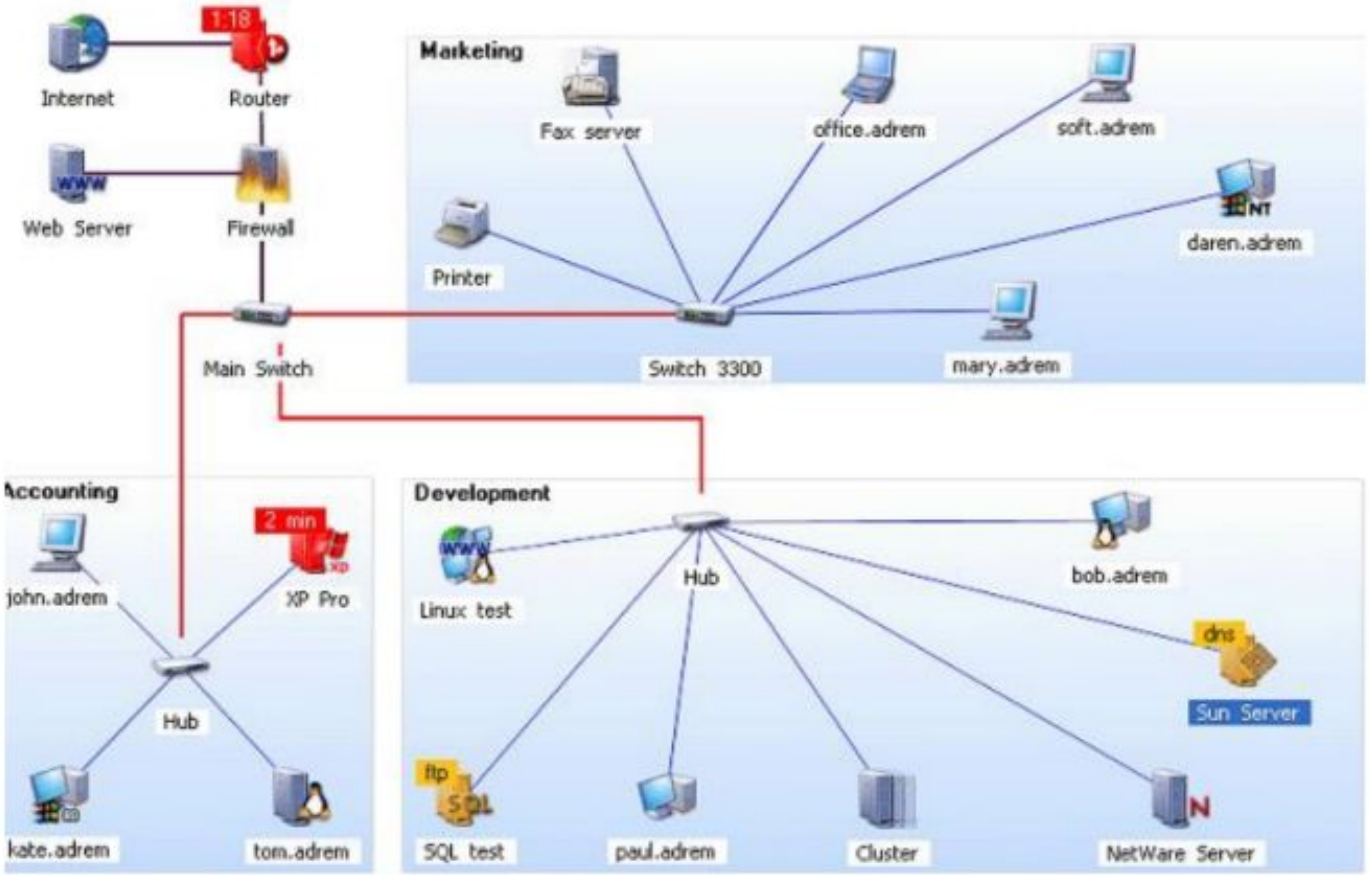
Mapa físic i lògic d’una xarxa

En cas de qualsevol problema, canvi o millora de la xarxa, és necessari tenir documentat correctament el sistema amb la informació el més detallada i actualitzada possible. Cada administrador de xarxa tria quines tècniques de documentació considera oportunes, però els següents documents no poden faltar:

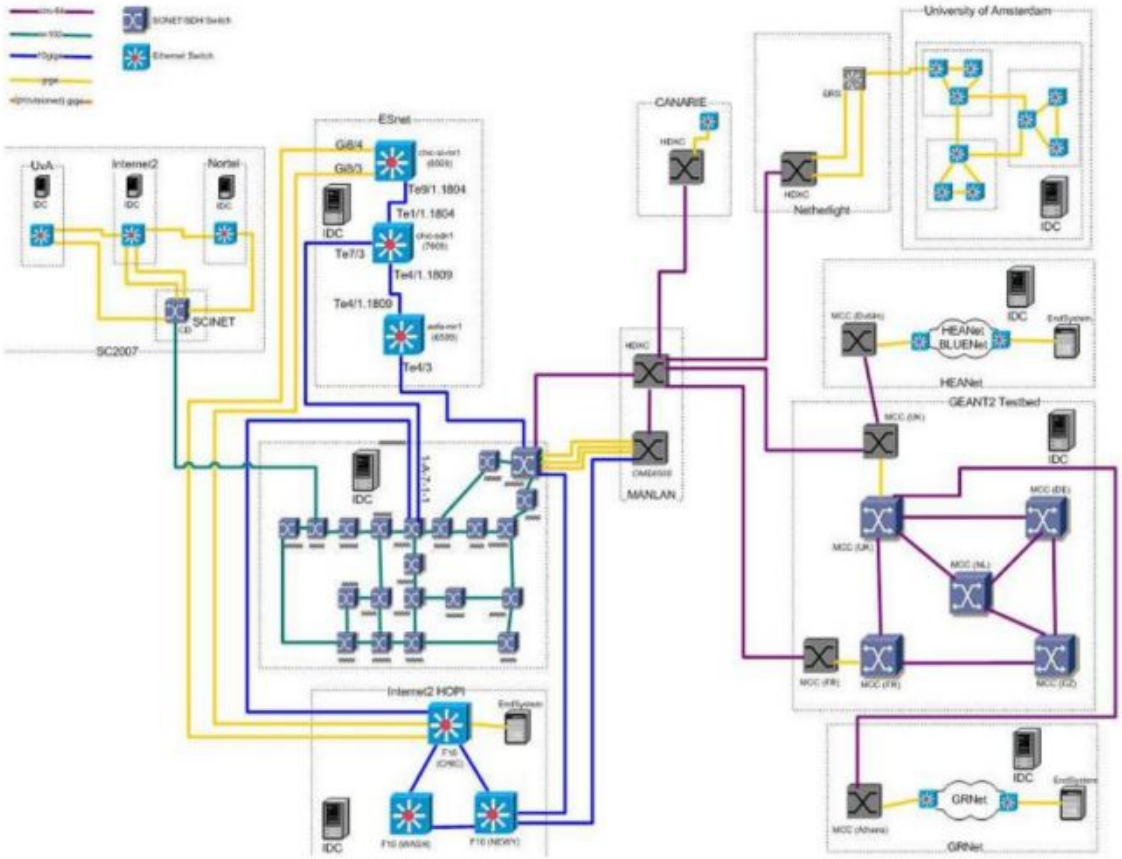
Mapa físic de la Xarxa: És la representació gràfica de la topologia de la xarxa, on s’inclouen les connexions internes i les externes. Aquesta documentació pot recolzar-se en un plànol de l’edifici on s’instal·la la xarxa. S’acostumen a fer-se dos tipus de mapes de xarxa:

- **Mapa lògic o funcional de la xarxa:** Aquest mapes indiquen la funcionalitat del diferents elements que es descriuen, així com les seves adreces i les funcions que realitzen.

- **Mapa físic de la xarxa:** Aquest mapa es centra en l’especificació de la connectivitat del cablejat.



Mapa lògic o funcional de la xarxa



Mapa físic de la xarxa

Mapa físic i lògic d’una xarxa

Mapa de nodes: Està format per una descripció del hardware i del software que s’instal·la en cada node, així com els paràmetres de la seva configuració, models, marques, adreces de xarxa... Aquesta documentació ha de permetre la creació d’un històric de cada node que registri l’evolució de les avaries, actualitzacions de software…

Mapa de protocols: És la descripció de l’organització lògica de la xarxa, així com dels protocols que es fan servir globalment, per exemple, la configuració dels enrutadors, dominis, grups de treball, relacions de confiança entre dominis..

Mapa de grups i usuaris: Consisteix en la descripció dels grups i usuaris de la xarxa tenint en compte les possibilitats d’accés als diferents recursos, així com els drets d’accés a les aplicacions, perfils, privilegis...

Mapa físic i lògic d’una xarxa

Mapa de recursos i serveis: Mostra tots els recursos disponibles identificant els seus noms, el servei que presten, el lloc físic o lògic en que resideixen i els usuaris o grups que tenen permís per accedir.

Calendari d’avaries: És el registre d’avaries del sistema, de forma que permeten un anàlisi de les possibles causes de errades dels diferents component de la xarxa, tant en software com en hardware i la seva evolució en el temps.

Informe de costos: És l’estudi econòmic tant del manteniment com de les noves inversions del sistema.

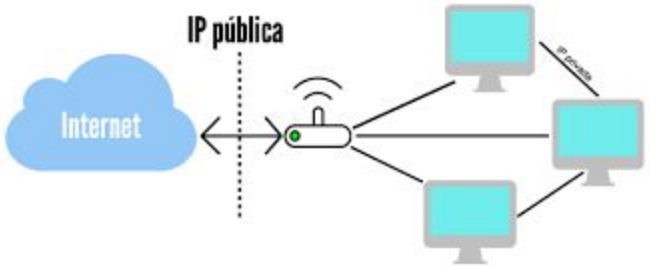
Pla de contingències: Base d’actuació davant qualsevol desastre.

Direccions IP

Xarxes. Protocol IP

- Aquest protocol ens envia paquets de la xarxa on s'originen fins a la xarxa de destí, necessitant un identificador per a les dues xarxes.
- Quan el paquet arriba a un router connectat a la xarxa de destí, IP ha de localitzar l'ordinador concret dins d'aquella xarxa. És, per tant, un procés de dos passos similar al d'entrega d'un correu en una oficina postal.
- Una adreça IPv4 és un número binari de 32 bits, separat per punts cada 8 bits i té dues parts ben diferenciades:
    - Identificador de xarxa
    - Identificador de Host dins la xarxa

[Protocol IP]



Xarxes. Direccions IP

Una direcció IP és com un número de telèfon o una adreça postal. Quan et connectes a Internet, el teu dispositiu té assignada una direcció d'IP (així com els llocs que visitem).

El sistema de direccionament que hem utilitzat des que va néixer internet és l'IPv4 i el nou sistema és el IPv6.
La raó d'aquest canvi és perquè amb IPv4 ens estàvem quedant sense direccions i IPv6 ens proporciona una llarga quantitat de direccions IP

Total d'espai IPv4: 4.294.967.296 direccions.
Total d'espai IPv6: 340.282.366.920.938.463.463.374.607.431.768.211.456 direccions.

Xarxes. IPv4

Perquè en IPv4 tenim aproximadament 4,3 mil milions de direccions?

Hem de descomposar una direcció IPv4 per entendre-ho.

- Una direcció IPv4 és un número de 32 bits format per quatre octets de 8 bits (4 bytes).
- Cada byte es transforma en la seva pròpia notació decimal separada per punts

  (ex: 192.168.1.0)
- D'aquests 32 bits uns formen part de l'identificador de xarxa i altres a l'identificador de host (ho veurem més endavant amb les classes de direcció IP)
- Entenem doncs, que si tenim 4 octets de 8 bits podrem arribar a tenir 256 possibilitats a cada secció (256*256*256*256 = 4.294.967.296). També ho podem llegir com 2^{32}

Xarxes. IPv6

Les direccions IPv6 estan basades en 128 bits. Fent servir la mateixa matemàtica que en el cas de les direccions IPv4 obtindrem 2^{128} direccions diferents.

IPv6 està composat per vuit seccions de 16 bits, separades per dos punts (:). A cada secció tindrem doncs 2^{16} variacions (65.536 possibilitats). Per simplificar la notació, en comptes de representar els nombres en decimal, el que farem serà fer-ho en hexadecimal (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

Exemple: 2607 : F0D0 : 4545 : 3 : 200 : F8FF : FE21 : 67CF

Segueix sent una expressió molt llarga però més fàcil d’intepretar que en nombres decimals.

Xarxes. Classes de direccions IPv4

A Internet hi ha molts tipus de xarxes en funció de l’organització a la que pertanyen.
El protocol IP ha de ser capaç de dimensionar qualsevol tipus de xarxa, des d’una multinacional amb moltes xarxes amb moltes màquines, fins a petites organitzacions que treballen amb un nombre petit de màquines.

Es van definir diferents tipus de xarxes:

- Classe A: xarxes grans, amb un nombre gran de màquines/hosts
- Classe B: xarxes mitjanes, amb un nombre moderat de màquines/hosts
- Classe C: xarxes petites, amb poques màquines/hosts

Addicionalment, existeixen dos tipus més de xarxes destinades a dos tasques ben diferenciades.

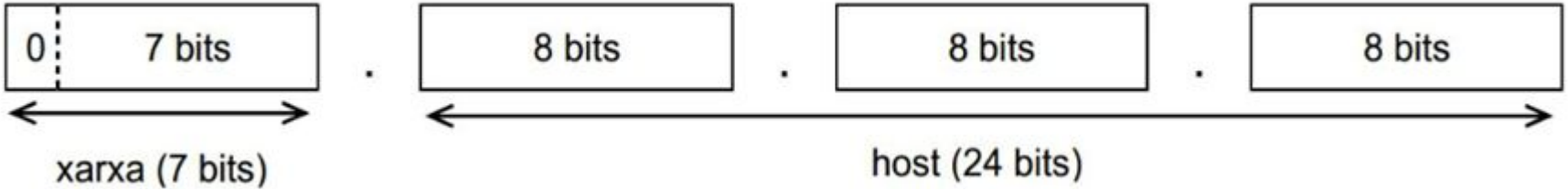
- Classe D: estan orientades al multicasting (és l'enviament de paquets d'informació a múltiples destinataris d'una xarxa informàtica)
- Classe E: estan orientades a fins experimentals.

Xarxes. Classes de direccions IPv4

Com podem identificar de quina classe és una IP? Ens hem de fixar en el primer número de la direcció IP (ex: **192**.168.100.50 - Classe C)

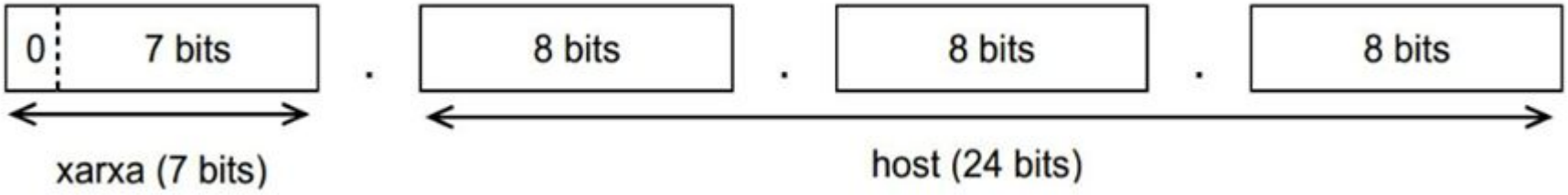
Classe A	El primer octet es troba entre 0-127
Classe B	El primer octet es troba entre 128-191
Classe C	El primer octet es troba entre 192-223
Classe D	El primer octet es troba entre 224-239
Classe E	El primer octet es troba entre 240-255

Xarxes. Classe A



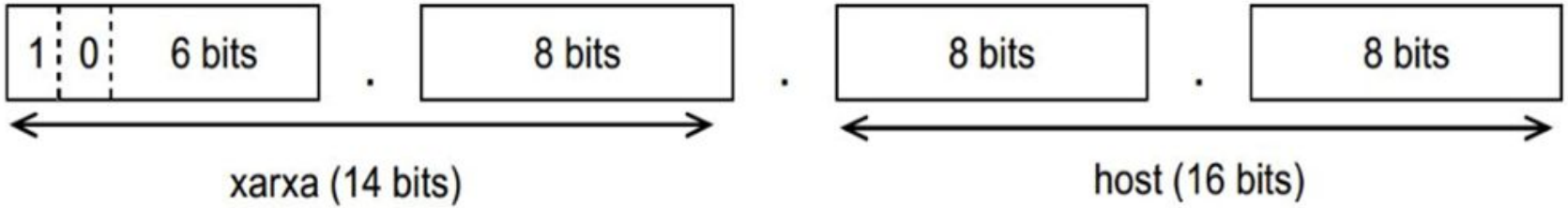
- 8 bits de Xarxa, 24 bits de Host
- 2^7-2 Xarxes (126), donat que:
  - 0.0.0.0 està reservat per a definir una xarxa per defecte
  - 127.0.0.0 - 127.255.255.255 estàn reservats com a direccions de loopback (representa el propi dispositiu).
- 2^{24}-2 Hosts (16777214), donat que:
  - X.0.0.0 serveix per a identificar la xarxa.
  - X.255.255.255 és ID que representa a tots els hosts de la xarxa X (broadcast)

Xarxes. Classe A



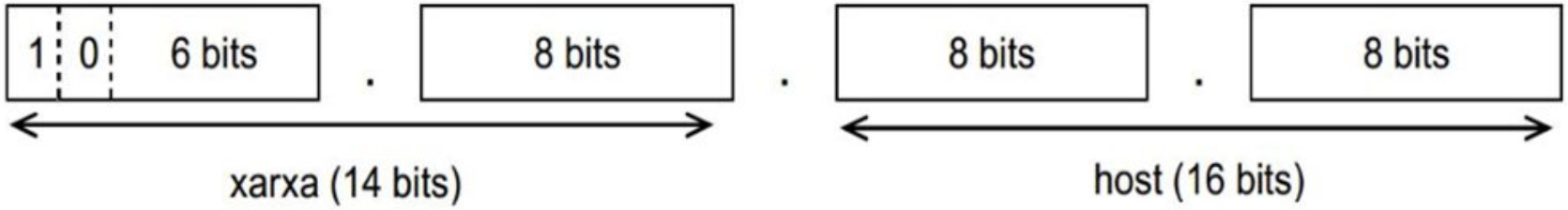
- X pot prendre valors entre 0 i 127: 00000000_2 i 01111111_2
- Rang d'IPs de classe A assignables (decimal)
    - De 1.0.0.0 fins a 127.255.255.255 (127.0.0.0 a 127.255.255.255 són reservats)
- Rang d'IPs de classe A (binari)
    - De **0**0000001.00000000.00000000.00000000
    - fins a **0**1111111.11111111.11111111.11111111

Xarxes. Classe B



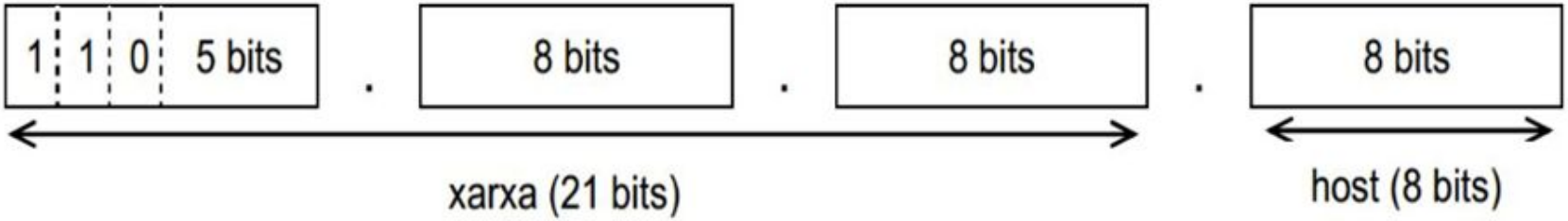
- 16 bits de Xarxa, 16 bits de Host
- 2^{14} xarxes (16384)
- 2^{16}-2 Hosts (65534), donat que:
  - X.Y.0.0 serveix per a identificar la xarxa.
  - X.Y.255.255 és ID que representa a tots els hosts de la xarxa X.Y (broadcast)
- X pot prendre valors entre 128 i 191: 10000000_2 i 10111111_2
- Y pot prendre valors entre 0 i 255: 00000000_2 i 11111111_2

Xarxes. Classe B



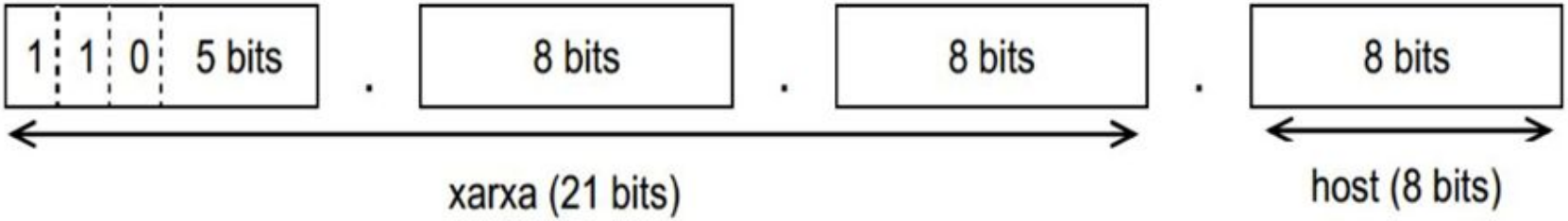
- Rang d'IPs de classe B (decimal)
  - De 128.0.0.0 fins a 191.255.255.255
- Rang d'IPs de classe A (binari)
  - De **10**000000.00000000.00000000.00000000
  - fins a **10**111111.11111111.11111111.11111111

Xarxes. Classe C



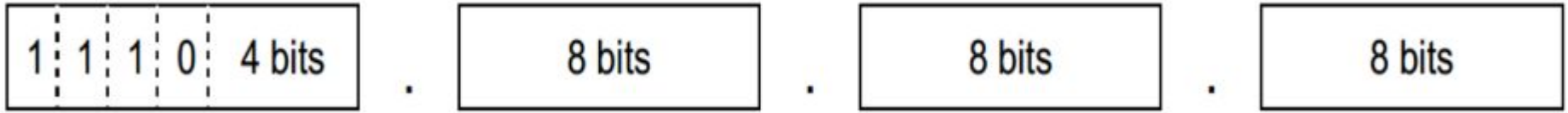
- 24 bits de Xarxa, 8 bits de Host
- 2^{21} xarxes (2097152)
- 2^{8}-2 Hosts (254), donat que:
    - X.Y.Z.0 serveix per a identificar la xarxa.
    - X.Y.Z.255 és ID que representa a tots els hosts de la xarxa X.Y.Z (broadcast)
- X pot prendre valors entre 192 i 223: 11000000_2 i 11011111_2
- Y i Z pot prendre valors entre 0 i 255: 00000000_2 i 11111111_2

Xarxes. Classe C



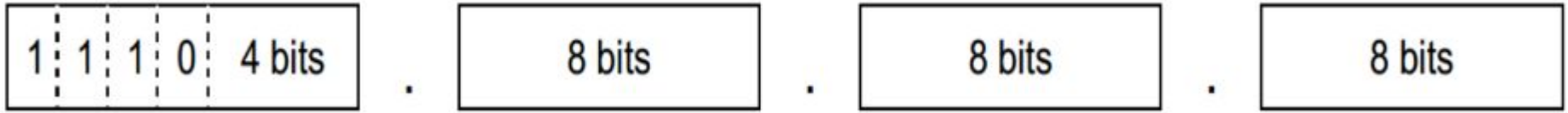
- Rang d'IPs de classe C (decimal)
  - De 192.0.0.0 fins a 223.255.255.255
- Rang d'IPs de classe A (binari)
  - De **110**00000.00000000.00000000.00000000
  - fins a **110**11111.11111111.11111111.11111111

Xarxes. Classe D



- És una classe definida per a suportar multicast.
- S'utilitzen per a enviar missatges a grups de hosts 'subscrits' a una determinada direcció IP de classe D.
- 1110 és l'identificador d' IP de classe D
    - La resta de bits (4b + 8b + 8b + 8b) = 28b són l'identificador de grup multicast.
    - Hi han 228 (268435456) grups multicast possibles.

Xarxes. Classe D



- Rang d'IPs de classe D (decimal)
  - De 224.0.0.0 fins a 239.255.255.255
- Rang d'IPs de classe D (binari)
  - De **1110**0000.00000000.00000000.00000000
  - Fins a **1110**1111.11111111.11111111.11111111

Xarxes. Classe E

- És una classe de tipus experimental.
- 1111 és l'identificador d' IP de classe E
- La resta de bits (4b + 8b + 8b + 8b) = 28b donen 228 (268435456)grups possibles d'IP.
- Rang d'IPs de classe E (decimal)
    - De 240.0.0.0 fins a 255.255.255.254, donat que:
    - 255.255.255.255 es fa servir per fer broadcast limitat

Xarxes. Classe E

- Rang d'IPs de classe E (binari)
  - De 11110000.00000000.00000000.00000000
  - fins a 11111111.11111111.11111111.11111110

Xarxes. Màscares

La màscara de xarxa és una combinació de bits que serveix per a delimitar l'àmbit d'una xarxa d'ordinadors.
És una seqüència de 32 bits que serveix per distingir amb facilitat quina part de l'adreça IP són bits de xarxa i quina part són bits de host.

- Es construeix, posant a 1 tots els bits que pertanyen a la part de xarxa i a 0 tots els bits que pertanyen a la part de host.
- L'Identificador de Xarxa és la part de la IP que encaixa amb els uns de la màscara

Classe A	11111111000000000000000000000000 → 255.0.0.0
Classe B	11111111111111110000000000000000 → 255.255.0.0
Classe C	11111111111111111111111100000000 → 255.255.255.0

La màscara de la direcció d'IP 192.168.168.100/24 com que és de classe C serà 255.255.255.0

Xarxes. Màscares

Si fem una opreració AND entre la direcció d'IP i la màscara obtindrem la direcció de xarxa a la que pertany aquesta xarxa.

Dir IP: 192.168.55.44 = 11000000.10101000.00110111.00101100
Dir màscara: 255.255.255.0 = 11111111.11111111.11111111.00000000 **(AND)**
192.168.55.0 = 11000000.10101000.00110111.00000000

La direcció de Xarxa serà: 192.168.55.0

La porció del host en una direcció de xarxa sempre serà 0

Xarxes. Màscares

- Sembla redundant assignar una màscara a una xarxa, sabent que a partir dels primers bits de l'adreça es pot saber a quin tipus pertany, però aquest sistema permet crear subxarxes (que es veuran més endavant)
- Exemple: 147.83.153.100/24
  - 24 és el nombre de bits a 1 a la màscara 11111111.11111111.11111111.00000000 classe C

Classe A	IP / 8. Ex: 100.10.5.1/8
Classe B	IP / 16. Ex: 128.100.4.10/16
Classe C	IP / 24. Ex: 200.1.30.5/24

Xarxes. Adreces reservades

IPs reservades que no es poden utilitzar per a un host:

- **Adreça de xarxa**: Una adreça de xarxa (IP que s'utilitza per especificar la xarxa) s'expressa posant tots els bits que corresponen a la part de host a 0.
- **Adreça 0.0.0.0:** S'utilitza com a xarxa per defecte (routers) o bé per escoltar tots els canals d'un determinat host (serveis).
- **Adreces 127.x.x.x:** S'utilitzen per operacions de loopback (normalment 127.0.0.1). Qualsevol paquet enviat a aquesta adreça s'envia automàticament al host que va enviar el paquet (a l'origen). És una adreça de classe A.
- **Adreça de Broadcast** Adreces amb els bits de Host a 1: Serveixen per fer broadcast dirigit. Adreça 255.255.255.255: Serveix per fer broadcast limitat.

Xarxes. Subxarxes

Quan s'utilitzen les subxarxes?
En el cas que vulguem administrar una oficina que estigui dividida en diversos departaments i que formin xarxes independents, necessitarem utilitzar la màscara de subxarxa.
Una subxarxa dins d'una xarxa es configura "agafant" bits de la part que correspon al host.
Per exemple:
255.255.255.0 → 11111111.11111111.11111111.00000000
255.255.255.192 → 11111111.11111111.11111111.11000000

En aquest cas, el que fem és "agafar" dos bits de la zona del host, de tal manera que podem crear $2^2 = 4$ subxarxes permetent l'existència de per exemple 4 departaments.
El número de IPs disponibles per hosts quedaria en $2^6-2=62$.

Restarem 2 perquè la primera direcció correspondrà amb la direcció de xarxa i l'última és la de multidifusió (broadcast)

Implementació IPv6

Click

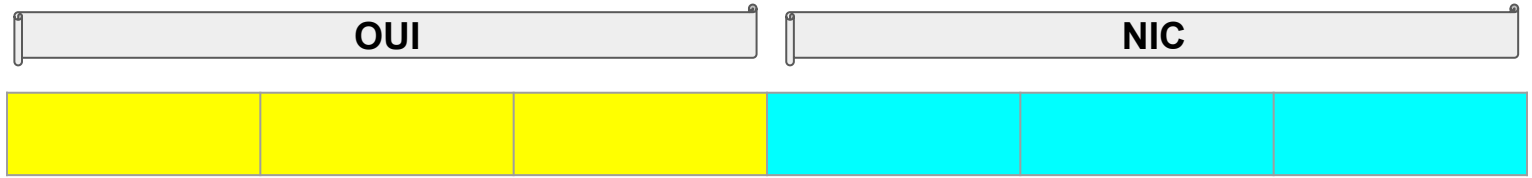
Adreces MAC

protocol ARP

Adreces MAC

- Les adreces MAC són adreces de la capa d’enllaç (capa 2 OSI)
- Les **adreces MAC són úniques**, assignades a les NIC.
- Aquest valor l’assigna el fabricant, i no es pot modificar.
- Són de **6 bytes**, generalment es visualitzen en hexadecimal:
  - Per exemple: 94-DE-80-13-45-F4
- Les xarxes LAN fan servir les adreces MAC.
  - **El Switch fa servir l’adreça MAC per enviar els paquets**.

Estructura adreces MAC: 6 bytes (48 bits)



Els tres primers bytes són l'OUI que identifica el **fabricant**.

Només 22 d'aquests bits corresponen a l'identificador del fabricant.

OUI: Organizationally Unique Identifier

Els 3 últims bytes els assigna el fabricant, de forma que cada NIC tingui un identificador únic.

NIC: Network Interface Card

Adreces MAC especials

Els 2 bits menys significatius del primer byte tenen significat específic

1 és reservat per adreces sense **OUI**, generades localment

b7	b6	b4	b4	b3	b2	b1	b0

1 és reservat per a broadcast i multicast

OUI

Els identificadors únics d’organització (*Organizationally Unique Identifiers*) permeten conèixer el fabricant d’una NIC (*Network Interface Card*) per la seva adreça MAC.

- Els 3 primers bytes identifiquen el fabricant.
- En internet trobem cercadors per identificar el fabricant amb un codi OUI.
- Molts fabricants tenen múltiples codis OUI assignats.

Funcionalitat adreces MAC assignades localment

Permet a l'**administrador de xarxa** assignar adreces MAC.

- Això es fa per definir **adreces MAC per a interfícies lògiques**.
  - Les interfícies lògiques es defineixen pel programari.
  - Per tant, no tenen adreça MAC amb OUI assignada, que s'assigna al maquinari.
  - Generalment seran adreces de la forma: **2-X-X-X-X-X**, però pot ser-ho qualsevol adreça amb el **segon bit del primer byte** activat (el número 2 és el cas més fàcil).
  - S'anomenen *locally administered addresses* (*LAA*)
- Les adreces MAC de les interfícies físiques no es poden canviar.
  - Tot i això, existeix la possibilitat de canviar l'adreça MAC d'un dispositiu físic; és el que es diu MAC spoofing, que es fa per a suplantar la identitat d'un ordinador, tant per motius legítims com per motius il·legals.

Adreces MAC per a multicast

Per enviar un missatge a tots els dispositius de la xarxa es fa servir l'adreça MAC de **broadcast**; Naturalment el valor del bit de multicast és 1:

BROADCAST

FF	**FF**	**FF**	**FF**	**FF**	**FF**

- La **resta d'adreces amb el bit de multicast activat** permeten definir **grups de multicast**.
- Els grups de multicast es defineixen també a les adreces IP de classe D (nivell 3 OSI).
  - Els missatges de multicast s'han d'enviar a una adreça de multicast, que inclou una adreça IP de multicast (classe D) i una adreça MAC de multicast.
  - Els destinataris s'han de subscriure a l'adreça de multicast per rebre els missatges de multicast.

Protocol ARP

- Per enviar un missatge dins de la nostra xarxa cal saber l’adreça MAC
- L’adreça que coneixen les aplicacions d’usuari són les adreces IP
- El protocol ARP serveix per descobrir l’adreça MAC quan coneixem l’adreça IP
- Algorisme:
  - Consultar la IP en la taula ARP. Si la trobem ja tenim l’adreça MAC.
  - Si no hi és, enviar un missatge de broadcast (adreça MAC FF-FF-FF-FF-FF-FF) amb l’adreça IP que volem descobrir.
  - Si obtenim resposta (el host amb l’adreça IP ha respost), afegir el resultat a la taula.

Determinar si una adreça IP pertany a la nostra xarxa

Abans d’aplicar el protocol ARP **cal determinar si l’adreça IP pertany a la nostra xarxa**.

- Aplicant l’operació AND lògica de la nostra IP amb la màscara de xarxa n’obtenim l’adreça de la nostra xarxa.

Una adreça IP pertany a la nostra xarxa quan aplicant-hi l’operació AND lògica amb la nostra màscara de xarxa n’obtenim l’adreça de la nostra xarxa.

Els missatges amb adreces IP que no pertanyen a la nostra xarxa s’han d’enviar a l’encaminador (passarel·la o *gateway*)

Exemple adreça IP i adreça de xarxa

- IP: **192.168.1.2**
- Màscara: **255.255.255.0**
- Notació compacta: **192.168.1.2 /24**
- Adreça de xarxa: **192.168.1.0**

- IP: **172.16.74.23**
- Màscara: **255.255.0.0**
- Notació compacta: **172.16.74.23 /16**
- Adreça de xarxa: **172.16.0.0**

L’adreça de xarxa s’ha obtingut fent AND lògic de l’adreça IP amb la màscara de xarxa.

Internet Protocol

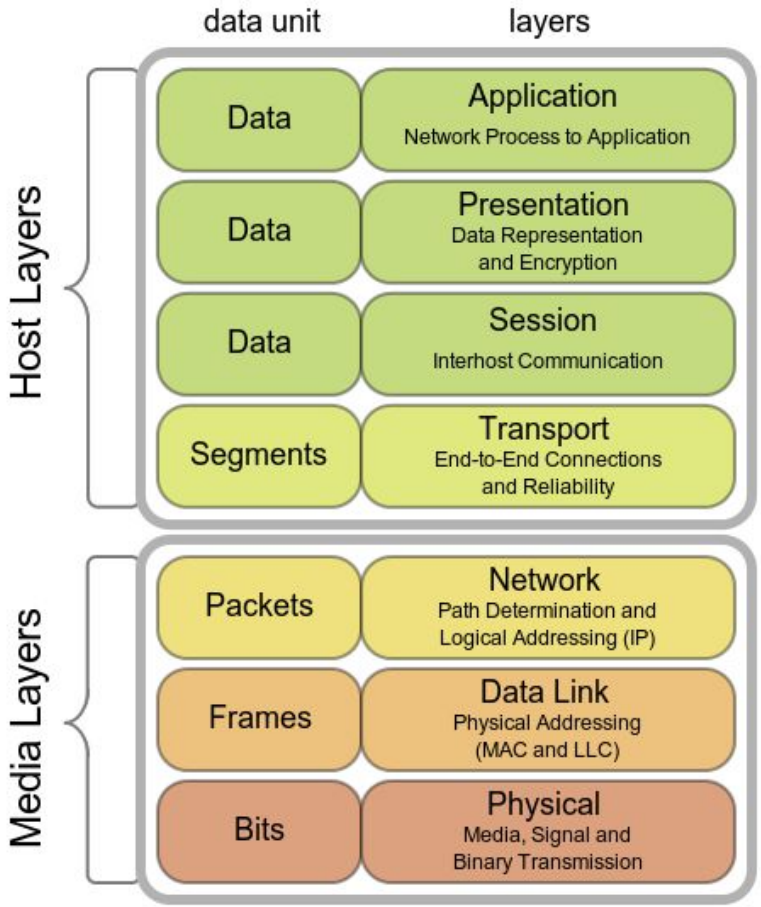
La capa ISO de xarxa (capa 3)

Capa física (nivell 1 OSI)

- **Interfície dels medis físics:**
  - Característiques mecàniques, com ara el tipus de cable i connectors.
  - Especificacions elèctriques (cablejat) o electromagnètiques (inalàmbriques o fibra òptica)
- **Ser capaç de comunicar un bit entre dos dispositius:**
  - Característiques de la comunicació:
    - Sèrie o Paral·lel.
    - Simplex, Half-Duplex o Duplex.
  - Sincronització en comunicació asíncrona
  - Codificació
  - Modulació
  - Multiplexació
- **Determinar inici i final d'una sèrie de bits.**
- **Detecció de portadora i de col·lisions en els protocols CSMA (Ethernet)**
- **També la commutació de circuits (línia telefònica tradicional)**

Capa d’enllaç de dades (nivell 2 OSI)

- Controlar l’**accés al medi físic** de transmissió per part dels dispositius que comparteixen el mateix canal.
- Comunicació en **trames**, segments d’informació.
- **Afegir l’adreça MAC en cadascuna de les trames** que es transmeten.
- Agrupar els bits que es reben en trames.
- Control d’errors transmissió.
- Descartar trames duplicades o errònies.
- També permet definir xarxes virtuals **VLAN** dins de la mateixa xarxa física, separant en diferents xarxes segments lògics de la xarxa que no han d’intercanviar dades.
    - L’aïllament afegeix seguretat i limita el tràfic, particularment de broadcast.



- 7. Aplicació
- 6. Presentació
- 5. Sessió
- 4. Transport
- **3. Xarxa**
- 2. Enllaç de dades
- 1. Físic

Internet Protocol (capa de xarxa, nivell 3 OSI)

- **Interconnectar les diferents xarxes**. (*Inter-net Protocol*)
- Estructura la informació a transmetre en **paquets** (*datagrames*).
  - El nivell d'enllaç és capaç de determinar l'inici i final d'una trama, així que la comunicació la podem fer en paquets.
- **Adreces IP**
- **Encaminament**
- **NAT** (*Network Address Translation*)

Adreces *IP*

Les adreces *IP* **identifiquen de forma única un dispositiu en la xarxa**.

- **Adreces públiques**: Identifiquen un dispositiu de forma única en Internet. Assignades per ***ICANN*** (*Internet Corporation for Assigned Names and Numbers*). Anteriorment l'organisme era IANA, que ara és part d'***ICANN***.
- **Adreces privades**: Identifiquen un dispositiu de forma única en una LAN.
  - L'administrador de la xarxa determina quina adreça s'assigna a cada dispositiu.
- Quan dues xarxes s'interconnecten amb un router no poden coincidir adreces.
  - Per aquest motiu en *IP* es defineixen rangs d'adreces privades.
  - Es pot fer servir ***NAT*** (***Network Address Translation***)

Adreces *IP*

- La **màscara de xarxa** divideix l'adreça Ip en dues parts:
  - Adreça de Xarxa
  - Adreça de Host
- Dues adreces queden "*reservades*":
  - Quan la porció de host (en binari) són tot zeros, tenim l'**adreça de xarxa**.
  - Quan la porció de host (en binari) són tot uns, tenim l'**adreça de broadcast** per aquesta xarxa.
- Les adreces dels hosts hi seran entre l'adreça de xarxa i l'adreça de broadcast.
  - Per conveni s'assigna la primera adreça a l'encaminador.

Máscara de red

- Porción de red y de host: por ej IP Clase C 192.168.1.0/24

Porción de Red							Porción de Host
192	.	168	.	1	.		0
11000000	.	10101000	.	00000001	.		00000000
255	.	255	.	255	.		0
11111111	.	11111111	.	11111111	.		00000000 = /24

Càlcul de la adreça de xarxa

Dirección de red resultante

Dirección IP	192 .	168 .	10 .	10
Binario	11000000	10101000	00001010	00001010
Máscara de subred	255 .	255 .	255 .	0
	11111111	11111111	11111111	00000000
Resultados de AND	11000000	10101000	00001010	00000000
Dirección de red	192 .	168 .	10 .	0

Cálculo del ID de red

Dirección IP en notación CIDR: 10.217.123.7/20

Dirección IP	10 .	217 .	123 .	7
	00001010	11011001	01111011	00000111
Máscara de subred	255 .	255 .	240 .	0
	11111111	11111111	11110000	00000000
ID de red	00001010	11011001	01110000	00000000
ID de red en notación CIDR	10.217.112.0/20			

Encaminadors (*routers*)

Per establir comunicació entre **diferents xarxes** es requereix un **encaminador**.

Dins una xarxa no caldrien adreces IP, perquè finalment les comunicacions es produeixen per les adreces MAC, però és més fàcil treballar amb adreces IP.

Per comunicar-nos amb qualsevol ordinador de qualsevol xarxa no podem fer servir adreces MAC, perquè caldria tenir la llista de tots els ordinadors del món, i les xarxes es colapsarien amb missatges.

Els **encaminadors** gestionen **la millor ruta** (*camí*) **per arribar a la xarxa de l'adreça destí**. Els encaminadors ***connecten xarxes i es connecten amb altres encaminadors***, *que connecten xarxes i….*

Porta d’enllaç (*gateway*)

- Quan s’envia un missatge es verifica si el destí pertany a **la mateixa xarxa** que l’emissor.
  - Això es fa aplicant la màscara a l’adreça destí, per veure si és la mateixa **adreça de xarxa** que l’emissor.
- Si és de la mateixa xarxa
  - Com que el destí hi és a la mateixa xarxa, és accessible a nivell d’enllaç.
  - Obtindrem l’adreça MAC (fent servir el protocol ARP) i podem enviar el paquet.
- **Quan el destí no pertany a la nostra xarxa**
  - Enviem el paquet a l’**adreça de la porta d’enllaç** (*gateway*)
  - En la porta d’enllaç es faran servir algorismes d’encaminament per assolir el destí.

Per tant, com a *gateway* farem servir l’encaminador, perquè faci arribar el missatge a una altra xarxa.

complement avançat

Capa de Xarxa (nivell 3 OSI)

- **Encaminadors** de xarxa (*porta d'enllaç*) fan servir **algoritmes per determinar les millors rutes** per a les dades (*circuits virtuals*).
- **Encaminament estàtic**
  - Rutes assignades manualment
- **Encaminament dinàmic**: adaptatius als canvis de la xarxa
  - **RIP** (*Routing Information Protocol*)
    - Fa servir el camí més curt, considerant el nombre de salts.
  - **OSPF** (*Open Shortest Path First*)
    - Pren en consideració la velocitat i congestió dels enllaços.

L'encaminament entre dues xarxes té múltiples camins possibles, perquè la xarxa té una topologia en malla. És per això que els algorismes d'encaminament dinàmic cerquen el millor camí.

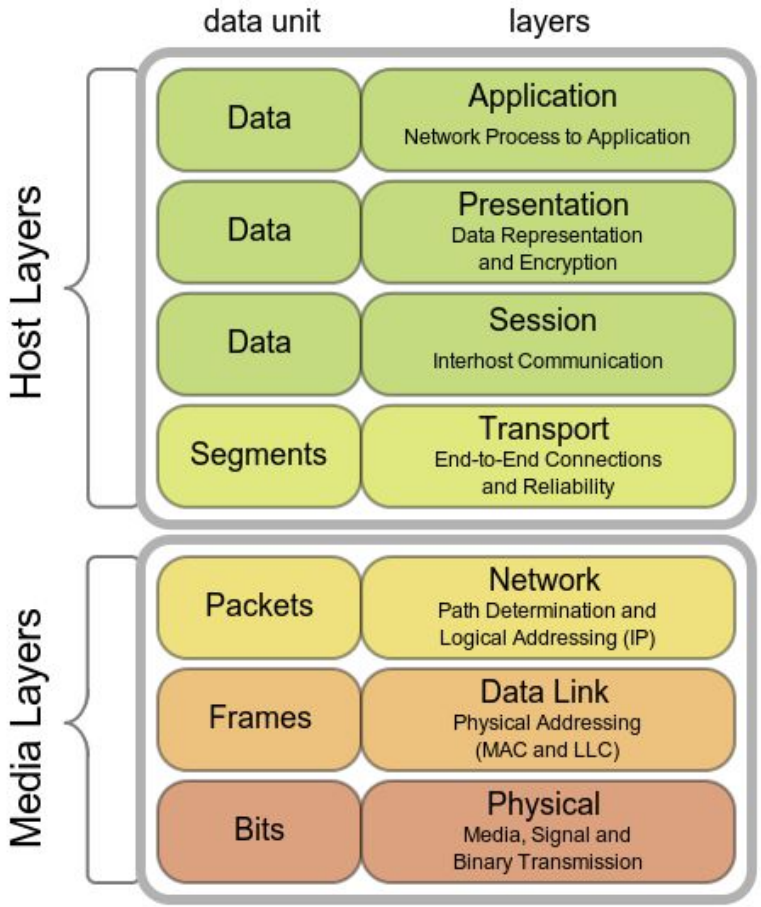
Altres funcionalitats de la capa de xarxa (nivell OSI 3)

- ***Network Address Translation***, característica que es descriu més endavant.
- També fa servir el protocol de missatges de control d'Internet (**ICMP**) per a la manipulació d'errors i per fer diagnòstics per assegurar que els paquets s'envien correctament (recordem l'aplicació ***Ping***, que fa servir *ICMP*).
- La ***qualitat del servei*** (**QoS**) també es gestiona en aquest nivell, per **permetre prioritzar** certs trànsits sobre altres.

TCP / UDP

Primera introducció a la capa de transport. Capa 4 OSI.

Referències:
http://www.highteck.net/EN/Transport/OSI_Transport_Layer.html
https://www.practicalnetworking.net/series/
https://microchipdeveloper.com/networking:start
https://www.homenethowto.com/



- 7. **Aplicació**
- 6. **Presentació**
- 5. **Sessió**
- **4. Transport**
- 3. **Xarxa**
- 2. **Enllaç de dades**
- 1. **Físic**

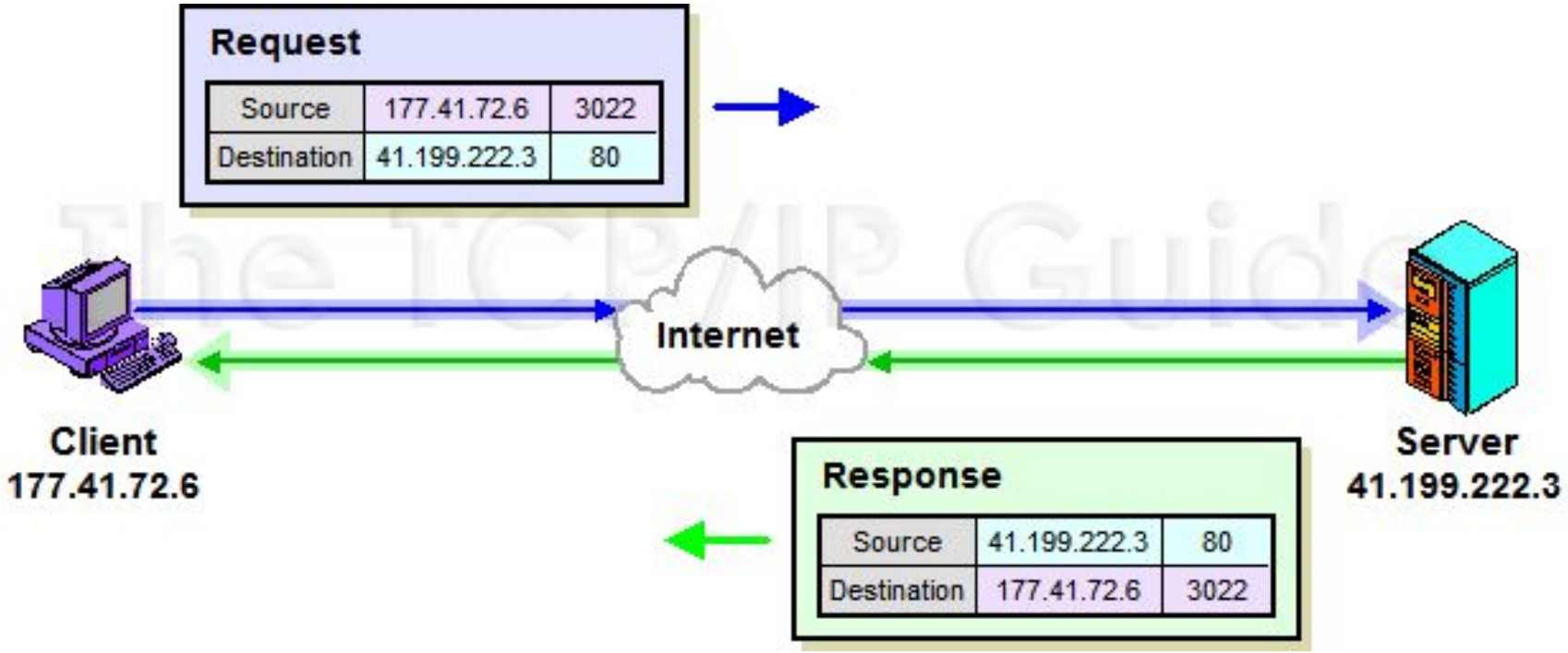
Funcionalitats de la capa de transport (OSI capa 4)

La capa de transport garanteix la transmissió de dades. A més, introdueix el concepte de **Port**. Un port **és un número** de 16 bits, per tant n'hi ha 2^{16} ports (65_536).

Els paquets en el nivell de transport es dirigeixen a un port d'una adreça IP, que apareix **en cada paquet**. Podem pensar en l'**adreça de transport** com la parella:

- L'**adreça IP** identifica el host de destí.
- El **port** és un identificador per vincular els missatges amb la interfície d'un **programari del host de destí**.

De la mateixa manera, el **remitent d'un paquet** de la capa de transport incorpora, a més de l'**adreça IP**, el **port** del programari remitent, per permetre la resposta.



Un exemple d'una petició Web http. El protocol ***http*** fa servir el ***port 80***.

Estat d’un port

En una comunicació es fa servir un port de l’emisor i un altre en el destí. Els ports tenen memòria intermèdia associada (*buffer*), tant per les dades rebudes com per les dades que s’envien.

Podem veure les comunicacions actives amb l’aplicatiu ***netstat***. Amb l’opció ***netstat -a*** veiem connexions TCP establertes i també els ports *oberts* (ports a l’aguait).

L’estat d’un port pot ser:

- **Obert**: Existeix una aplicació escoltant el port.
- **Tancat**: No n’hi ha cap aplicació escoltant.
- **Filtrat**: El tallafocs (*firewall*) no permet comunicacions fent servir aquest port.

Assignació dels ports

- ***Well Known Ports*** (Ports ben coneguts): Ports **fins el 1024**. Són serveis acordats que **gestiona el sistema operatiu**.
- ***Registered Ports*** (Ports registrats): Són ports que les fan servir diferents aplicacions; IANA manté una llista amb quin protocol fa servir cada port.
- ***Dynamic Ports / Private Ports***: Ports amb **número superior a *C000*** (hexadecimal). Ports que generalment s'assignen dinàmicament quan es crea la connexió.

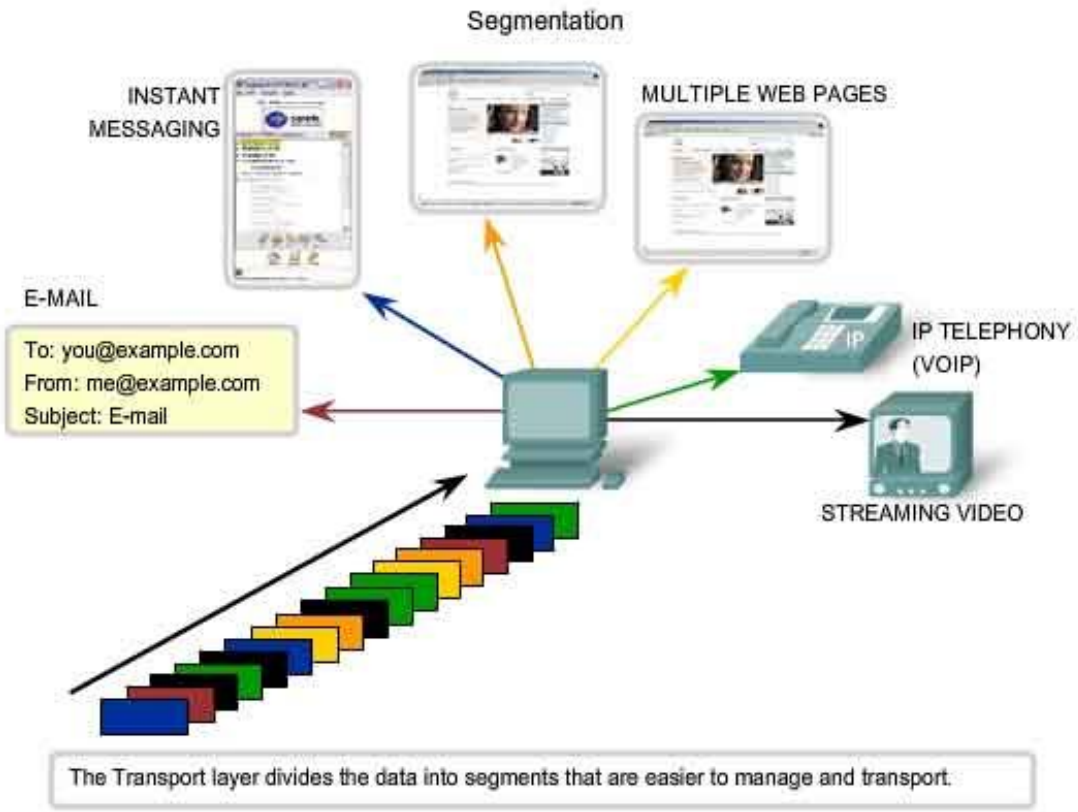
Els ports 25, 53, 80 i 443 són alguns dels ports més importants d'Internet, com podem veure més endavant en la llista.

Ports més freqüents

- **80 - HTTP** (*Hypertext Transfer Protocol*): Pàgines dels llocs Web.
- **443 - HTTPS** (*HTTP Secure*): Les pàgines dels llocs Web s'encripten i se autentiquen, garantint que provenen del servidor amb qui s'ha establert la connexió.
- **20, 21 - FTP** (*File Transfer Protocol*): Port 20 es fa servir per la transferència d'arxius, i el port 21 per les ordres i comandes.
- **67, 68 - DHCP** (Dynamic Host Configuration Protocol): Configuració automàtica: obtenir adreça IP, Gateway i servidor DNS. El port 67 correspon al servidor, i el 68 és el port del client. Fa servir missatges **UDP**.

Ports més freqüents (continuació)

- **23 - Telnet**: Connexió remota a un host en mode terminal.
- **22 - SSH** (*Secure Shell*): Connexió remota a un host en mode terminal, però incorpora encriptació i també transferència d'arxius.
- **25 - SMTP** (*Simple Mail Transfer Protocol*): Enviament de correu.
- **143 - IMAP** (*Internet Message Access Protocol*): Permet accedir al correu d'un servidor, sense haver de descarregar-lo (*correu Web*).
- **110 - POP3** (*Post Office Protocol Version 3*): Més antic que IMAP, protocol senzill que descarrega el correu del servidor i l'esborra.
- **53 - DNS** (Domain Name System): Servei de noms de domini. Fa servir **UDP**, però DNS també és capaç de funcionar per TCP, quan no pot fer-ho per UDP.
- **3389 - RDP** (*Remote Desktop Protocol*): Escriptori remot, només Windows.



Els **ports** actuen ***multiplexant*** el transport de dades de diferentes **aplicacions**.

És en les capes inferiors on els paquets IP i les trames *Ethernet* introdueixen la *multiplexació* que permet ***compartir el canal***. A més, les capes OSI superiors també poden afegir *multiplexació*; per exemple, podem obrir múltiples pàgines web simultàniament.

La **capa de transport** divideix les dades a transmetre en **paquets** (segments o *PDU*) que fa més fàcil la ***gestió d'errors***, i permet enviar les dades en ***paquets IP***, que s'envien en ***trames Ethernet***.

Per saber-ne més de consultar ports

L'aplicatiu **netstat** també ens pot informar quina aplicació fa servir el port (calen permisos d'administrador)

- En Windows: ***netstat -b***
- En Linux: ***netstat -p***

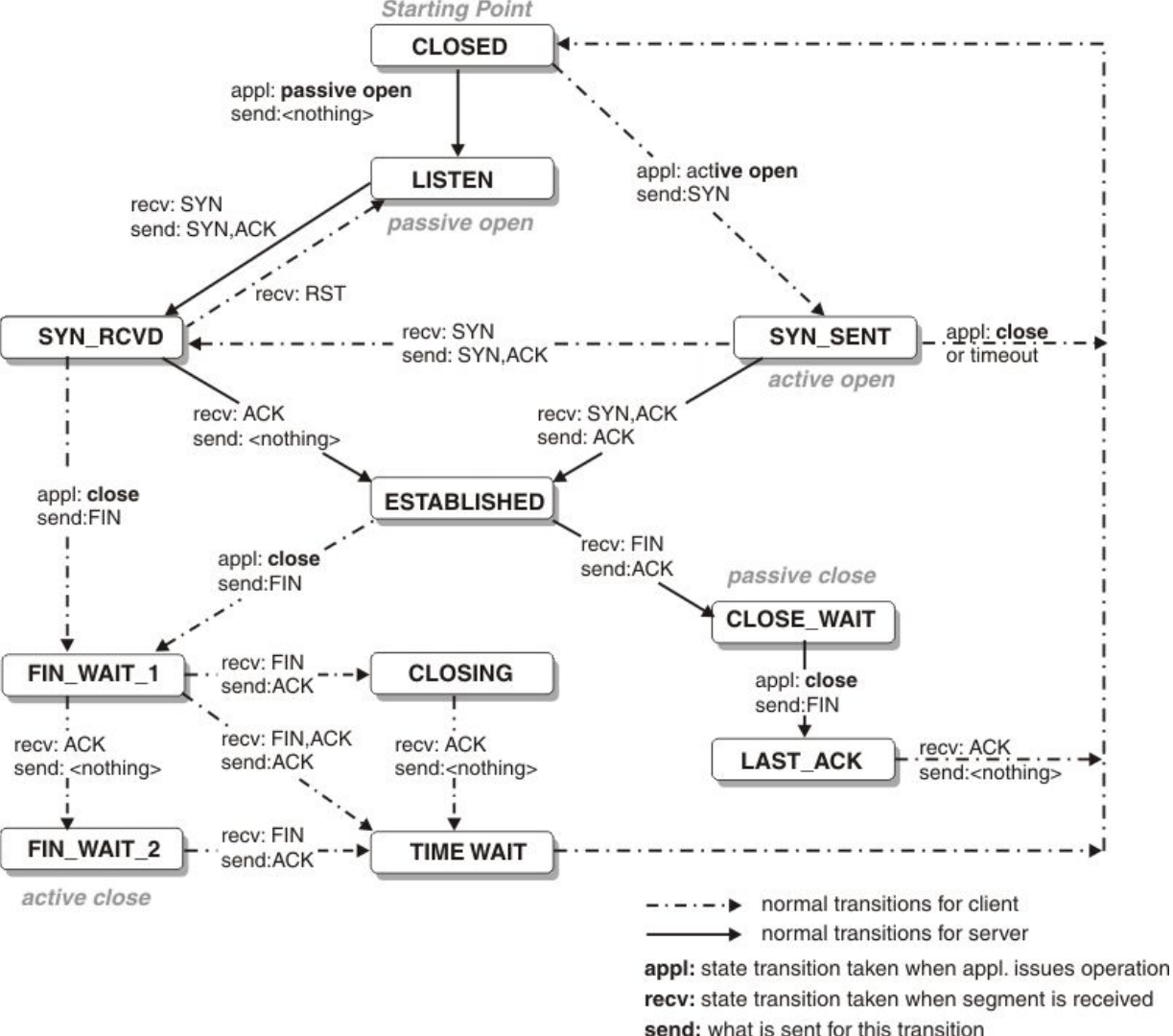
Es pot combinar amb l'opció -a

- En Windows: ***netstat -ab***
- En Linux: ***netstat -ap***

Per consultar l'estat dels ports, l'eina més coneguda per hackers i administradors de xarxa és **nmap**, que ens informa remotament de l'estat dels ports; així podem obtenir un informe de tota la xarxa, i detectar *vulnerabilitats* en qualsevol *host*.

TCP (*Transmission Control Protocol*)

- Orientat a la **connexió**:
  - Inicialment es produeix una **fase de connexió**, abans de la transferència de dades.
  - Una **fase de desconnexió** es produeix quan es finalitza:
    - Activament, amb una petició de desconnexió.
    - Pasivament, per time-out, si durant un temps no n'hi ha activitat.
- **Ordena i confirma** els paquets rebuts, **garantint que s'han rebut tots els paquets i en l'ordre correcte**.
- Inclou una **suma de control** (*checksum*) en cada paquet TCP, per garantir que les **dades de cada paquet s'han rebut sense errors**.

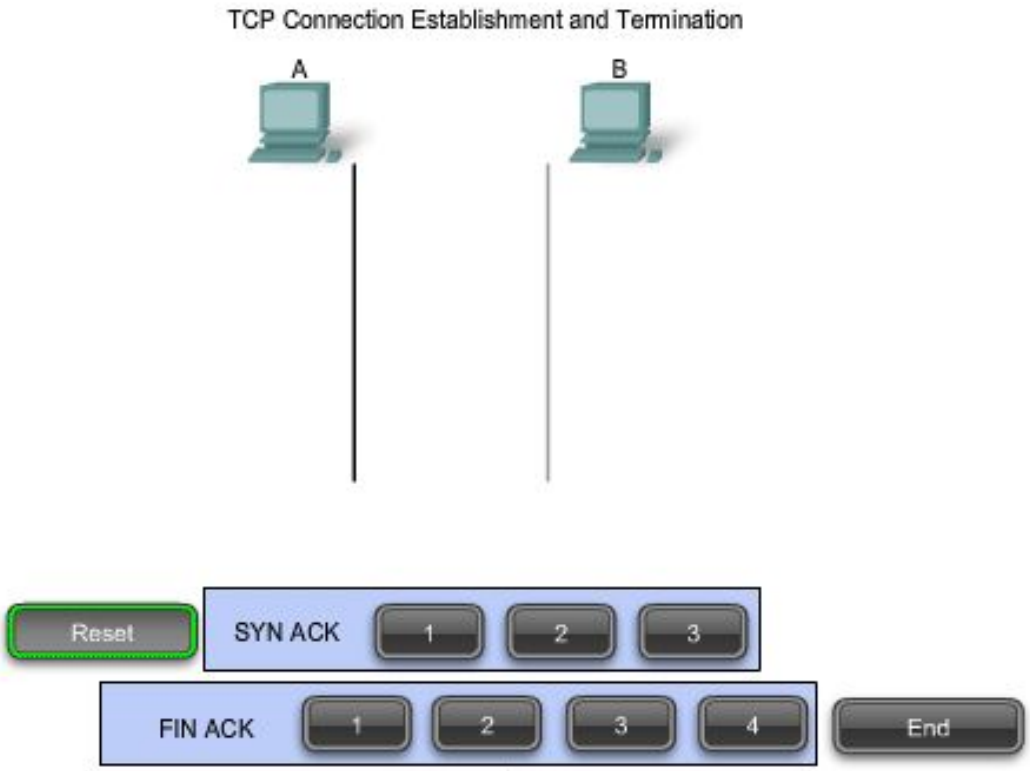


Per saber-ne més de les connexions TCP:

Estats d'una ***connexió***

TCP

La part esquerra del gràfic és qui rep la petició de connexió, que es troba *ESCOLTANT* (*LISTEN*), i la dreta qui emet la petició.



Per saber-ne més de les connexions TCP:

Missatges que s'intercanvien en les fases de ***connexió*** i ***desconnexió*** de **TCP**

Connexió en el nivell de capa física (nivell 1 OSI)

Les **xarxes telefòniques commutades** (***commutació de circuits***) proporcionen connexió entre dos punts (nivell físic) quan fem una trucada telefònica.

Es fan servir senyals de control per establir la connexió: **pulsos** (*obsolet*) o **tons**. Aquests senyals permeten que la connexió física s'estableixi punt a punt, fent servir mecanismes electrònics de commutació. Un **modem** faria servir aquests senyals per establir una connexió (*de nivell 1*) entre dos ordinadors o dues xarxes.

En canvi les connexions ***VoIP***, telefonia per Internet, com ara ***Skype***, i ara ja alguna companyia telefònica, fan servir **connexions** de ***nivell de transport*** (**TCP, OSI nivell 4**), o fins i tot de *nivell d'aplicació*, fent servir una xarxa de *commutació de paquets* (*Internet Protocol*).

Connexió en el nivell d’enllaç de dades (nivell 2 OSI)

Les connexions de nivell 2 s’anomenen ***Circuits Virtuals***

Per exemple, *Frame Relay* és una tecnologia que ofereix *Circuits Virtuals Permanents*, circuits predefinits que no cal establir i alliberar, en la capa d’enllaç de dades.

El concepte d’***Ethernet*** (CSMA/CD i CSMA/CA) és de medi compartit, són **protocols sense connexió**.

Connexió en el nivell de xarxa (nivell 3 OSI)

IP (***Internet Protocol***) és un protocol **sense connexió**.

La capa de xarxa pot admetre xarxes orientades a connexió o sense connexió (*recordem que la capa de xarxa interconnecta diferents xarxes*), però cada xarxa només pot ser d'un tipus.

Fiabilitat de la comunicació

Mecanismes de control d’errors de les capes inferiors:

- **Control de paritat** (*nivell físic, capa 1*)
- **CRC** (*Control de redundància cíclica*) valida les dades de les trames d’Ethernet (*capa d’enllaç de dades, capa 2*)

En la capa de transport s’afegeix un **Checksum** per a cada paquet.

A més a més, **TCP confirma els paquets**, de manera que els paquets que no arriben, o que arriben amb error, l’emissor els reenvia.

La connexió permet garantir el correcte ordre dels paquets, i que no s’ha perdut cap.

UDP (*User Datagram Protocol*)

- Protocol **sense connexió**.
- No ofereix garantia de que els paquets hagin estat entregats.
- No ofereix garantia en l'ordre en que paquets enviats seran rebuts.
- Senzillament, a les trames IP afegeix:
  - **Checksum** per garantir que en cada paquet les dades es reben sense errors.
  - Multiplexació d'aplicacions, fent servir **ports**.

És un protocol ***senzill***, amb menys sobrecàrrega que TCP.

És molt útil per missatges petits, de només un paquet, però cal considerar la situació que no arribi el paquet o arribi danyat. També es fa servir per *streaming* en temps real, perquè si un paquet no arriba es millor descartar-lo que esperar.

Els ports de capa 4 (*transport*) en la capa 3 (*xarxa*)

La capa 3 és capaç de fer la traducció d'adreces de xarxa (**NAT** *Network Address Translation*).

La principal finalitat és la **traducció d'adreces privades en adreces públiques**, amb dos objectius:

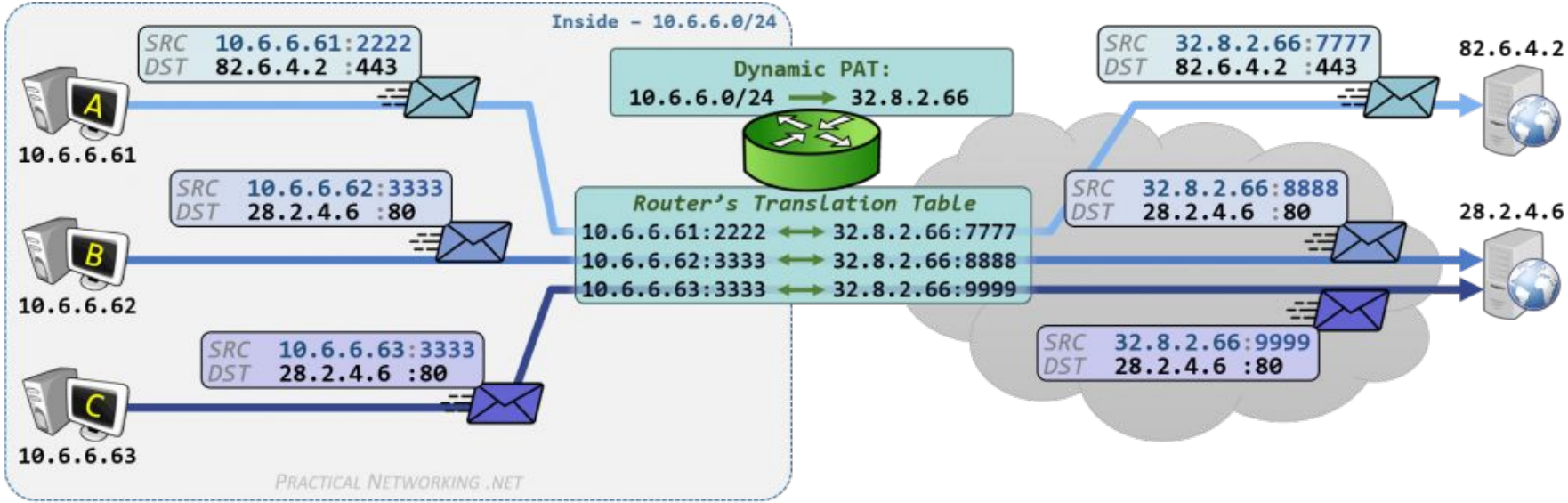
- Les adreces IP públiques són escases, i fent servir NAT podem ***compartir una adreça pública*** per a molts hosts d'una xarxa privada.
- ***Seguretat***: Si un host **no té adreça pública, no és accessible des de l'exterior**, quedant així amagada la xarxa privada darrera el servidor NAT.

Tipus de NAT

Tenim principalment dos tipus de **NAT**, i tots dos fan servir els ports de la capa de transport (TCP i UDP) :

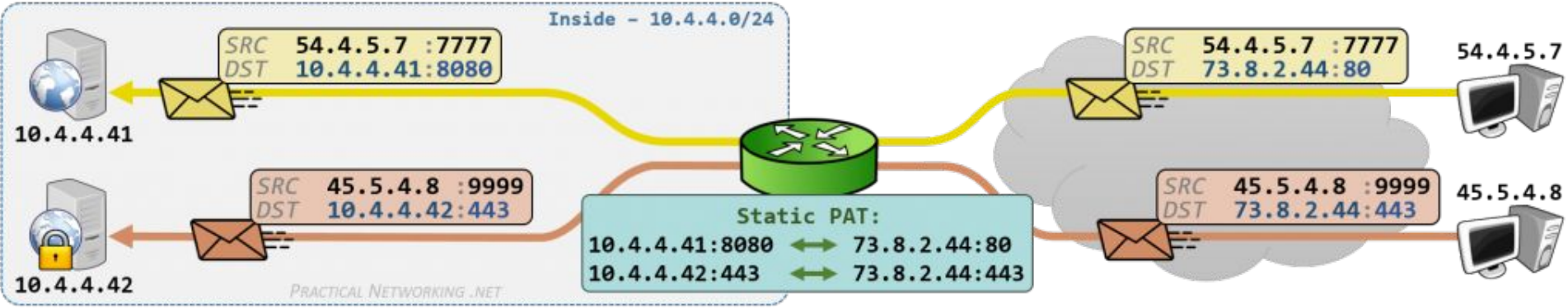
- **PAT** (*Port Address Translation*): Es fa servir per traduir l'**adreça d'origen**, convertint ***adreces IP privades en una adreça IP pública***.
  - De vegades també s'anomena ***IP masquerading***, o també ***dynamic PAT***.
- **DNAT** (*Destination Network Address Translation*): Es fa servir per traduir una **adreça de destí (servidor)** en una adreça privada, i permet tenir servidors especialitzats per a cada servei, compartint la mateixa adreça IP pública. Els servidors tindran adreces privades que amaguen l'accés a la resta de ports.
  - De vegades també s'anomena ***Port forwarding***, o també ***static PAT***.

PAT (també anomenat ***IP masquerading***, o també ***dynamic PAT***)



Moltes adreces privades comparteixen una mateixa adreça pública, fent servir un diferent port per a cada parella *adreça-port* mapejada.

DNAT (també anomenat *Port forwarding*, o també *static PAT*)



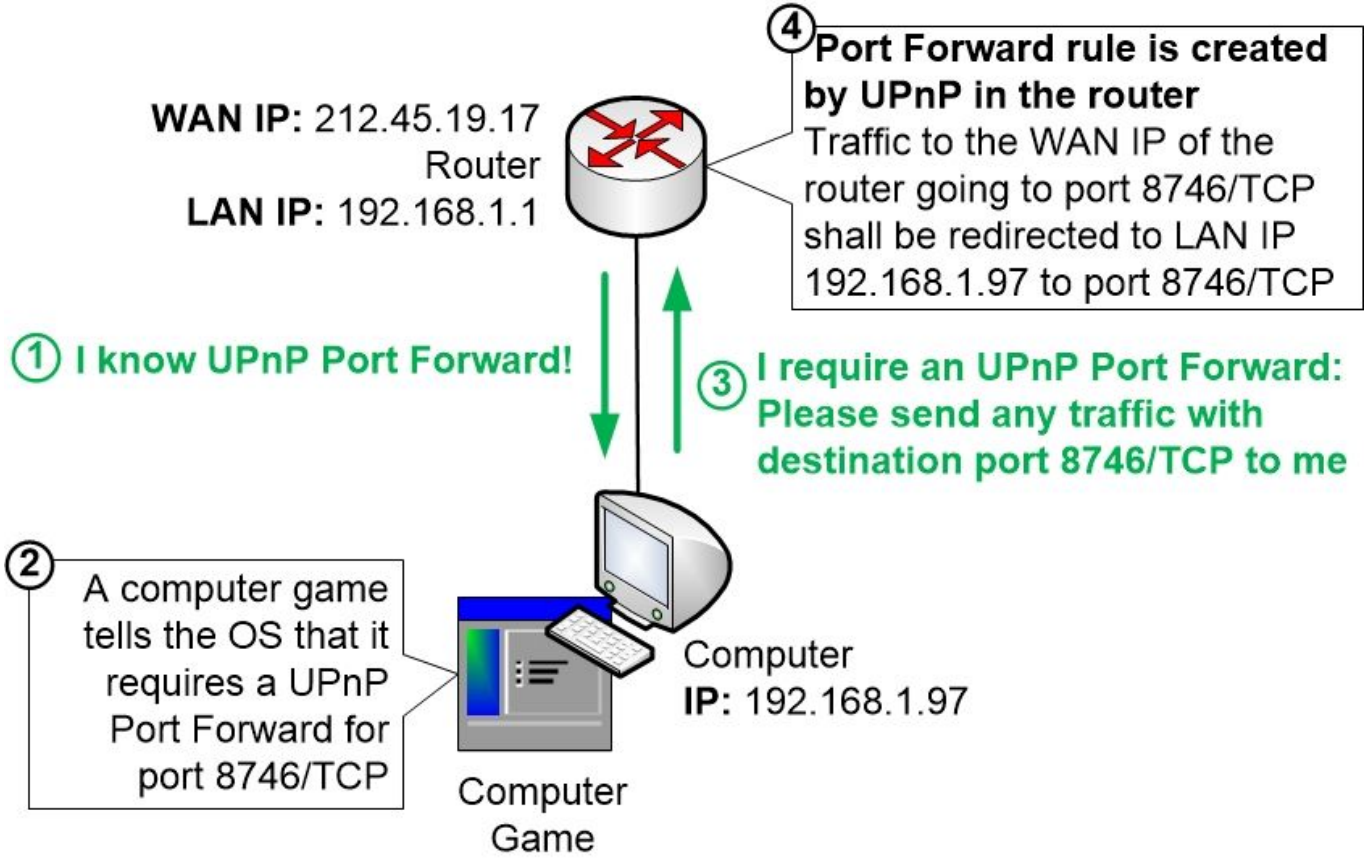
Es fa servir per mapejar en una **mateixa adreça** diferents **serveis oferts per diferents servidors**. Cada servei es correspon amb un port, i assignem una adreça local diferent per a cada port.

En la imatge d'exemple, un servidor HTTP (port 80) i un HTTPS (port 443).

Per saber-ne més:

UPnP port forward

Universal Plug&Play port forward (DNAT)



UF1-NF3
Direccionamiento IP

Eduard Lara

INDICE

1. Clases direcciones IP
2. Direcciones especiales
3. Colapso direcciones IPv4. IPv6
4. Concepto de Mascara
5. Subnetting
6. VLSM

ASIGNACIÓN DIRECCIONES IP

- ICANN, es el organismo encargado de asignarlas. Asigna bloques de direcciones a los Regional Internet Registries (RIR):
  - RIPE: Europa y África del Norte (http://www.iana.net).
  - ARIN: USA (http://www.arin.net).
  - APNIC: ASIA (http://www.apnic.net).
  - LACNIC: América Latina (http://www.lacnic.net).
- RIPE asigna grupos de direcciones IP a los operadores de Telecomunicaciones (Operadoras, Telefónica), éstos a su vez a los ISP (wanadoo.es, ya.com) y a sus abonados.
- Estas direcciones se llaman públicas, globales o registradas. Son únicas en todo Internet

NOTACIÓN DIRECCIÓN IP

- Cada dirección IP consta de 32 bits agrupados en grupos de 8 bits (8 bits son un octeto o un byte)
- La notación que se utiliza para expresar una dirección IP se conoce como *notación por puntos:* Consiste en expresar los 4 octetos de la dirección en formato decimal separados por puntos.
- Cada uno de estos números varia entre 0 y 255, por ejemplo

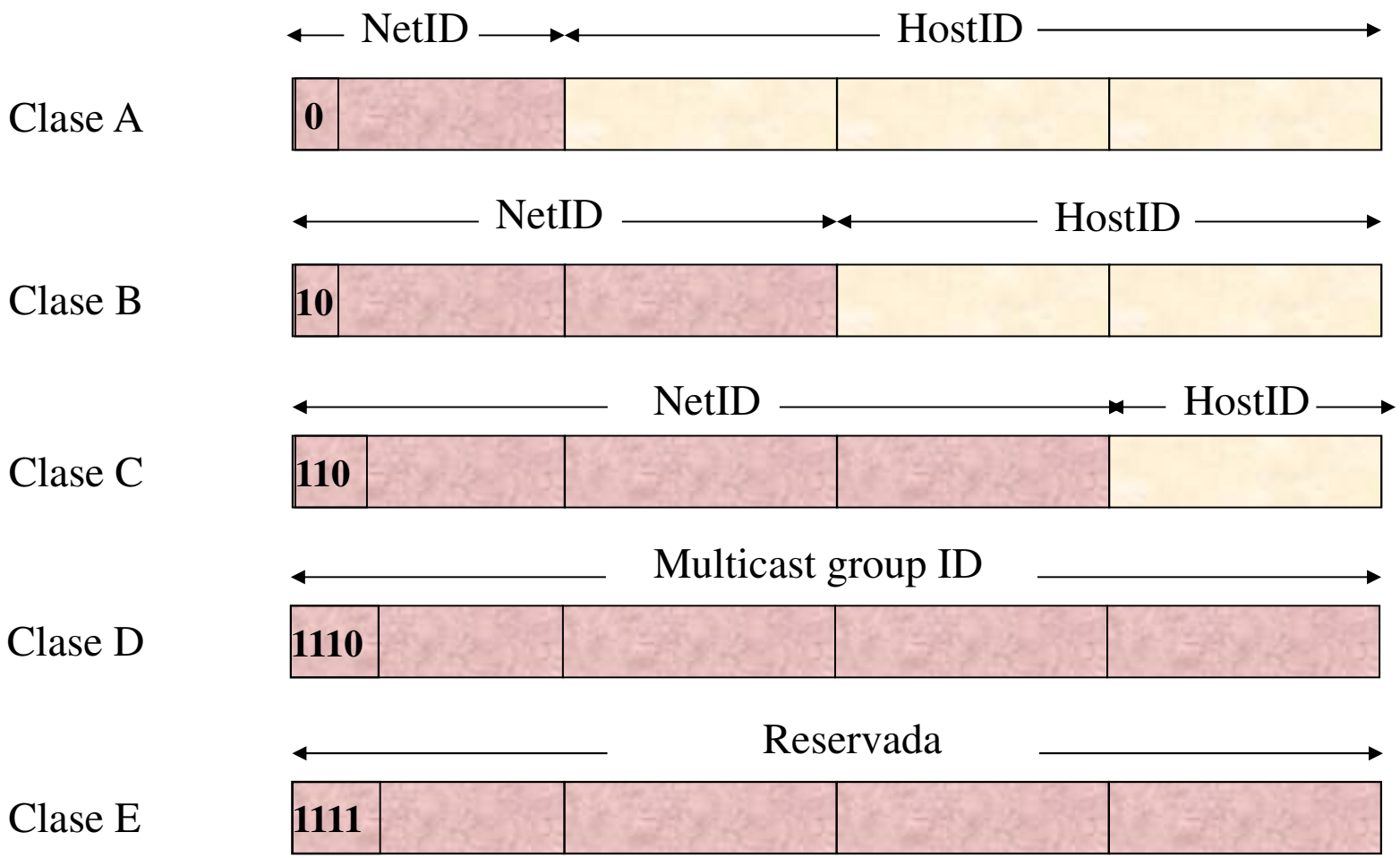
147.83.34.25

10010011. 01010011. 0100010. 00011001

NOTACIÓN DIRECCIÓN IP

- Como IP es un protocolo pensado para la interconexión de subredes, toda dirección IP contiene dos identificadores:
  - NetID: Porción de la dirección IP que identifica la red o subred a la que pertenece
  - HostID: Identifica una máquina dentro la red
- Atendiendo a los primeros bits de cada dirección se averigua el tipo de subred de que se trata. Los bits restantes codifican el *host* dentro de esa subred.

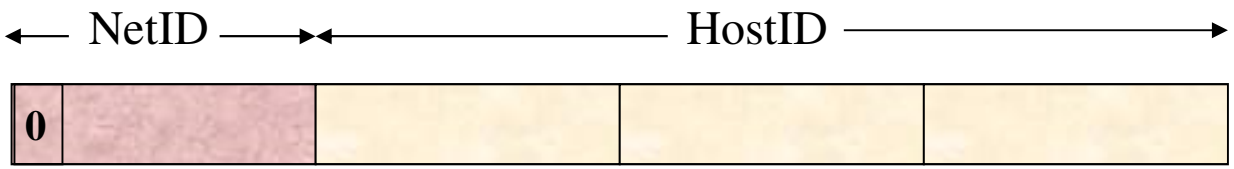
CLASES DIRECCIONES IP



CLASES DIRECCIONES IP

Clase	NetID (bytes)	hostID (bytes)	1º byte	Codificación
A	1	3	0 – 127	0xxx…xxx
B	2	2	128 – 191	10xx…xxx
C	3	1	192 – 223	110x….xxx
D	-	-	224 – 239	1110x…xxx
E	-	-	240 - 255	1111x…xxx

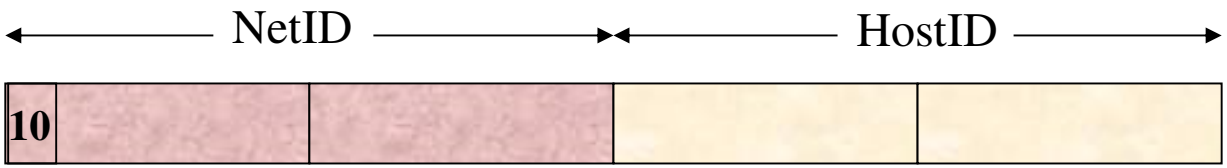
DIRECCIONES CLASE A



- Se caracterizan porque el primer bit es un 0
- Los siete bits siguientes codifican la subred y los 24 restantes al *host* dentro de esa subred.
- Van de 0.0.0.0 hasta 127.255.255.255
- Nº redes = 128 subredes posibles de tipo A.
- Nº hosts/red = $2^{24} - 2$ = 16.777.214 *hosts* distintos.

Sistema de direccionamiento utilizado en redes grandes

DIRECCIONES CLASE B



- Se caracterizan porque los dos primeros bits de la dirección son 10.
- Los 14 bits siguientes codifican la subred y los 16 restantes al *host* dentro de esa subred.
- Van de 128.0.0.0 hasta 191.255.255.255
- Nº redes = 64 x 256 = 16.384 subredes de tipo B.
- Nº hosts/red = $2^{16} - 2$ = 65.534 *hosts* distintos

DIRECCIONES CLASE C

- Se caracterizan por tener sus tres primeros bits con el valor 110.
- Los 21 bits siguientes codifican la subred y los 8 restantes el *host* dentro de la subred.
- Van de 192.0.0.0 a 223.255.255.255
- Nº subredes = 32 x 256 x 256 = 2.097.152 redes
- Nº de hosts/red = 2^8 - 2 = 254 host/red

DIRECCIONES CLASE D y E

- Clase D (Multicast): Las direcciones que comienzan por la secuencia 1110, se tratan de direcciones de multidifusión, es decir, una dirección especial en donde el destinatario no es único.
  - Van de 224.0.0.0 a 239.255.255.255
- Clase E (Usos futuros): Las direcciones que comienzan por 1111 se reservan para protocolos especiales, como los de administración de grupos de internet, multitransmisión y otras futuras implementaciones.
  - Van de 240.0.0.0 a 255.255.255.255

ASIGNACIÓN DIRECCIONES IP

La asignación de las direcciones se hace teniendo en cuenta que:

- Una dirección identifica una interfaz.
- Todas las direcciones/interfaces de una misma red IP (segmento de red) tienen el mismo NetID.
- El router ha de tener asignada una dirección IP por cada interfaz de salida. El NetID de cada dirección asignada a una interfaz debe tener el NetID de la red donde esta conectada.

DIRECCIONES ESPECIALES

- No todas las direcciones se pueden utilizar para numerar las interfaces. Algunas son especiales.
- Cada subred tiene dos: la de red y la de broadcast.
  - **Dirección de red**: HostID = todo 0's. Identifica una red. Se utiliza en las tablas de encaminamiento.
  - **Dirección broadcast**: HostID = todo 1's

DIRECCIONES ESPECIALES

- **Dirección 0.0.0.0:** Identifica "este host" en "esta red". Se utiliza como dirección origen en protocolos de configuración (BOOTP, DHCP). Nunca se utiliza como dirección destino.
- **Dirección 255.255.255.255:** Indica broadcast en "esta red". Se utiliza como dirección destino en protocolos de autoconfiguración.
- **Interfície loopback 127.0.0.1**: Interfaz que permite a un cliente comunicarse con un servidor dentro de la misma maquina a través de TCP/IP, sin tener que usar una tarjeta de red.

DIRECCIONES PRIVADAS

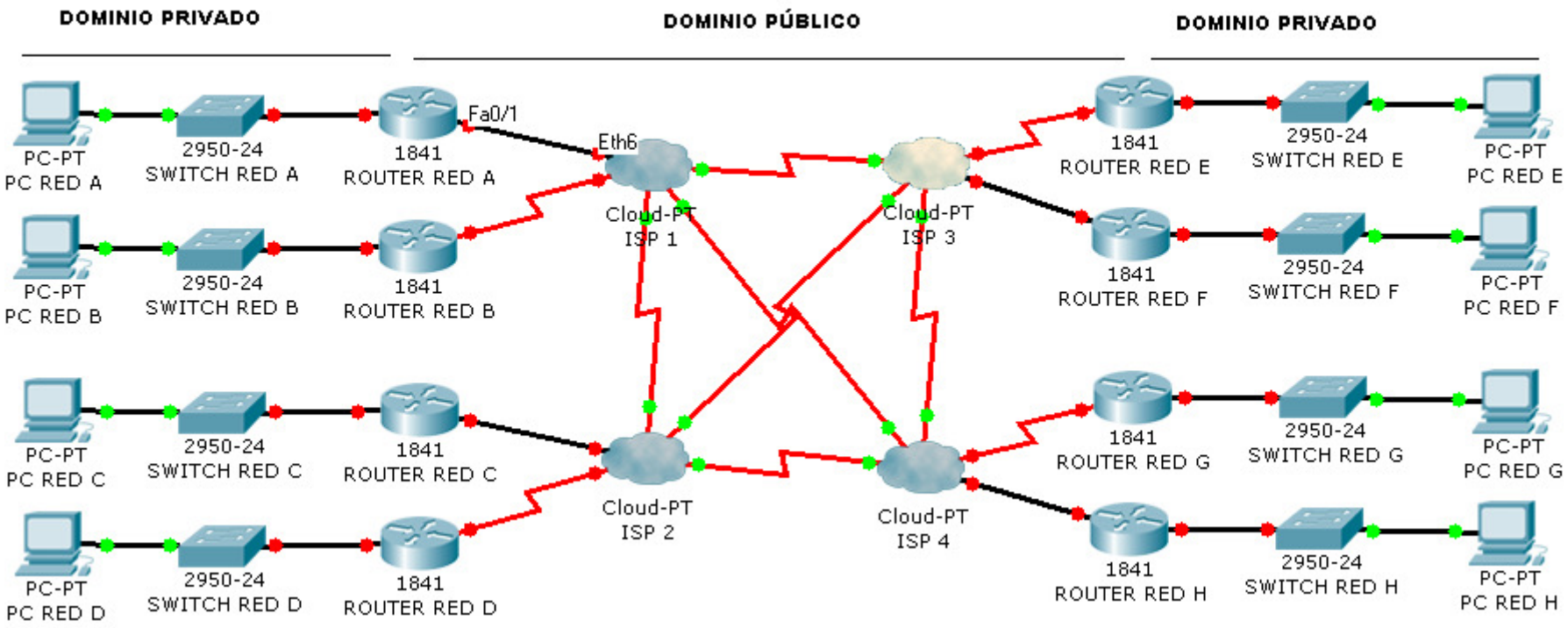
- Son direcciones para el uso particular de la red interna: en redes o hosts que hacen servir TCP/IP y que no tienen una dirección pública.
- Son direcciones reservadas y cualquiera las puede utilizar. Cada uno en su organización puede utilizar las mismas direcciones privadas.
- Los paquetes que contienen a estas direcciones no son enrutables a través de Internet.
- NAT ofrece grandes beneficios a empresas individuales y a la Internet. Permite la traducción de direcciones IP privadas en públicas.

DIRECCIONES PRIVADAS

Clase	Prefijo CIDR	Rango	
A	10.0.0.0/8	10.0.0.0 – 10.255.255.255	1 dirección clase A
B	172.16.0.0/12	172.16.0.0 - 172.31.255.255	16 direcciones clase B
C	192.168.0.0/16	192.168.0.0 - 192.168.255.255	256 direcciones clase C

- El resto de direcciones IPs son públicas.

DOMINIO PÚBLICO Y PRIVADO



COLAPSO DIRECCIONES IPv4

- Al actual protocolo IP, se le suele llamar **IPv4**
- IPv4 utiliza direcciones de red de 32 bits
- Durante un tiempo, IPv4 ofreció una estrategia de direccionamiento escalable. Pero a mediados de los 80, el exponencial crecimiento de Internet hizo que entrara en crisis el esquema de direcciones de IPv4:
  - Asignación de direcciones ineficiente y limitado
  - La escasez de direcciones de red clase B IPv4 no asignadas
  - El rápido aumento del tamaño de las tablas de enrutamiento de internet

COLAPSO DIRECCIONES IPV4

- IPv6 se presento como la solución definitiva que reemplazara IPv4. Su espacio de direccionamiento de 128 bits es prácticamente ilimitado
- Gran parte de los sistemas operativos modernos, así como los dispositivos de red, ya vienen preparados para la migración de IPv4 a IPv6.
- Está siendo implementada lentamente.
- Pero mucho antes de la definitiva implantación de IPv6, se comenzó a modificar con éxito el protocolo IPv4 para que pudiera sobrevivir al crecimiento exponencial de Internet.

COLAPSO DIRECCIONES IP

Las soluciones a corto plazo para la escasez de direcciones IPv4 han sido:

- Introducción de la máscara
- La división en subredes en 1985 (subnetting)
- La división en subredes de longitud variables en 1987 (VLSM)
- Superneting o agregación de rutas
- El enrutamiento interdominio sin clase en 1993 (CIDR)
- Las direcciones IP privadas
- La traducción de direcciones de red (NAT)

CONCEPTO DE MÁSCARA

- El sistema de clases de las direcciones IP (A, B, C y D) es muy rígido.
- Una mascara de subred es una secuencia de 32 bits que sirve para distinguir con facilidad que parte de una dirección codifica la subred y que parte el *host.*
- La máscara permite hacer variable el límite entre el NetID y el HostID de una dirección IP.
- Poniendo a 1 los bits más significativos de la máscara se identifica el NetID, y el resto de bits a 0 para señalizar el HostID.

CONCEPTO DE MÁSCARA

@IP = xxxxxxxx. xxxxxxxx. xxxxhhhh. hhhhhhhh

Mask = 11111111.11111111.11110000.00000000 "xxxx" identifica el NetID

"hhhh" identifica el HostID

Notaciones de la máscara:

1) Dirección IP/nº-bits-a-1-mascara

197.35.187.138 / 24

2) Notación con puntos:

197.35.187.138 - 255.255.255.0

CÁLCULO DIRECCIÓN DE RED Y DIRECCIÓN BROADCAST

¿Cuál es la dirección de red y de broadcast a la que pertenece un PC con IP 192.168.1.23/24 ?

192.168.1.23 = 11000000.10101000.00000001.00010111
/24 = 11111111 . 11111111 .11111111 .00000000
Bits de red Bits de host

Para red se realiza la operación lógica AND entre la IP y la máscara
192.168.1.0 = 11000000.10101000.00000001.00000000

Para broadcast se realiza la operación lógica OR entre la IP y la NOT de la máscara
192.168.1.255 = 11000000.10101000.00000001.11111111

PRÁCTICA 1. CÁLCULO DE REDES Y MÁSCARAS

1. Calcular la dirección de red y dirección de *broadcast* de las máquinas con las siguientes direcciones IP y máscaras de subred (si no se especifica, se utiliza la máscara por defecto):

- 18.120.16.250:
- 18.120.16.255 / 255.255.0.0:
- 155.4.220.39:
- 194.209.14.33:
- 190.33.109.133 / 255.255.255.0:

PRÁCTICA 1. CÁLCULO DE REDES Y MÁSCARAS

2. Encuentre la IP de red y de broadcast de los siguientes IP de host:

a) 11.2.3.78/8 b) 152.63.98.45/16

c) 222.222.253.56/24 d) 192.168.52.34/24

e) 129.23.56.65/16 f) 1.1.1.1/8

3. Determinar la máscara de una IP de clase A, B y C. Expresar las máscaras en todos los formatos posibles

PRÁCTICA 1. CÁLCULO DE REDES Y MÁSCARAS

4. Suponiendo que nuestro ordenador tiene la dirección IP 192.168.5.65 con máscara 255.255.255.0, indicar qué significan las siguientes direcciones especiales:

- 0.0.0.0:
- 192.168.67.0:
- 255.255.255.255:
- 192.130.10.255:
- 127.0.0.1:

PRÁCTICA 1. CÁLCULO DE REDES Y MÁSCARAS

5. Calcular la dirección de red y dirección de *broadcast* de las máquinas con las siguientes direcciones IP y máscaras de subred:

- 190.33.109.133 / 255.255.255.128:
- 192.168.20.25 / 255.255.255.240:
- 192.168.20.25 / 255.255.255.224:
- 192.168.20.25 / 255.255.255.192:
- 140.190.20.10 / 255.255.192.0:
- 140.190.130.10 / 255.255.192.0:
- 140.190.220.10 / 255.255.192.0:

PRÁCTICA 1. CÁLCULO DE REDES Y MÁSCARAS

6. Viendo las direcciones IP de los hosts públicos de una empresa observamos que todas están comprendidas entre 194.143.17.145 y 194.143.17.158, ¿Cuál es (probablemente) su dirección de red, broadcast y máscara?

INTRODUCCIÓN SUBNETTING

Siempre que necesitemos una dirección de red ¿deberemos estar condicionados por las redes de clase A, B o C?
Si necesito una red donde sólo hacen falta 20 IPs, ¿Cuántas direcciones IP malgastamos al coger una dirección de Clase C?
Malgastaríamos 2^8 - 2 – 20 = 234 IP's.

Con una de Clase B o A malgastaría muchas más.
¿Se puede hacer algo para evitar esto? Subnetting.

SUBNETTING (RFC 950)

Dirección base: 192.168.1.0/24

192.168.1.000xxxxx	Subred 0	192.168.1.0-192.168.1.31/27
192.168.1.001xxxxx	Subred 1	192.168.1.31-192.168.1.63/27
192.168.1.010xxxxx	Subred 2	192.168.1.64-192.168.1.95/27
192.168.1.011xxxxx	Subred 3	192.168.1.96-192.168.1.127/27
192.168.1.100xxxxx	Subred 4	192.168.1.128-192.168.1.159/27
192.168.1.101xxxxx	Subred 5	192.168.1.160-192.168.1.191/27
192.168.1.110xxxxx	Subred 6	192.168.1.192-192.168.1.223/27
192.168.1.111xxxxx	Subred 7	192.168.1.224-192.168.1.255/27

SUBNETTING (RFC 950)

- El subnetting apareció a medida que los sitios Web empezaron a desarrollarse.
- La división en clases (A, B, C) era demasiado rígida y no permitía aprovechar bien las direcciones en redes de área local.
- A mediados de los 80, no había máscaras en las tablas de encaminamiento, ya que todo se realizaba observando los primeros bits de la dirección IP (detección de las clases A, B y C).
- La máscara y el subnetting permitieron multiplicar de forma eficiente los distintos tipos de subredes

SUBNETTING (RFC 950)

- El subnetting permitió que las redes se pudieran dividir en redes más pequeñas llamadas subredes, haciendo variable las máscaras de las clases y aprovechando mejor el número de direcciones IP.
- Para crear subredes a partir de una dirección base (obtenida a través de un ISP), se "cogen bits prestados" de la porción de host para ser bits de subred (subnetID). Estos bits de subred han de ser los más significativos del hostID original (los que están más cercanos a los bits de red).

VENTAJAS DEL SUBNETTING

¿Por qué interesa dividir una red en subredes?

❖ Flexibilidad de direccionamiento en la red.

❖ Eficiencia. Permite que el administrador de la red brinde contención de broadcast. Los routers filtran los broadcast de nivel 2.

❖ Seguridad de bajo nivel en una LAN. El acceso a las subredes se realiza sólo a través de routers. Éstos permiten controlar mejor el tráfico (ACL's).

❖ Fuente de ingresos: La división en subredes crea una fuente de ingresos a través del alquiler o venta de direcciones IP que no se utilizaban.

EJEMPLO SUBNETTING

IANA asigna la dirección 147.83.0.0 (clase B) a un ISP. Esta es la dirección de partida para hacer el subnetting la cual llamaremos dirección base. Calcularemos las subredes /20 que se pueden crear y el nº de host que se pueden conectar a cada subred.

147.83.XXXXXXXX.XXXXXXXX → 147.83.0.0 /16

Red	Subred	Host			
147.83.0	0 0 0	XXXX.XXXXXXXX	→	147.83.0.0 /20	→ Nº hosts = 4094
147.83.0	0 0 1	XXXX.XXXXXXXX	→	147.83.16.0 /20	→ Nº hosts = 4094
147.83.0	0 1 0	XXXX.XXXXXXXX	→	147.83.32.0 /20	→ Nº hosts = 4094
147.83.0	0 1 1	XXXX.XXXXXXXX	→	147.83.48.0 /20	→ Nº hosts = 4094
. .. .	. . . . .	. . . . .			
147.83.1	1 1 1	XXXX.XXXXXXXX	→	147.83.240.0 /20	→ Nº hosts = 4094

CONCLUSIONES SUBNETTING

- Se reasignan bits del campo de host original al campo de subred de la dirección IP.
- La máscara de subred determinará la nueva subred creada = bits red original + bits subred.
- La selección del número de bits de host a utilizar dependerá del número máximo de hosts por subred y del número de subredes necesarias.
- Nº subredes = $2^{\text{nº bits subred}}$
- Nº hosts/subred = $2^{\text{nº bits restantes de host}} - 2$

CONCLUSIONES SUBNETTING

- Se pueden utilizar todos los bits de host para crear subredes, excepto los dos últimos bits menos significativos.
  - /29 → Nº de hosts = 2^3 -2 = 6 IP's
  - /30 → Nº de hosts = 2^2 -2 = 2 IP's, justo para configurar un enlace serie.
  - /31 → Nº de hosts = 2^1 - 2 = 0 IP's,
- Al hacer subnetting, perdemos 2 direcciones por subred, que no son asignables a hosts, pero nos ajustamos mejor a las necesidades de cada red

EJERCICIOS RESUELTOS DE SUBNETTING

1) Una compañía utiliza una @ de red 192.168.4.0 y una máscara 255.255.255.224 para la creación de subredes. ¿Cuál es la mayor cantidad de hosts utilizables para cada subred?

255.255.255.224 = 255.255.255.11100000

→ Nº Hosts por subred = 2^5 -2 = 30

2) Una empresa posee una @ de red de clase C y necesita crear 5 subredes, cada una de las cuales debe poder albergar por lo menos 20 hosts. Indica el subnetting.

20 hosts necesitan → nº bits = 5 bits para la parte de host. Como la dirección es de clase C (24 bits NetID), nos quedan por tanto 3 bits para subredes.

La máscara tendrá 27 bits a 1 → 255.255.255.224

EJERCICIOS RESUELTOS DE SUBNETTING

3) ¿Cuántas subredes utilizables de clase C se crean a partir de una máscara de subred 255.255.255.224?

255.255.255.224 = 255.255.255.11100000

→ nº subredes = 2^3= 8 subredes

4) ¿Cuáles son las @ de red y de broadcast para una dirección de host 198.101.6.55/28?

198.101.6.55/28 = 198.101.6.00110111

@ red = 198.101.6.00110000 = 198.101.6.48

@ broadcast = 198.101.6.00111111 = 198.101.6.63

EJERCICIOS RESUELTOS DE SUBNETTING

5) ¿Cuántas subredes utilizables se crean aplicando la máscara de subred 255.255.255.0 a una red clase B?

255.255.255.0

Red Subred → 8 bits → Nº subredes = 2^8 = 256

6) ¿Cuántos bits están disponibles para los hosts en las direcciones IP clase B que utilizan la máscara de subred por defecto? 16 bits para hostID.

7) Dado un host con @IP 172.32.65.13 y máscara por defecto ¿a qué red pertenece el host?

172.32.65.13 es de clase B (máscara 255.255.0.0), por lo tanto la @ de red es 172.32.0.0

EJERCICIOS RESUELTOS DE SUBNETTING

8) ¿Cuántos bits están disponibles en la máscara de la subred 255.255.248.0?

255.255.248.0 = 255.255.11111000.00000000

11 bits parte de host

9) Determina las @ IP que se pueden usar como @ de host en las subredes de las redes 200.100.50.0/28.

200.100.50.25	→	0001 01001	Si
200.100.50.80	→	0101 0000	No
200.100.50.100	→	0110 0100	Si
200.100.50.143	→	1000 1111	No
200.100.50.208	→	1101 0000	No
200.100.50.170	→	1010 1010	Si
200.100.50.90	→	0101 1010	Si
200.100.50.79	→	0100 1111	No

PRACTICA 2. EJERCICIOS DE SUBNETTING

1) Divide la dirección de clase C 195.100.5.0 para tener 50 máquinas en cada subred.

a) ¿Cuantos bits necesitamos para representar la dirección de máquinas?

b) ¿Cuantas subredes tendremos?

c) ¿Cuál es la mascara de subred?

d) ¿Cuál es la dirección de la primera máquina en cada subred?

PRACTICA 2. EJERCICIOS DE SUBNETTING

2) Se necesitan crear 14 subredes a partir de la dirección de clase C 210.30.4.0.

a) ¿Cuantos bits hemos de pedir prestados a la parte de host?

b) ¿Cual será la máscara de subred?

c) ¿Cual es la dirección de la primera subred?

d) ¿Cuantas direcciones para hosts tenemos en cada subred?

e) ¿Cuál es la dirección de subred y la de broadcast de la primera subred obtenida?

PRACTICA 2. EJERCICIOS DE SUBNETTING

3) Obtén 16 subredes a partir de la dirección base 200.200.128.0/22.

a) ¿Cuál es la nueva máscara de subred?

b) ¿Cuántos hosts puede disponer cada subred?

c) ¿Cuantos bits de host tenemos?

d) Determina el rango de direcciones en cada subred

PRACTICA 2. EJERCICIOS DE SUBNETTING

4) Divide la dirección de clase B 130.240.0.0 para conseguir tener 500 máquinas en cada subred.

a) ¿Cuantos bits necesitamos en la parte de host?

b) ¿Cuantas subredes tendremos?

c) ¿Cuál es la máscara de subred?

d) ¿Cuál es la dirección del primer host de la 5º subred?

PRACTICA 3. EJERCICIOS DE SUBNETTING

1. Si un nodo de una red tiene la dirección 172.16.45.14/30, ¿Cuál es la dirección de la subred a la cual pertenece ese nodo?
A. 172.16.45.0
B. 172.16.45.4
C. 172.16.45.8
D. 172.16.45.12
E. 172.16.45.18
F. 172.16.0.0

2. ¿Cuáles de las que se mencionan a continuación son 2 direcciones IP que pueden ser asignadas a nodos de la subred 192.168.15.19/28?
A. 192.168.15.17
B. 192.168.15.14
C. 192.168.15.29
D. 192.168.15.16
E. 192.168.15.31
F. Ninguna de las que se menciona.

PRACTICA 3. EJERCICIOS DE SUBNETTING

3. Usted se encuentra trabajando en una empresa a la que le ha sido asignada una dirección clase C y se necesita crear 10 subredes. Se le requiere que disponga de tantas direcciones de nodo en cada subred, como resulte posible. ¿Cuál de las siguientes es la máscara de subred que deberá utilizar?
A. 255.255.255.192
B. 255.255.255.224
C. 255.255.255.240
D. 255.255.255.248
E. 255.255.255.242
F. Ninguna de las que se menciona.

4. ¿Cuántas subredes y nodos disponibles por subred se obtienen si usted aplica una máscara /28 a la red clase C 210.10.2.0?
A. 30 subredes y 6 nodos por subred.
B. 6 subredes y 30 nodos por subred.
C. 8 subredes y 32 nodos por subred.
D. 32 subredes y 8 nodos por subred.
E. 14 subredes y 14 nodos por subred.
F. Ninguna de las anteriores.

PRACTICA 3. EJERCICIOS DE SUBNETTING

5. ¿Cuál es la dirección de subred que corresponde al nodo 172.16.210.0/22?
A. 172.16.42.0
B. 172.16.107.0
C. 172.16.208.0
D. 172.16.252.0
E. 172.16.254.0
F. Ninguna de las anteriores.

6. ¿Cuál es la dirección de subred que corresponde a la dirección IP 201.100.5.68/28?
A. 201.100.5.0
B. 201.100.5.32
C. 201.100.5.64
D. 201.100.5.65
E. 201.100.5.31
F. 201.100.5.80

PRACTICA 3. EJERCICIOS DE SUBNETTING

7. Considere las siguientes direcciones IP expresadas en binario:
A. 01100100.00001010.11101011.00100111
B. 10101100.00010010.10011110.00001111
C. 11000000.10100111.10110010.01000101
¿Cuáles de las siguientes afirmaciones son correctas respecto de estas 3 direcciones? (Elija 3)
A. La dirección C es una dirección pública clase C.
B. La dirección C es una dirección privada clase C.
C. La dirección B es una dirección pública clase B.
D. La dirección A es una dirección pública clase A.
E. La dirección B es una dirección privada clase B.
F. La dirección A es una dirección privada clase A.

8. ¿Cuál es el rango, expresado en nomenclatura binaria, del primer octeto de direcciones IP clase B?
A. 00001111 - 10001111
B. 00000011 - 10011111
C. 10000000 - 10111111
D. 11000000 - 11011111
E. 11100000 - 11101111
F. Ninguna de las anteriores.

PRACTICA 3. EJERCICIOS DE SUBNETTING

9. Su empresa tiene asignada la dirección clase B 172.12.0.0. De acuerdo a las necesidades planteadas, esta red debería ser dividida en subredes que soporten un máximo de 459 nodos por subred, procurando mantener en su máximo el número de subredes disponibles ¿Cuál es la máscara que deberá utilizar?

A. 255.255.0.0
B. 255.255.128.0
C. 255.255.224.0
D. 255.255.254.0
E. 255.255.248.0
F. 255.255.192.0

10. Escriba TODAS las redes que se generan cuando se hace un subnetting /28 a partir de una red de clase C 192.168.1.0/24:

Red0.IP_Network:
Red0.IP_Broadcast:
Red1.IP_Network:
Red1.IP_Broadcast:
Red2.IP_Network:
Red2.IP_Broadcast:
Red3.IP_Network:
Red3.IP_Broadcast:
(Continuar con las redes que faltan)

PRACTICA 3. EJERCICIOS DE SUBNETTING

11. ¿Cuál es la dirección de broadcast que corresponde a la IP 10.254.255.19 / 255.255.255.248?
A. 10.254.255.23
B. 10.254.255.24
C. 10.254.255.255
D. 10.255.255.255

12. ¿Cuál es la dirección de broadcast de la dirección de subred 172.16.99.99 / 255.255.192.0?
A. 172.16.99.255
B. 172.16.127.255
C. 172.16.255.255
D. 172.16.64.127

13. Si usted deseara tener 12 subredes con un ID de red Clase C, ¿qué máscara de subred debería utilizar?
A. 255.255.255.252
B. 255.255.255.248
C. 255.255.255.240
D. 255.255.255.255

PRACTICA 3. EJERCICIOS DE SUBNETTING

14. ¿Cuál es el número máximo de subredes que pueden ser asignadas a una red, cuando se utiliza la red de clase B 172.16.0.0 y la máscara de subred 255.255.240.0?
A. 16
B. 32
C. 30
D. 14
E. La máscara de subred es inválida para esa dirección de red.

15. Ud. ha dividido en subredes la red de clase C 213.105.72.0, utilizando una máscara de subred /28. ¿Cuántas subredes utilizables y direcciones de nodo utilizables por subred obtiene de esta manera?
A. 62 redes y 2 nodos.
B. 6 redes y 30 nodos.
C. 8 redes y 32 nodos.
D. 16 redes y 16 nodos.
E. 14 redes y 14 nodos.

VLSM

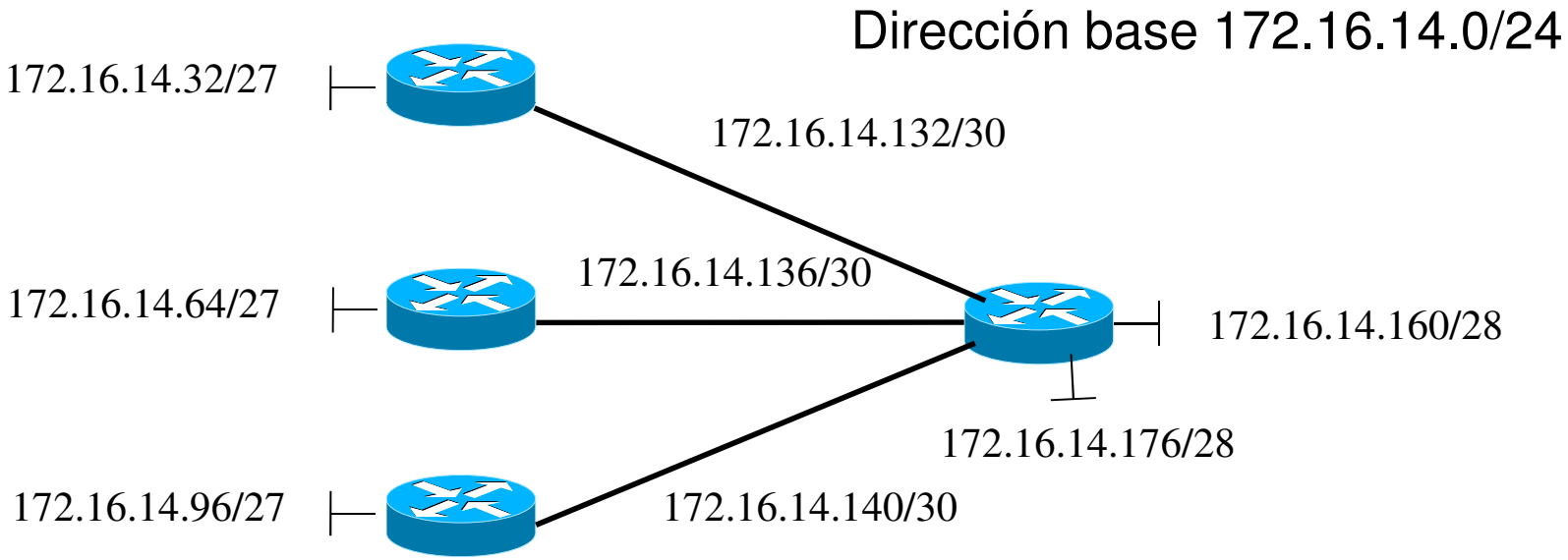
- La máscara de subred de longitud variable (VLSM) se utiliza para crear esquemas de direccionamiento eficientes y escalables.
- VLSM se puede usar para dividir en subredes una dirección que ya está dividida en subredes.
- La utilización de VLSM permite a una organización (AS) tener subredes de diferentes tamaños, es decir, utilizar más de una máscara de subred dentro del mismo espacio de direccionamiento de red.

VLSM

- VLSM permite adoptar un esquema de direccionamiento en función de las necesidades de la red. Con VLSM, un administrador de red puede usar una máscara larga en las redes con pocos hosts, y una máscara corta en las subredes con muchos hosts.
- Máscaras de 30 bits → Conexiones seriales, con sólo dos direcciones de host válidas
- Máscaras de 22-24 bits → Para 500-1000 usuarios.

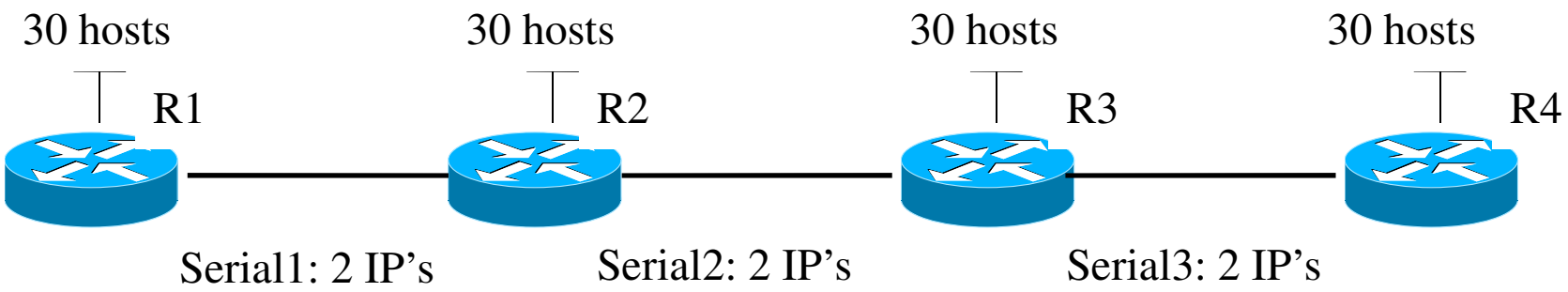
VLSM

- Una subred /27 sin utilizar se subdivide en tres subredes /30.
- Una subred /27 se divide en dos subredes /28



EJEMPLO PRACTICO VLSM

Partiendo de la @ base 192.168.187.0/24, realizar un esquema de direccionamiento IP que implique un mayor aprovechamiento de direcciones, evitando el desperdicio en los enlaces punto a punto y ajustándose a las necesidades de la red.



EJEMPLO PRÁCTICO VLSM

Primera aproximación: Subnetting

30 hosts se pueden codificar con 5 bits

→ 2^5 -2 = 30 hosts

192.168.187.XXXXXXXX

192.168.187. 000 XXXXX	→ 192.168.187.0/27	→ Ethernet E1
192.168.187. 001 XXXXX	→ 192.168.187.32/27	→ Ethernet E2
192.168.187. 010 XXXXX	→ 192.168.187.64/27	→ Ethernet E3
192.168.187. 011 XXXXX	→ 192.168.187.96/27	→ Ethernet E4
192.168.187. 100 XXXXX	→ 192.168.187.128/27	→ Serial 1
192.168.187. 101 XXXXX	→ 192.168.187.160/27	→ Serial 2
192.168.187. 110 XXXXX	→ 192.168.187.192/27	→ Serial 3
192.168.187. 111 XXXXX	→ 192.168.187.224/27	→ Libre

EJEMPLO PRÁCTICO VLSM

Primera aproximación: Subnetting

❖Direcciones libres = 30 @ IP. Hemos agotado todas las direcciones disponibles y no habrá más espacio para el crecimiento.

❖ Direcciones desperdiciadas en enlaces punto a punto = 28 x 3 = 84 @ IP. Este direccionamiento implica un desperdicio de 1/3 del espacio de direccionamiento potencial. Ha sido ocasionado por direccionar 3 redes punto a punto con subredes /27.

EJEMPLO PRÁCTICO VLSM

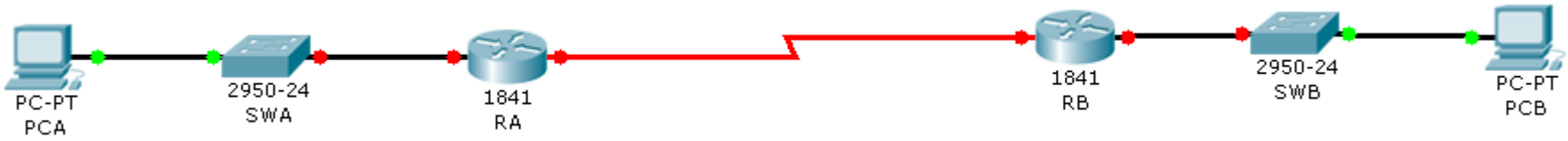
Segunda aproximación: VLSM

Asignaremos subredes /30 a las conexiones punto a punto. Dividiremos una subred que antes asignábamos enteramente a un enlace WAN

192.168.187. XXXXXXXX
192.168.187. 000 XXXXX → 192.168.187.0/27 → Ethernet E1
192.168.187. 001 XXXXX → 192.168.187.32/27 → Ethernet E2
192.168.187. 010 XXXXX → 192.168.187.64/27 → Ethernet E3
192.168.187. 011 XXXXX → 192.168.187.96/27 → Ethernet E4
192.168.187. 100 000 XX → 192.168.187.128/30 → Serial 1
192.168.187. 100 001 XX → 192.168.187.132/30 → Serial 2
192.168.187. 100 010 XX → 192.168.187.136/30 → Serial 3
192.168.187. 100 . . . . . → Podemos obtener direcciones IP para 8 enlaces series
192.168.187. 101 XXXXX → 192.168.187.192/27 → Libre
192.168.187. 110 XXXXX → 192.168.187.192/27 → Libre
192.168.187. 111 XXXXX → 192.168.187.224/27 → Llibre.

PRACTICA 4. EJERCICIOS DE VLSM

1) ¿Cuántas redes a nivel IP hay?



Si en la red Ethernet A hacen falta 20 IP's, en la red Ethernet B hacen falta 40 IP's, y en la red WAN hacen falta 2 IP's.

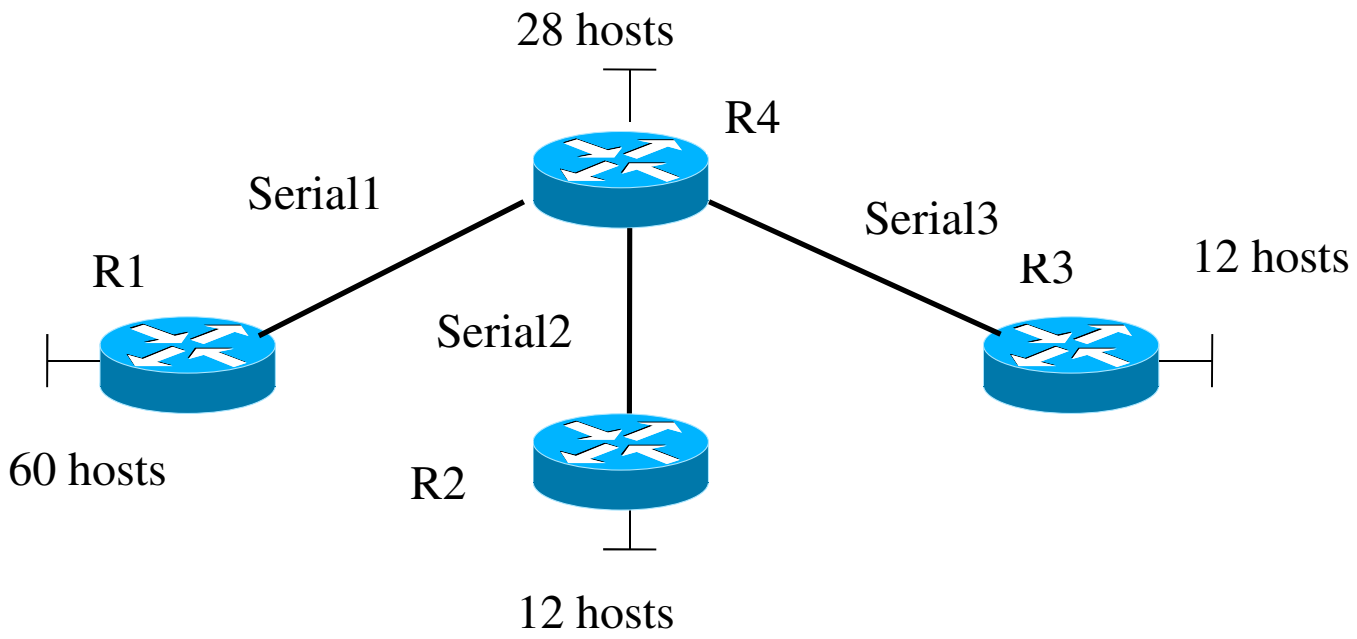
a) ¿Qué redes y máscara elegirías según has visto estos días en clase?

b) Calcula la cantidad de IPs que no se usarían

PRACTICA 4. EJERCICIOS DE VLSM

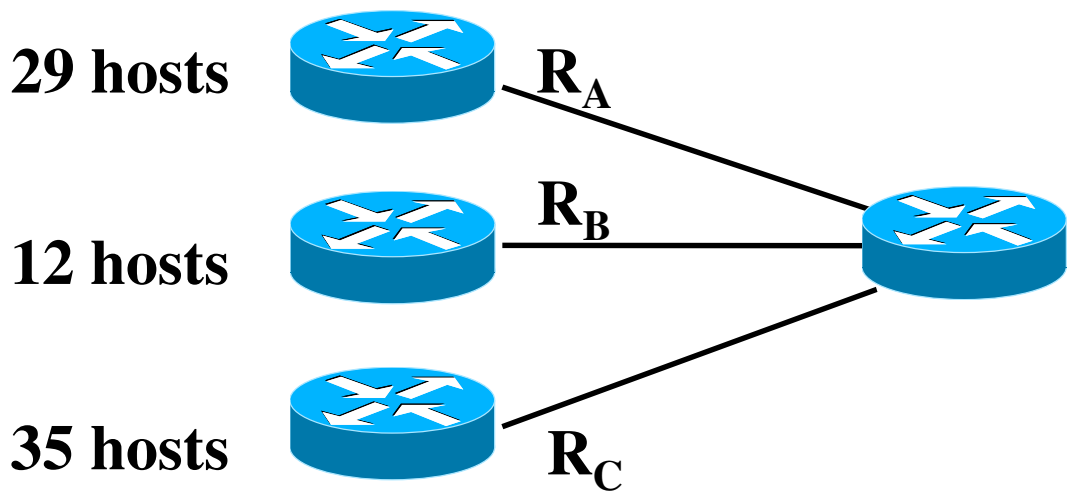
2) Realiza un esquema de direccionamiento optimizado que desperdicie pocas direcciones IP.

Dirección base: 192.168.10.0/24



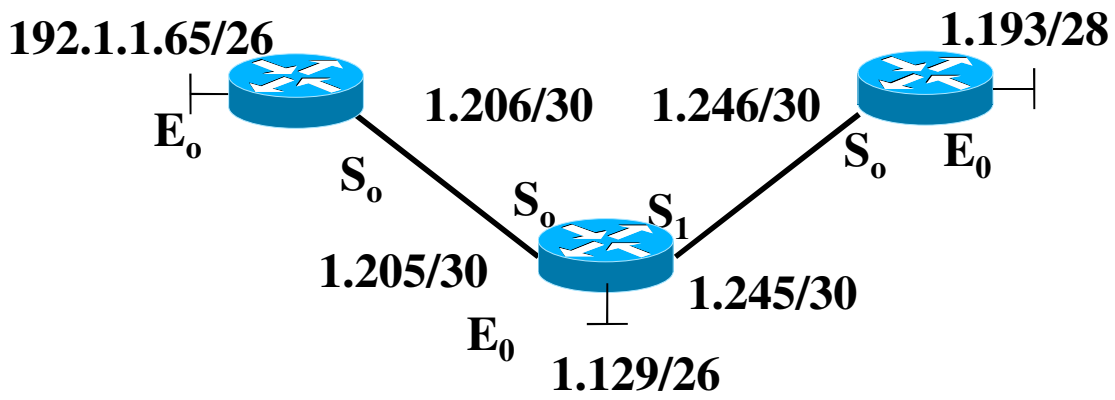
PRACTICA 4. EJERCICIOS DE VLSM

3) Máscara de bits que se debe de utilizar en las 3 redes para suministrar con eficiencia la cantidad de @host y desperdiciar la menor cantidad de direcciones posibles



PRACTICA 4. EJERCICIOS DE VLSM

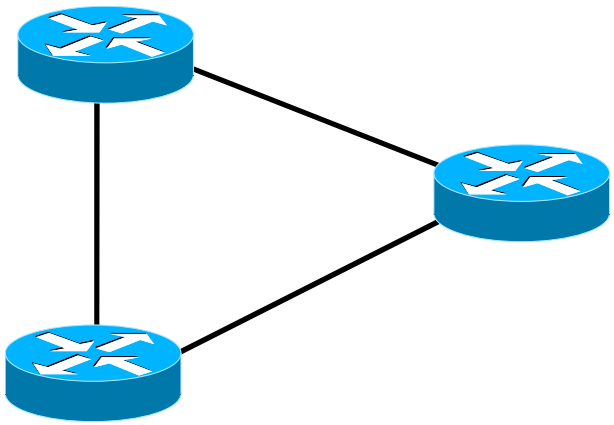
4) ¿Cuál es el problema de direccionamiento que se usa en la topología?



A) La dirección asignada a la interfaz E0 de R1 es una @ broadcast.
B) La subred configurada en el enlace serial entre R1 y R2 se superpone con la subred E0 de R3.
C) La subred asignada a la interfaz S0 de R1 está en una subred distinta que la @ de S0 de R2.
D) La subred asignada a la interfaz E0 de R2 se superpone con la subred asignada a E0 de R3

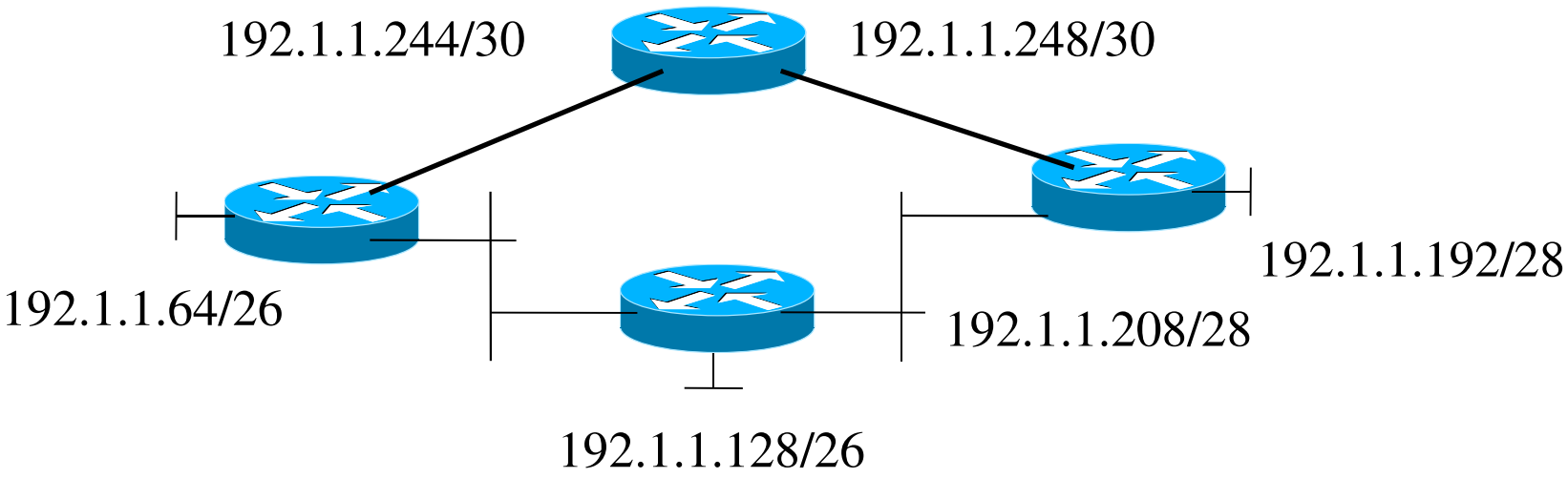
PRACTICA 4. EJERCICIOS DE VLSM

5) Se piden prestados 3 bits de la porción de host de una dirección de clase C ¿Cuántas direcciones de host válidas no se utilizarán en los 3 enlaces punto a punto si no se usa VLSM?



PRACTICA 4. EJERCICIOS DE VLSM

6) Indicar qué dirección de subred se puede utilizar para suministrar una cantidad máxima de 14 direcciones utilizables desperdiciando la menor cantidad de direcciones posibles



M01 Sistemes Informàtics

CFGS Desenvolupament d’Aplicacions Multiplataforma
UF1. Instal·lació, configuració i explotació del sistema informàtic.

NF2.1

Configuració i explotació de sistemes informàtics

Administració d’usuaris i grups en Linux

Administració Linux

Linux és un sistema operatiu multiusuari perquè una de les seves característiques principals és la capacitat de gestionar diferents usuaris en un mateix sistema.

Té també la capacitat de donar suport a diferents usuaris que treballin de manera simultània al mateix equip, localment o a través de la xarxa.

Cadascun d’aquests usuaris del sistema té assignat un directori personal, dins del qual té tots els permisos possibles (lectura, escriptura, execució). Així doncs, cada usuari pot tenir els seus documents, fitxers descarregats, preferències i aplicacions, sense que hi hagi cap interferència entre ells. De fet, un usuari no pot ni llegir ni modificar els arxius propis d’altres usuaris

Administració Linux

En les versions modernes de Linux ja creem per defecte un usuari durant la instal·lació. Aquest usuari no té drets d'administració, és un usuari que pot utilitzar el sistema però no administrar-lo (és un perfil d'usuari predeterminat).

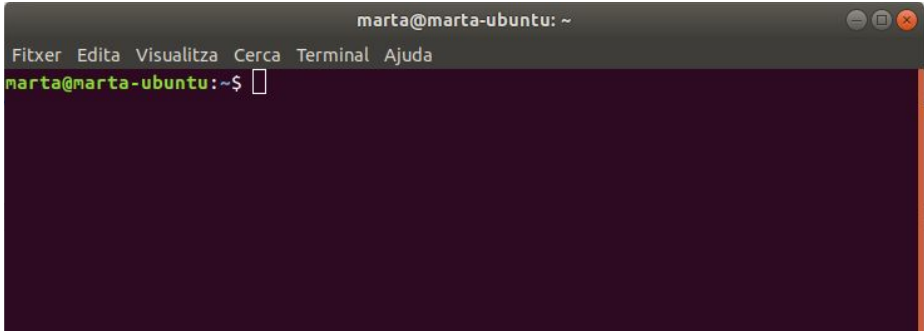
En el procés d'instal·lació també haurem d'assignar la contrasenya d'administrador (root).

L'usuari una vegada instal·lat el sistema podrà realitzar operacions a nivell d'usuari, però si necessita fer accions d'una complexitat superior requerirà l'administrador del sistema. Per a fer-ho necessitarem deixar el ratolí i obrir la **Terminal o Consola**.

La Terminal

Característiques principals:

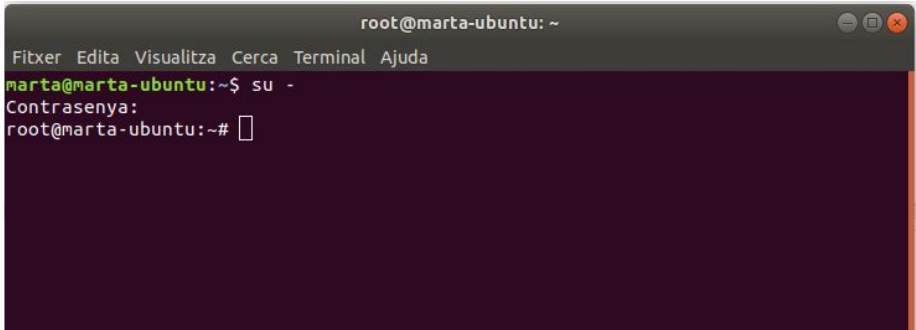
- Accés des del menú d'aplicacions → Utilitats o Alt + Cntrl + T
- Ens dóna control del sistema
- Té un **indicador d'ordres** (command prompt) que serveix per indicar-nos quan el sistema està preparat per rebre una nova ordre. L'indicador mostra el nom d'entrada de l'usuari i el directori de treball actual. El directori personal de l'usuari es representa mitjançant el caràcter ˜.
- El format serà: [usuari@host dir] on **usuari** serà el nom d'accés, **host** el nom de la màquina i i **dir** serà una indicació de la nostra ubicació actual en el sistema d'arxius.



La Terminal

Característiques principals:

- Si l’indicador d’ordres es visualitza en pantalla amb el caràcter $ seguit del cursor per escriure aleshores vol dir que l’usuari encara no té privilegis d’administració
- En moltes distribucions, per iniciar una sessió com a usuari administrador hem d’escriure la següent ordre: usuari@localhost ~ $ su
- Ens demanaran la contrasenya
- Moltes de les tasques d’administració en sistemes Linux requereixen la modificació de fitxers de configuració.
- Normalment aquests arxius es troben dins del directori /etc.



bash vs dash

En els entorns Linux, l’intèrpret d’ordres més utilitzat actualment és bash (bourne again shell), tot i que s’està començant a estendre la utilització de l’intèrpret dash.

Tipus d’ordres

Diccionari d’ordres

Ordres bàsiques

ls	Mostra una llista d’arxius en el directori de treball actual
cd directori	Canvia al directori especificat
passwd	Canvia la contrasenya de l’usuari actual
file nomarxiu	Mostra el tipus d’arxiu de l’arxiu nomarxiu
pwd	Mostra el directori de treball actual
exit o logout	Tanca la sessió actual
info ordre	Mostra les pàgines d’informació referents a l’ordre
man ordre	Mostra les pàgines del manual referents a l’ordre

Opcions de les comandes

Les opcions indiquen a una ordre que tingui un comportament diferent de l'habitual. Afegir una opció a una ordre és escriure-hi al darrera una lletra normalment precedida per un guió (-)

Per exemple: ls -a. L'opció -a indica a ls que mostri tots els arxius, fins i tot els ocults.

Les ordres també permeten que s'indiqui sobre quins paràmetres s'ha d'actuar. Un exemple seria ls /etc que mostraria tots els arxius de la carpeta /etc.
Algunes ordres requereixen paràmetres obligatòriament, i per a altres els paràmetres són opcionals.
Quan un paràmetre és opcional, s'acostuma a escriure entre claudàtors en la descripció de la sintaxi de l'ordre (els claudàtors no s'han d'introduir en la línia de comandes).

Opcions de les comandes

ls --help	Mostra l’ajuda breu de l’ordre ls
man ls	Mostra el manual de l’odre ls

Comprovant l’ajuda en línia d’una ordre, podrem saber si l’ordre accepta opcions i paràmetres i quins d’aquests són vàlids.

Usuaris a Linux

Els usuaris a Unix/linux s’identifiquen per un número únic d’usuari **User ID o UID**.

Cada usuari forma part d’un grup principal d’usuari, identificat també per un número únic de grup **Group ID o GID**.

Tot i això, l’usuari pot formar part d’altres grups apart del principal.

Usuaris a Linux. Usuari root

Usuari root:

- També es pot dir superusuari o administrador
- El seu UID (User ID) és 0
- És el compte d’usuari amb privilegis sobre tot el sistema
- Té accés total a tots els arxius i directoris amb independència de propietari i permisos
- Controla l’administració de comptes d’usuari
- Pot executar tasques de manteniment del sistema
- Pot instal·lar software al sistema
- Pot modificar o reconfigurar el Kernel, controladors, etc.

Usuaris a Linux. Usuaris del sistema

Usuaris del sistema:

- Exemples: bin, daemon, adm, lp, sync, shutdown, mail, operator, squid, apache, etc.
- No tenen tots els privilegis de l'usuari root però depenent del compte assumeixen privilegis del root.
- No tenen contrasenyes ja que són comptes que no estan definits per iniciar sessions. Per això també se les pot anomenar com a comptes de "no inici de sessió" (nologin)
- Es creen, generalment, automàticament en el moment de la instal·lació de Linux o de l'aplicació.
- Generalment s'assignen UID entre l'1 i el 100 (definits a /etc/login.defs)

Usuaris a Linux. Usuaris individuals

Usuaris individuals:

- Cada usuari disposa d’un directori de treball, ubicat generalment a /home.
- Cada usuari pot personalitzar el seu entorn de treball.
- Tenen només privilegis complerts en el seu directori de treball.
- Per seguretat, sempre és millor treballar com un usuari individual en comptes de root. Quan es requereixi fer ús de comandes específiques del root podrem utilitzar la comanda *su*.
- Generalment s’assignen UID superiors a 500.

Arxius. /etc/passwd

Tots els comptes d'usuari es troben definits a l'arxiu de configuració **'passwd'** dins del directori /etc. Aquest arxiu és de text (tipus ASCII), es crea en el moment de la instal·lació amb l'usuari root, els usuaris del sistema i els comptes que s'hagin indicat en el moment de la instal·lació.

L'arxiu /etc/passwd conté una línia per a cada usuari similar a les següents:

root:x:0:0:root:/root:/bin/bash
minion:x:501:500:Minion Banana:/home/minion:/bin/bash

Els camps estan separats per **dos punts (:)**

Analitzem cada camp →

Camp 1	És el nom de l'usuari, l'identificador d'inici de sessió. Ha de ser únic.
Camp 2	La 'x' indica la contrasenya encriptada de l'usuari, també indica que s'està utilitzant l'arxiu /etc/shadow. Si no es fa ús d'aquest arxiu veuríem quelcom semblant a això: 'uXCc12r5gt78uuu6R'.
Camp 3	Número identificador de l'usuari (UID). Ha de ser únic. 0 per al root | 1 al 100 comptes de sistema | usuaris individuals a partir del 101 o en les distribucions més recents a partir del 500.
Camp 4	Identificador del grup (GID). És el grup principal de l'usuari però pot aparèixer en altres, això ho configurarem al /etc/groups.
Camp 5	Comentaris o el nom complet de l'usuari.
Camp 6	Directori de treball (Home) on es situa l'usuari després de l'inici de sessió.
Camp 7	Shell que utilitzarà l'usuari de manera predeterminada.

Arxius. /etc/shadow

Anteriorment (en sistemes Unix) les contrasenyes xifrades s'emmagatzemàven en l'arxiu /etc/passwd.
El problema és que 'passwd' és un arxiu que pot ser llegit per a qualsevol usuari del sistema encara que només pugui ser modificat pel root.

Així doncs, amb qualsevol equip potent, avui en dia, i amb un bon programa per desxifrar contrasenyes es podrien "crackejar" contrasenyes dèbils. En aquest sentit és important donar una data de caducitat a les contrasenyes i regenerar-les sovint.

L'arxiu 'shadow' soluciona aquest problema ja que només hi pot accedir l'usuari root. Es pot entendre com una extensió del 'passwd' ja que no només emmagatzema una contrasenya encripada sinó que té altres camps de control sobre la contrasenya.

*Hi ha dos algorismes d'encriptació de les contrasenyes: MD5 i DES. MD5 és un mètode fort d'encriptació.

Arxius. /etc/shadow

L’arxiu ‘shadow’ només hauria de tenir permisos de lectura i escriptura per a l’usuari administrador, la resta d’usuaris no hauria de tenir cap permís.

*Evitar contrasenyes molt òbvies com ara dates de naixement, 123456, qwerty però evitar també 123k$%”!asdfMDasd123 ja que són impossibles de recordar.
Alguns consells:

- Incloure signes de puntuació
- Barrejar majúscules i minúscules
- Llargada mínima de 8 caràcters
- Substituir caràcters per altres d’aspectes semblants. Ex: $ per S
- No donar mai la contrasenya a ningú ni deixar-la anotada
- No enviar-la per correu electrònic
- Canviar la contrasenya sovint.

Arxius. /etc/shadow

L’arxiu /etc/shadow conté una línia per a cada usuari, similar a les següents:

root:ghy675gjuXCc12r5gt78uuu6R:10568:0:99999:7:7:-1::

minion:rfgf886DG778sDFFDRRu78asd:10568:0:-1:9:-1:-1::

Els camps estan separats per **dos punts (:)**
Analitzem cada camp →

Camp 1	Nom del compte de l’usuari
Camp 2	Contrasenya xifrada o encriptada. Un ‘*’ indica que és un compte que no té login ‘nologin’
Camp 3	El nombre de dies des de l’1 de gener de 1970 fins al darrer canvi de contrasenya. Aquesta data és coneguda en el món de l’UNIX com a epoch.
Camp 4	Número de dies mínim que han de passar perquè la contrasenya es pugui tornar a canviar
Camp 5	Número de dies màxim que han de passar per canviar la contrasenya. El ‘-1’ significa que no s’ha de canviar. A partir d’aquesta dada es pot obtenir la data d’expiració de la contrasenya.
Camp 6	Número de dies abans de l’expiració de la contrasenya en que s’avisarà a l’usuari a l’inici de la sessió
Camp 7	Número de dies després de l’expiració en que la contrasenya s’inhabilitarà si no s’ha canviat.
Camp 8	Data de caducitat del compte. S’expressa en dies transcorreguts des de l’1 de gener del 1970 (epoch)
Camp 9	Reservat

Arxius. /etc/group

Aquest arxiu guarda la relació dels grups als que formen part els usuaris del sistema. Conté una línia per a cada usuari amb tres o quatre camps per usuari:

Camp 1	Indica l'usuari
Camp 2	La 'x' indica la contrasenya del grup, que no existeix, si n'hi hagués, mostraria un 'hash' encriptat
Camp 3	És el Group ID (GID) o identificador del grup.
Camp 4	És opcional i indica la llista de grups secundaris als que forma part l'usuari

```
root:x:0:root
ana:x:501:
minion:x:502:ventas,supervisores,produccion
```

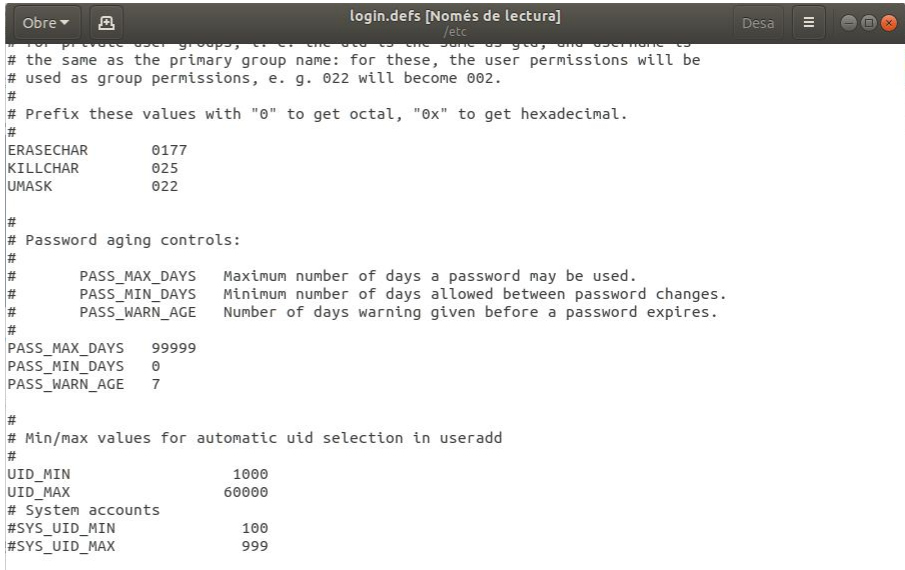
Actualment en crear un usuari amb useradd es crea també automàticament el seu grup principal de treball GID amb el mateix nom que el d'usuari. És a dir si s'afegeix l'usuari 'minion' també es crearà el grup 'minion'.

Hi ha un fitxer anomenat '/etc/gshadow' que conté informació encriptada sobre els grups d'usuari

Arxius. /etc/login.defs

A l’arxiu de configuració /etc/login.defs estan definides les variables que controlen els aspectes de la creació d’usuaris i dels camps de shadow utilitzades per defecte. Algunes variables interessants d’estudiar són:

- PASS_MAX_DAYS: número màxim de dies de validesa d’una contrasenya
- PASS_MIN_LEN: número mínim de caràcters en una contrasenya
- UID_MIN: valor mínim per al UID pels usuaris individuals quan s’afegeix un nou usuari
- CREATE_HOME: directori per defecte que se li assigna a un usuari




```
# the same as the primary group name: for these, the user permissions will be
# used as group permissions, e. g. 022 will become 002.
#
# Prefix these values with "0" to get octal, "0x" to get hexadecimal.
#
ERASECHAR       0177
KILLCHAR        025
UMASK           022

#
# Password aging controls:
#
#       PASS_MAX_DAYS   Maximum number of days a password may be used.
#       PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#       PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7

#
# Min/max values for automatic uid selection in useradd
#
UID_MIN                  1000
UID_MAX                 60000
# System accounts
#SYS_UID_MIN              100
#SYS_UID_MAX              999
```

Afegir un usuari

Per tal d'afegir un usuari utilitzarem les comandes **useradd o adduser**. Les opcions més comuns i importants són les que trobareu en la següent taula →

-c	Afegeix un comentari al moment de crear l'usuari (camp 5 'passwd')
-d	Directori de treball o home de l'usuari (camp 6 'passwd')
-e	Data d'expiració del compte. Format AAAA-MM-DD (camp 8 'shadow')
-g	Número de grup principal de l'usuari (GID) (camp 4 'passwd')
-G	Altres grups als que pot formar part l'usuari (separat per comes)
-r	Crea un compte del sistema, el seu UID serà menor al definit en etc/login.defs en la variable UID_MIN, no es crearà el directori d'inici.
-s	Shell per defecte de l'usuari quan entra al sistema. Si no s'especifica serà bash
-u	UID de l'usuari, si no s'indica aquesta opció automàticament s'establirà el següent número disponible a partir de l'últim creat.

Afegir un usuari

Si fem useradd + “nom d’usuari” #> useradd minion

Es crearà l’usuari i un grup, així com les entrades corresponents als arxius /etc/passwd, /etc/shadow i /etc/group. També es crearà el directori d’inici o de treball: /home/minion i els arxius de configuració que van dins d’aquest directori.

Les dates d’expiració de contrasenya, per defecte, queden molt àmplies així que en principi no hi ha problema amb que el compte caduqui. Per tant ens quedarà afegir la contrasenya de l’usuari i algun comentari o identificador del compte.

#> useradd -d /usr/juan -s /bin/csh -u 800 -c "Juan Perez Hernandez" juan

Amb aquesta comanda estem canviant el directori d’inici, el shell per defecte i l’UID.

Modificar un usuari

Per tal de modificar o actualitzar un usuari ja existent farem ús de la comanda **usermod**. Les opcions més comuns són →

-c	Modifica el comentari de l'usuari (camp 5 'passwd')
-d	Modifica el directori de treball o home de l'usuari (camp 6 'passwd')
-e	Modifica la data d'expiració del compte. Format AAAA-MM-DD (camp 8 'shadow')
-g	Canvia el número de grup principal de l'usuari (GID) (camp 4 'passwd')
-G	Modificar altres grups als que pot formar part l'usuari (separat per comes)
-l	Canvia el login o el nom de l'usuari (camp 1 de 'passwd' o 'shadow')
-L	Bloqueja el compte d'usuari, no esborra res només el deshabilita
-s	Canvia el shell per defecte de l'usuari quan entra al sistema.
-u	Canvia el UID de l'usuari
-U	Desbloqueja un compte previament bloquejat amb l'opció -L

Modificar un usuari

Si volem canviar el nom de l’usuari de minion a minion.banana farem:

```
#> usermod -l minion.banana minion
```

En la majoria dels casos el nom del directori d’inici es canviarà de manera automàtica, si no fos així hauríem de fer:

```
#> usermod -d /home/minion.banana minion.banana
```

Altres canvis o modificacions que podem fer són:

```
#> usermod -c "supervisor de area" -s /bin/ksh -g 505 minion.banana
```

D’aquesta manera modifiquem el comentari del compte, el shell i el grup principal de l’usuari.

Si volem deshabilitar un usuari perquè ningú faci servir el seu compte farem:

```
#> usermod -L minion.banana
```

Eliminar un usuari

Si volem eliminar un usuari haurem de fer ús de la comanda **userdel** per eliminar tot el compte de l'usuari. Podem fer-ho de tres maneres diferents:

#> userdel minion.banana

Si no posem cap opció més eliminarem el compte d'usuari de 'passwd' i de 'shadow' però no eliminarem el seu directori de treball, d'aquesta manera s'eliminarà el compte però no la informació.

#> userdel -r minion.banana

Amb aquesta opció eliminarem el directori de treball amb els arxius que hi hagi dins. El compte no es podrà eliminar si l'usuari està loguejat al sistema en el moment d'executar la comanda.

#> userdel -f sergio

L'opció -f és igual que l'opció -r, elimina tot l'usuari, compte, directori i arxius però sense tenir en compte si l'usuari està actualment treballant al sistema. És una opció "més radical" i podria causar inestabilitat al sistema així que es fa servir en casos molt aïllats.

Modificar la contrasenya

Mitjançant la comanda passwd caldrà fer els següents passos:

```
#> passwd minion.banana
Changing password for user prueba.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
#>
```

L’usuari root és l’únic que pot realitzar aquesta tasca a qualsevol usuari. Els usuaris individuals poden canviar la seva contrasenya en qualsevol moment escrivint passwd (sense el seu nom d’usuari al darrera).
El sistema guardarà la contrasenya encriptada, en cap cas l’original.

L’administrador del sistema haurà de seguir una política de gestió de les contrasenyes, és a dir, establir les contrasenyes permeses i prohibides, la freqüència de canvi, la recuperació de contrasenyes...

Modificar la contrasenya

La comanda passwd té integrada la validació de contrasenyes comuns, curtes, de diccionari… així que si per exemple si intentem canviar la contrasenya i posar ‘qwerty’ el sistema ens mostrarà el següent error:

```
$> passwd
Changing password for user prueba.
New UNIX password:
BAD PASSWORD: it is based on a dictionary word
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
$>
```

En escriure ‘qwerty’ el sistema detecta que és una seqüència coneguda i ens mostra el missatge de "BAD PASSWORD: it is based on a dictionary word", tot i això, ens deixa continuar si la verifiquem. És a dir, el sistema ens avisa que la contrasenya és dèbil però permet establir-la si realment es desitja.

Afegir un grup

Per tal d'afegir un nou grup farem ús de la comanda **groupadd**
#groupadd nomgrup

Podrem modificar paràmetres de configuració amb l'ordre **groupmod** o eliminar-los mitjançant **groupdel.** El resultat el trobarem a /etc/group.

Un usuari pot pertànyer a més d'un grup. Això és especialment útil per gestionar l'accés d'usuaris de diferents grups a un mateix recurs.

També es pot canviar el grup al qual forma part un usuari després d'haver-lo creat. Per defecte es crea un nou grup d'usuaris per a cada usuari que afegim al sistema.

L'ordre **usermod** permet canviar el grup primari d'un usuari després d'haver-lo creat.

Monotoritzar usuaris

La monitorització de l’activitat de l’usuari és una tasca fonamental de l’administrador del sistema per comprovar com es fan servir els recursos del sistema.
Les ordres principals de monitorització de l’activitat dels usuaris són:

- -w: dóna informació sobre quins usuaris estan autenticats en el sistema, des de quin moment han iniciat la sessió i què estan fent. Cap usuari es pot amagar del superusuari. Podem utilitzar, com a paràmetre de l’ordre w, el nom d’un usuari concret per mostrar només què està fent aquell usuari.
- -ac podem veure el temps total de connexió d’un usuari al sistema. Si no està instal·lada per defecte, la podem instal·lar amb l’ordre #apt-get install acct. Aquesta ordre accedeix al fitxer /var/log/wtmp per obtenir la informació.
- -last: cerca en el fitxer /var/log/wtmp i fa una llista de tots els usuaris que han iniciat i aturat la sessió des de la seva creació
- -lastb: mostra tots els intents fallits d’autenticació o d’inici de sessió al sistema. És útil per determinar si un usuari legítim té problemes per accedir al sistema o si estan intentant hackejar comptes.

Resum de comandes

Existeixen altres comandes que s’utilitzen poc en l’administració d’usuaris però que permeten personalitzar més detalladament els usuaris.

adduser	Afegir usuari
chage	Permet canviar o establir paràmetres de les dates de control de la contrasenya
id	Mostra el UID i els grups als que forma part
gpasswd	Administra la contrasenya de grups (/etc/group i /etc/shadow)
groupadd	Afegeix grups al sistema (/etc/group)
groupdel	Elimina grups del sistema

Resum de comandes

groupmod	Modifica grups del sistema
groups	Mostra els grups als que forma part un usuari
pwconv	Estableix la protecció shadow (/etc/shadow) a l’arxiu /etc/passwd
pwunconv	Elimina la protecció shadow (/etc/shadow) a l’arxiu /etc/passwd
useradd	Afegeix un usuari al sistema
userdel	Elimina usuaris del sistema
usermod	Modifica usuaris

Resum d’arxius

Arxius d’administració i control d’usuaris:

/etc/group	Usuaris i els seus grups
/etc/gshadow	Contrasenyes encriptades dels grups
/etc/passwd	Usuaris del sistema
/etc/shadow	Contrasenyes encriptades i control de dates d’usuari del sistema

Noms especials per a directoris

- El caracter **~** representa $HOME, el meu directori.
- El directori actual es representa amb un **punt** (.)
- El directori superior es representa amb **punt punt** (..)
- El **camí absolut** (absolute path) és un camí que comença el el directori arrel (/).
- El **camí canònic** (canonical path) és el camí absolut més curt.
    - Existeixen camins absoluts que no són canònics:
        - /home/cesc/../cesc

Directoris: organització de la jerarquia

Documentació completa en: [Filesystem Hierarchy Standard]

/boot
/dev
    Dispositius com arxius. També alguns dispositius especials, com ara /dev/null, /dev/random ...
/proc
    Arxius especials representant informació del sistema i dels procesos.
/bin
    Arxius binaris (executables)
/sbin
    Arxius binaris per l'administració del sistema
/lib
/etc
    Arxius de configuració.
/home
    Directoris dels usuaris.
/root
    $HOME per a l'usuari root.
/tmp
    Arxius temporals
/lost+found
/usr
    arxius pels usuaris, binaris i read-only. Trobem tot un arbre:
    /usr/bin
    /usr/sbin
    /usr/lib

/usr/local

/var

arxius d’escriptura. Per exemple /var/log

Tipus d’arxiu

Amb l’ordre ls -l obtenim el llistat llarg, on s’indica el tipus d’arxiu amb la primera lletra de cada línia:

caracter	
-	Arxiu
d	Directori
l	Enllaç simbòlic
c	Dispositiu de caracters
b	Dispositiu de blocs
p	Pipe
s	Socket

Permisos

- S’estableixen permisos pel **propietari**, el **grup** principal del propietari i per a la **resta** d’usuaris. Per a cada usuari s’aplica només una de les tres categories.
- El permís consisteix en tres operacions bàsiques: (r) lectura, (w) escriptura i (x) execució. Apareixen 3 permisos indicats primer per al propietari, seguit per 3 permisos pel grup i 3 permisos per la resta d’usuaris. Quan no hi és el permís apareix un (-) guió, de forma que sempre tenim 9 caràcters (3 x 3) indicant els permisos.
    - En arxius:
        - r: Obrir i llegir l’arxiu.
        - w: Modificar o esborrar el contingut de l’arxiu.
        - x: Executar l’arxiu (binari o script)
    - En directoris:

        - r: Permet veure el contingut
        - w: Permet crear i esborrar fitxers i directoris dins del directori.
        - x: Permet posicionar-se dins i copiar arxius del directori o al directori.
- Existeix una notació numérica alternativa a la notació rwx. En binari, $r=2^2$, $w=2^1$, $x=2^0$; és a dir, r=4,w=2,x=1, i així tenim un dígit en octal (sempre menor 8). Podem fer servir un dígit octal per a l'usuari, un per al grup i un altre per a la resta, i els permisos queden representats per un número octal de 3 xifres.
    - Exemples:
        - 666 = rw-rw-rw-
        - 777 = rwxrwxrwx

Permisos especials per a arxius executables i scripts

- **setuid**: el procés s'executarà amb impersonant el propietari de l'arxiu executable.
    - Apareix com a **S** (majúscula) en el lloc on hauria d'haver (x) el permís d'execució del propietari.
    - Si també n'hi ha el permís d'execució apareix **s** (minúscula), que és el cas habitual, donat que és estrany tenir setuid però no tenir permís d'execució.
    - En mode numèric s'afegeix una quarta xifra, amb el valor 4.
        - 4777 = rw**s**rwxrwx
- **setgid**: Quan s'estableix en un **fitxer ordinari** indica que, quan s'executi, el procés s'executarà fent servir el grup principal del propietari de l'arxiu. Quan s'estableix en un **directori** indica que tots els arxius i directoris creats dins del directori es creen fent servir el seu grup com a grup propietari (però compte, per crear-los encara es requereix el permís d'escriptura). Es fa servir per a gestionar directoris compartits on els usuaris poden crear arxius.
    - Apareix com a **S** (majúscula) en el lloc on hauria d'haver (x) el permís d'execució del grup.
    - Si també n'hi ha el permís d'execució apareix **s** (minúscula), que és el cas habitual, donat que és estrany tenir setgid però no tenir permís d'execució.
    - En mode numèric s'afegeix una quarta xifra, amb el valor 2.
        - 2777 = rwxrw**s**rwx

Permisos especials per a directoris

- **sticky bit (restricted deletion flag)**: *Per a fitxers ordinaris està en desús; antigament indicava que l'arxiu quedava en memòria (o memòria swap) quan finalitzava l'execució, per tornar-lo a executar més ràpidament (per això el nom de sticky)*. **Per a directoris**, prohibeix eliminar o renombrar arxius; es fa servir per a directoris compartits on els usuaris poden crear arxius. Els fitxers només els podrà eliminar:
    - El propietari de l'arxiu.
    - El propietari de la carpeta.
    - L'usuari root.
  - Apareix com a **T** (majúscula) en el lloc on hauria d'haver (x) el permís d'execució per als altres.
  - Si també n'hi ha el permís d'execució apareix **t** (minúscula), que és el cas habitual, donat que és estrany tenir el ***restricted deletion flag*** però no tenir permís d'execució.
  - En mode numèric s'afegeix una quarta xifra, amb el valor 1.
    - 1777 = rwxrwxrw**t**

Links

Soft links

Un *soft-link* és un enllaç a l'autèntica ubicació del fitxer. Aquests enllaços poden ser-hi a diferents sistemes d'arxius (diferents dispositius). Esborrar el *soft-link* no implica esborrar el fitxer (excepte quan es demana explícitament).

Hard links

Un fitxer pot ser hi a més d'un lloc, amb diferents noms, però sempre dins del mateix sistema d'arxius (mateix dispositiu); podríem dir que són "*sinònims*" del nom del fitxer, indistinguibles entre ells, tots amb la mateixa categoria.

Arxius d'inici de sessió

globals

`/etc/profile`

Arxiu d'inici per a tots els usuaris en sessions interactives; compte, executa els scripts `/etc/profile.d/*.sh`, que seria el lloc on afegir el nostre codi.

`/etc/bash.bashrc`

Arxiu d'inici per a tots els usuaris en sessions interactives i no interactives; per als usuaris interactius es crida des-de `/etc/.profile`. El nom pot ser `/etc/bash.bashrc` o bé només `/etc/bashrc`.

`/etc/bash.bash_logout`

Arxiu de finalització de sessió interactiva per a tots els usuaris.

d'usuari

`~/.bash_profile`
`~/.bash_login`
`~/.profile`

Arxiu d'inici per a l'usuari en sessions interactives. S'executa només un dels tres arxius, comprovant-los en l'ordre que se esmenten: primer es comprova `~/.bash_profile`, i només quan no existeix es miren els altres dos, per ordre.

`~/.bashrc`

Arxiu d'inici per a l'usuari en sessions interactives i no interactives.

`~/.bash_logout`

Arxiu de finalització de sessió interactiva per a l'usuari.

Ordres (case sensitive)

ajut	
--help	Opció disponible a totes les ordres. De vegades també l'opció **-h**
help	Ajut únicament per a les ordres internes de bash.
man	Manual. Conté informació de totes les ordres, una mica més extensa que fent servir l'opció --help o -h en l'ordre.
whatis	Fa cerques per a una ordre en el manual; el mateix que: man -f

apropos	Fa cerques en el manual per a una paraula clau; el mateix que: man -k
info	Sistema d'ajuda de GNU. Obté la informació d'arxius en format *info*, però no totes les ordres ho tenen, i els arxius *info* poden no ser sincronitzats (tenir referències a ordres obsoletes, o referències absents). La informació, quan hi és, generalment és més complerta que l'obtinguda en les pàgines del manual.
/usr/share/doc	Directori de documentació.
on-line	
Documentació de Gentoo (FreeBSD), molt completa: https://www.gentoo.org/support/documentation/	
The Linux Command Line (William Shotts) (Creative Commons book)	

general		
uname -r	Versió del kernel del sistema	
lsb_release -a	Versió de la distribució	
lsb_release -v	Versió LSB que compleix (Linux Standard Base)	
uptime	Data/hora d'inici del sistema.	
clear	Esborra el terminal.	
date	Data i hora.	
echo	Còpia en la sortida l'argument.	
banner	Mostra un text en lletres grans.	
type	Indica com seria interpretat el paràmetre si fos una ordre (interna de bash, alias o execució d'un binari).	
	type -a echo	Mostra totes les ubicacions per a l'ordre echo
which	Busca el binari per a una ordre. Algorisme: recorre $PATH fins que troba el directori on és.	

	Compte! No indica quines ordres són internes de bash	
	which cd	no troba binari per a cd
	which echo	troba un binari, però en realitat s'executa com a ordre interna de bash; es pot comprovar amb: ***type echo***
whereis	Informa del directori on trobar una comanda: binari, pàgines del man(ual) i d'info i el codi font.	
	-l	Mostra els directoris en què busca.
time		
alias		
yes	Còpia en la sortida el text de l'argument de manera infinita. Retorna afirmativament de manera infinita (o negativament, si fem servir el valor "n" com a argument); serveix per automatitzar respostes.	

Variables i entorn	
echo $HOME	Directori de l'usuari
echo $PATH	Directoris per la búsqueda de fitxers binaris per l'execució d'ordres.
echo $PS1	Prompt de l'usuari.
printenv	Mostra les variables de l'entorn, cadascuna amb el seu nom. Podem indicar com argument la(les) variable(s) que volem que mostri, i mostrarà el valor, cadascuna en una línia.
printenv HOME PATH	
export / env	
unset	

getent

maquinari		
free	Memòria disponible (RAM)	
hwclock	Date i hora de les BIOS. El sistema quan arranca agafa aquesta data/hora, excepte quan fem servir un servei NTP per obtenir la data/hora de la xarxa. Per l'ordre **date**: la sincronització amb el hwclock és només en iniciar el sistema o manual, però el servei NTP sí que manté una sincronització automàtica.	
hwinfo	Informació del maquinari. Si no està instal·lat comprovar lshw.	
	--short	breu
	--cpu	cpu
	--disk	discs
lscpu	cpu	
lspci	pci	
lsusb	usb	
sudo dmidecode	Fa servir estructures i mètodes definits per a obtenir informació del maquinari fent servir directament codi del microprogramari (firmware). Estandards definits pel [Distributed Management Task Force](#) (DMTF)	
sudo hdparm	Modificacions en paràmetres de disc	
Addicionals (potser s'han d'instal·lar)		
sudo lshw	Alternativa similar a hwinfo	
Mode gràfic addicionals (potser s'han d'instal·lar)		
sudo hardinfo	recorda l'aplicació CPU-Z	
kinfocenter	visualització sencilla	
cpu-x	inspirat en CPU-Z (preferible hardinfo)	

disc		
df	disk free. Informa del punt de muntatge de les particions muntades.	
du	disk usage	
	-sh	Summarize, Human readable
	--time	data/hora modificació més recent dels arxius de la carpeta
lsblk	dispositius de bloc	
blkid	tipus i UUID de les particions	
parted	gestionar particions GPT (nou, UEFI)	
fdisk	gestionar particions MBR (antic, BIOS)	
fsck	corregir errors de disc. Escriu en /lost+found els arxius parcialment recuperats.	

Xarxa	
hostname	Nom de host.
hostnamectl	Nom de host (permet canviar-lo).
ip address	Adreces IP de les interfícies.
ip address add	Definir adreça IP per a una interfície manualment.
ip neigh	Neighbours; protocol ARP.
ip route	Rutes manuals
ss -t state connected	ports connectats (opció -t es refereix a ports tcp)
ping	Eco del protocol ICMP.

Usuaris		
whoami	Mostra el nom de l'usuari. Equivalent a **id -u**	
who	Mostra els usuaris que hi són en el sistema en aquest moment.	
users	Mostra els usuaris que hi són en el sistema en aquest moment; molt semblant al comando **who** bàsic, però sense opcions ni informació addicional.	
w	Mostra els usuaris que hi són en el sistema en aquest moment, i què estan fent. Molt semblant a **who**, però amb informació addicional.	
last		
id		
groups		
passwd	Canvia la contrasenya.	
chage	Change age: caducitat de la contrasenya.	
	-l usuari	Mostra caducitat per a l'usuari
chfn	Change user finger (finger=additional) information.	
su	Switch user (sense argument, root)	
	usuari	indicant l'usuari, un usuari determinat.
sudo	Permet fer accions amb privilegi d'administrador. Es configura qui té privilegi per executar sudo en el fitxer /etc/sudoers.	
sudoedit	Edita arxius amb privilegi d'administrador.	
visudo	Editor de l'arxiu /etc/sudoers.	
useradd		
usermod		
groupadd		
groupmod		

gpasswd	Permet afegir i treure usuaris a un grup, o establir una contrasenya.
groupdel	
userdel	Elimina un usuari. Compte! generalment es deshabilita el login. La eliminació es fa quan ja decidim eliminar els seus directoris, sense pressa.
lslogins	
exit	
login	
logout	
pwck	Verifica la integritat dels arxius de definició d'usuaris (passwd, shadow i group
arxius de configuració	
/etc/passwd	Usuaris.
/etc/group	Grups.
/etc/shadow	Contrasenyes.
/etc/gshadow	Gairebé no es fa servir (contrasenyes de grups).
/etc/login.defs	Rang dels UID d'usuari i de sistema i rang dels GID. Durada (aging) de les contrasenyes. Reintents de login per contrasenya fallida. UMASK per defecte (permisos) etc...
nsswitch.conf	Arxiu de configuració de NSS (Name Service Switch), que permet seleccionar, per a diferents elements, el mecanisme de resolucions de noms: arxiu local, LDAP, l'antic NIS, prèviament denominat Yellow Pages, DNS… PAM (Pluggable Authentication Module) és el sistema d'autentificació que fa servir les bases de dades d'usuaris seleccionades pel NSS.

Fitxers	
file stat ls ls -l ls -a find locate updatedb cat head tail touch ln readlink link unlink cp mv rm rm -r shred	

Directoris		
pwd		
cd		
mkdir		
rmdir		
ls	Llista els fitxers.	
	-l	Detalls (format llarg)

	-a	Mostra tots, també amagats
	-A	quasi tots.
tree	Visualitza l'arbre de directoris; no instal·lat per defecte.	

Permisos	
chmod	
umask	
chown	
chgrp	

exit
logout
shutdown / poweroff / halt / reboot

Captures de pantalla (mode gràfic)	
flameshot	Aplicació per a fer captures de pantalla, amb moltes opcions; no instal·lada per defecte.