

# **MÒDUL** ***Sistemes Informàtics***

## **UNITAT 1** ***Introducció als Sistemes Informàtics***

## ÍNDEX

1. Introducció
2. Components lògics dels sistemes informàtics: Representació de la informació.
3. Lògica de circuits
4. Components físics dels sistemes informàtics: Arquitectures.
5. Evolució dels sistemes informàtics
6. Virtualització
7. Seguretat informàtica

### 1.Introducció

#### Què són els ordinadors o ordinadors?

Dispositiu electrònic capaç de **processar la informació** rebuda, a través d'uns *dispositius de **entrada** (Input)*, i obtenir resultats que es mostraran fent ús d'uns *dispositius de **sortida** (Output)*, Gràcies a la direcció d'un **programa** escrit en el llenguatge de programació adequat.

# UNITAT 1. Introducció als Sistemes Informàtics

---

- Què dispositiu és?  
un **dispositiu electrònic**.
- Quin és el seu objectiu?  
**processar informació**.
- Com la processa?  
Rep dades dels **dispositius d'entrada**, Els processa i després mostra els resultats a través dels **dispositius de sortida**.
- Com sap l'ordinador com ha de processar les dades?  
Gràcies a un **programa, aplicació o conjunt d'instruccions** que indiquen a l'ordinador com ha de processar les dades d'entrada (input) i mostrar els resultats (output).

- **Hardware:** Són els **elements físics** com el monitor, teclat, ratolí o impressora. El que es pot veure i tocar.
- **Software:** És **informació.**, són les **dades** (*Música, documents pdf o word, Pel·lícules ...*) i els **programes** (*Jocs, editor de textos, reproductor de música o vídeo, visor de documents pdf...*).

### I què és un programa?

**programa:** Conjunt d'instruccions, escrites en un determinat llenguatge de programació, que controlen el funcionament d'un ordinador.

### Què és un Sistema Informàtic?

Uno o més ordinadors, el software associat, els perifèrics, els terminals, els operadors humans, els processos físics, els mitjans de transmissió de la informació, etc., que constitueixen un tot autònom capaç de realitzar un tractament de la informació

### Estructura d'un Sistema Informàtic



# 2. Components lògics dels sistemes informàtics: Representació de la informació.

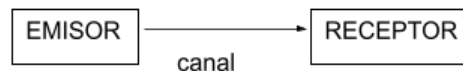
La informàtica és una tecno-ciència que estudia el **tractament automàtic** i racional **de la informació**, amb la finalitat d'obtenir la màxima utilitat de la mateixa.

Per a entendre què significa INFORMACIÓ, podem dir que és sinònim de coneixement, notícia, dades, etc. **Podríem pensar que sempre que hi ha comunicació hi ha informació, però no és així, la informació depèn molt del receptor.**

En realitat un SISTEMA DE COMUNICACIÓ, **està format per tres elements**:

- **Emissor**: Element que produeix un missatge
- **Canal**: Mitjà pel qual circula el missatge
- **Receptor**: Element que rep el missatge

• L'emissor i el receptor poden intercanviar els seus papers, fins i tot realitzar tots dos simultàniament.





# UNITAT 1. Introducció als Sistemes Informàtics

---

Amb tot això podem definir **INFORMACIÓ** com la **representació de fets, objectes, valors, idees, etc. que permet la comunicació entre emissor i receptor i l'adquisició de el coneixement de les coses.**

Per a l'adquisició del coneixement de les coses, **cal que emissor i receptor s'entenguin**, així que han d'utilitzar el mateix codi o, si no haurà de traduir aquesta informació, és a dir, codificar.

- **Codi**: Conjunt de condicions i convenis que permeten transformar la informació d'una representació concreta a una altra. **Està compost per:**

- conjunt de **regles** i conveni de l'alfabet font
- un nou **alfabet** que substituirà el font

- **Codificar**: Transformar unes dades a una representació predefinida i preestablerta.

Traslladant sistema de comunicació i d'informació a la informàtica obtenim:



## Tipus de sistemes de numeració

Es defineix sistema de numeració com: sistema que ens permet **representar quantitats numèriques seguint unes regles molt concretes**. És a dir, és un **codi compost per: un conjunt de símbols (nombres) i un conjunt de regles**.

Cada sistema de numeració es distingeix per la seva **BASE**, aquesta es defineix com: la quantitat de símbols que podem utilitzar per representar un nombre.

Per exemple:

- **Sistema de numeració decimal** (Utilitzat pels éssers humans), utilitza 10 símbols diferents per representar totes les quantitats.

$\text{BASE}_{(10)} \rightarrow$  utilitza 10 símbols  $\{0,1,2,3,4,5,6,7,8,9\}$

- **sistema binari** (Utilitzat per l'ordinador)

$\text{BASE}_{(2)} \rightarrow$  utilitza 2 símbols  $\{0,1\}$

# UNITAT 1. Introducció als Sistemes Informàtics

---

Existeixen molts sistemes de numeració, en aquest tema ens centrem en 3:

- Decimal (10 símbols): utilitzat pels éssers humans
- Binari (2 símbols): utilitzat per l'ordinador
- Hexadecimal (16 símbols): utilitzat per l'ordinador per aconseguir reduir la longitud de les dades que es produeix amb el sistema binari.

# UNITAT 1. Introducció als Sistemes Informàtics

---

## SISTEMA DECIMAL

Com ja hem comentat abans, el sistema de numeració decimal és l'utilitzat pels éssers humans, consta de 10 símbols  $\rightarrow \{0,1,2,3,4,5,6,7,8,9\}$  i és posicional.

## SISTEMA BINARI

Aquest és el sistema que va a manejar l'ordinador.

Com el seu nom indica, la seva base és 2, és a dir, utilitza tan sols dos símbols per representar les quantitats,  $\{0,1\}$ .

## SISTEMA HEXADECIMAL

La seva base és 16, utilitza 16 símbols  $\{0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F\}$ .

Es compleix que  $16 = 2^4$ , Cada dígit hexadecimal es correspon amb una combinació de 4 dígits binaris.

## UNITAT 1. Introducció als Sistemes Informàtics

Unidades básicas de información				
Prefijos del Sistema Internacional			Prefijo Binario	
Múltiplo - (Símbolo)	Estándar SI	Binario	Múltiplo - (Símbolo)	Valor
kilobit (kbit)	$10^3$	$2^{10}$	kibibit (Kibit)	$2^{10}$
megabit (Mbit)	$10^6$	$2^{20}$	mebibit (Mibit)	$2^{20}$
gigabit (Gbit)	$10^9$	$2^{30}$	gibibit (Gibit)	$2^{30}$
terabit (Tbit)	$10^{12}$	$2^{40}$	tebibit (Tibit)	$2^{40}$
petabit (Pbit)	$10^{15}$	$2^{50}$	pebibit (Pibit)	$2^{50}$
exabit (Ebit)	$10^{18}$	$2^{60}$	exbibit (Eibit)	$2^{60}$
zettabit (Zbit)	$10^{21}$	$2^{70}$	zebibit (Zibit)	$2^{70}$
yottabit (Ybit)	$10^{24}$	$2^{80}$	yobibit (Yibit)	$2^{80}$

# Conversions de sistemes de representació numèrica

- CONVERSIÓ DE DECIMAL A BINARI
- CONVERSIÓ DE BINARI A DECIMAL
- CONVERSIÓ DE DECIMAL A HEXADECIMAL
- CONVERSIÓ DE HEXADECIMAL A DECIMAL
- CONVERSIÓ DE BINARI A HEXADECIMAL
- CONVERSIÓ DE HEXADECIMAL A BINARI

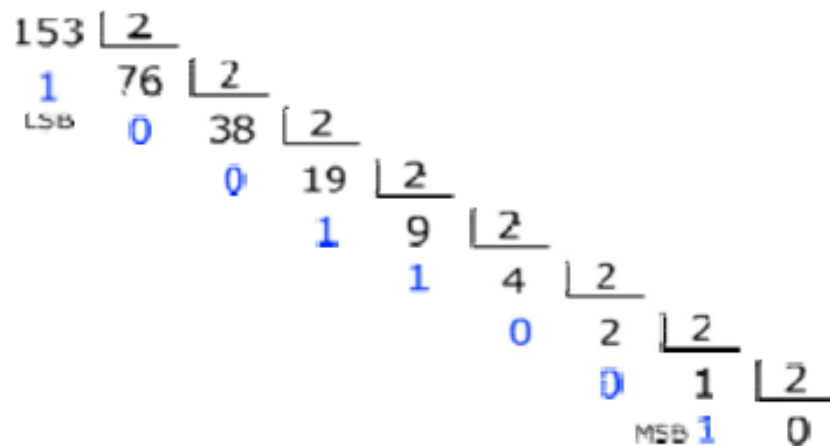
## CONVERSIÓ DE DECIMAL A BINARI.

En aquesta conversió es fan servir dos mètodes convencionals: El primer és *divisions successives* i el segon a través d'una taula.

### **Per divisions successives**

Aquest mètode consisteix a anar dividint la quantitat decimal per 2, apuntant els residus, fins a obtenir un quocient zero. l'últim residu obtingut és el bit més significatiu (*MSB*) i el primer és el bit menys significatiu (*LSB*).

**exemple:** Convertir el nombre 153 a binari.



El resultat en binari de 153 és  $(10.011.001)_2$



## UNITAT 1. Introducció als Sistemes Informàtics

---

### **A través d'una taula**

Aquest mètode consisteix a preparar una taula que és una successió exponencial de 2 escrita de dreta a esquerra, com aquesta:

2048 1024 512 256 128 64 32 16 8 4 2 1

A continuació s'agafa el nombre a convertir i es marca a la taula un 1 en la posició del nombre més alt de la taula inferior a aquest nombre.

Al nombre a convertir se li resta el marcat en la taula i es va fent el mateix amb el resultat successivament fins arribar a zero.

Les posicions no marcades a la taula s'omplen amb un 0.

## UNITAT 1. Introducció als Sistemes Informàtics

---

**exemple:** Conversió de l'nombre 1585 a binari

1024	512	256	128	64	32	16	8	4	2	1
1	1	0	0	0	1	1	0	0	0	1

$$1585 - 1024 = 561$$

$$561 - 512 = 49$$

$$49 - 32 = 17$$

$$17 - 16 = 1$$

$$1 - 1 = 0$$

El resultat en binari de 1585 és  $(11000110001)_2$

## UNITAT 1. Introducció als Sistemes Informàtics

---

### CONVERSIÓ DE BINARI A DECIMAL.

En aquesta conversió també es fan servir dos mètodes convencionals: El primer és *divisions utilitzant una fórmula i el segon emprant també una taula.*

#### **per fórmula**

Utilitzarem la fórmula següent:

$$a_n * b^{n-1} + a_{n-1} * b^{n-2} + a_{n-2} * b^{n-3} + \dots + a_2 * b^1 + a_1 * b^0$$

Sent a el dígit del nombre a convertir a la posició marcada  
b la base en la qual hi ha el nombre en aquest cas a l'ésser binari serà de 2  
n serà el nombre de dígitos que té el nombre binari a convertir.

## UNITAT 1. Introducció als Sistemes Informàtics

---

**Exemple:** Convertir el nombre 1001100101

Les posicions seran:

10	9	8	7	6	5	4	3	2	1
1	0	0	1	1	0	0	1	0	1

Amb el que aplicant les dades a la fórmula quedarà:

$$1 * 2^9 + 0 * 2^8 + 0 * 2^7 + 1 * 2^6 + 1 * 2^5 + 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 512 + 64 + 32 + 4 + 1 = 613$$

El resultat en decimal és 613.

### **A través d'una taula**

Aquest mètode consisteix a preparar una taula que és una successió exponencial de 2 escrita de dreta a esquerra, com aquesta:

2048 1024 512 256 128 64 32 16 8 4 2 1

A continuació es posa el nombre binari sota de cada posició de la taula.

Finalment se sumen els nombres de la taula de la primera fila marcada amb un 1.

## UNITAT 1. Introducció als Sistemes Informàtics

---

**exemple:** Conversió de l'nombre 1001101101

1024	512	256	128	64	32	16	8	4	2	1
	1	0	0	1	1	0	1	1	0	1

$$512 + 64 + 32 + 8 + 4 + 1 = 621$$

El resultat en decimal és: 621

# UNITAT 1. Introducció als Sistemes Informàtics

---

## CONVERSIÓ DE DECIMAL A HEXADECIMAL.

Hexadecimal és un sistema numèric base 16, això vol dir que té 16 dígitos on els deu primers números són els mateixos que en el sistema digital i els últims 6 són les 6 primeres lletres de l'alfabet, quedant la seva conversió de la següent manera:

dec	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

## UNITAT 1. Introducció als Sistemes Informàtics

---

En aquesta conversió anem a emprar el mateix mètode de la divisió emprat per convertir un nombre de decimal a binari, només que en aquesta situació el divisor serà 16.

**exemple:** Convertir el nombre 186.910 a hexadecimal.

$1869 \div 16 = 116$  ( $116 \times 16 = 1856$ ,  $1869 - 1856 = 13$ ) Resta = 13  
**13**  $116 \div 16 = 7$  ( $7 \times 16 = 112$ ,  $116 - 112 = 4$ ) Resta = 4

**abril 7**  $\div 16 =$  (Divisió no sencera, fins aquí es divideix)

Per tant la conversió seria: 7, 4 i 13. Com és hexadecimal, s'emporta el 13 al seu equivalent en aquest sistema  $13 = D$

El resultat en hexadecimal de 186.910 és 74D16.



## UNITAT 1. Introducció als Sistemes Informàtics

---

### CONVERSIÓ DE HEXADECIMAL A DECIMAL.

En aquesta ocasió utilitzarem el mateix mètode de la fórmula de la conversió de binari a decimal.

$$a_n * b^{n-1} + a_{n-1} * b^{n-2} + a_{n-2} * b^{n-3} + \dots + a_2 * b^1 + a_1 * b^0$$

Sent a el dígit del nombre a convertir a la posició marcada  
b la base en la qual hi ha el nombre en aquest cas a l'ésser hexadecimal serà 16  
n serà el nombre de dígit que té el nombre binari a convertir.

**Exemple:** Convertir el nombre B2C7

Aplicant les dades a la fórmula quedarà:

$$11 * 16^3 + 2 * 16^2 + 12 * 16^1 + 7 * 16^0 = 11 * 4.096 + 2 * 256 + 12 * 16 + 7 * 1 = 45.056 + 512 + 192 + 7 = 45.767$$

El resultat en decimal és 45.767.

# UNITAT 1. Introducció als Sistemes Informàtics

## CONVERSIÓ DE BINARI A HEXADECIMAL.

En aquesta ocasió utilitzarem la següent taula:

hexadecimal	DECIMAL	BINARI
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

## UNITAT 1. Introducció als Sistemes Informàtics

---

En la conversió agruparem els números de 4 en 4 sempre començant per la part dreta, si l'últim nombre no es queda amb una mida de 4 dígit s'omple amb zeros a la part esquerra.

Aquests grups es tradueixen directament fixant-nos en l'anterior taula, per exemple, si hi ha un grup que és 1010 es convertirà a l'dígit A.

El nombre resultant serà el seu equivalent valor en hexadecimal.

## UNITAT 1. Introducció als Sistemes Informàtics

---

**exemple:** Convertir el nombre 10111001101101

Els dividim en grups de 4 i omplim l'esquerra amb zeros:

10111001101101

0010 1110 0110 1101

Els seus equivalent en hexadecimal són:

0010: 2

1110: E

0110: 6

1101: D

El resultat seria: 2E6D

## UNITAT 1. Introducció als Sistemes Informàtics

---

### CONVERSIÓ DE HEXADECIMAL A BINARI.

Utilitzarem la mateixa taula que en la conversió de binari a hexadecimal, tan sols haurem d'agafar cada xifra hexadecimal i fer la traducció en la taula.

Exemple: Convertir el nombre BF79

L'equivalent a la taula és:

B: 1011

F: 1111

7: 0111

9: 1001

Si ajuntem tot el resultat serà: 1011111101111001

# OPERACIONS ARITMÈTIQUES A BINARI:

Suma:

0	1	0	1
+0	+0	+1	+1
<u>00</u>	<u>01</u>	<u>01</u>	<u>10</u>

carried bit

1110	→	1110	→	1110
+ 0111	→	+ 0111	→	+ 0111
<u>1110</u>	→	<u>1110</u>	→	<u>1110</u>
		1		01

11	11
1110	1110
+ 0111	+ 0111
<u>1110</u>	<u>1110</u>
101	10101

## OPERACIONS ARITMÈTIQUES A BINARI:

Resta:

$$\begin{array}{r} \text{-1} \\ 1101101101 \\ - 100101011 \\ \hline 1001000010 \end{array} \qquad \begin{array}{r} 877 \\ - 299 \\ \hline 578 \end{array}$$



## OPERACIONS ARITMÈTIQUES A BINARI:

Multiplicació:

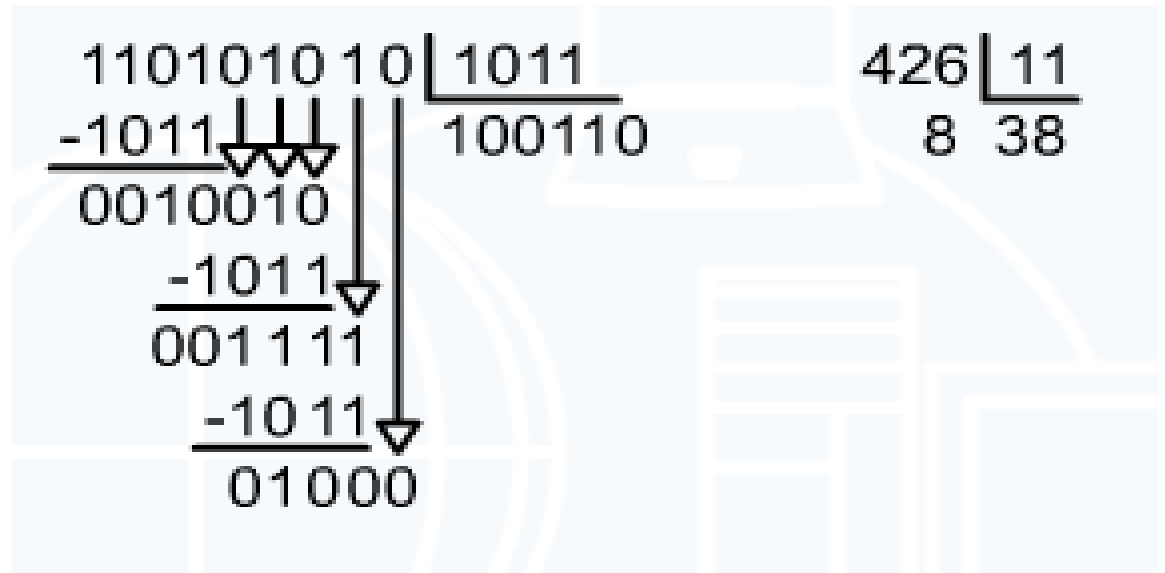
The diagram illustrates the binary multiplication process for the numbers 101101011 and 1101. It shows three steps:

- Step 1:** The initial multiplication of the least significant bit (1) of the multiplier by the multiplicand (101101011). The result is 101101011.
- Step 2:** The result from Step 1 is shifted one position to the left, resulting in 1101010110. This step is indicated by a red arrow labeled '2'.
- Step 3:** The results from Step 1 and Step 2 are added together to produce the final product, 1001001101111. This step is indicated by a red arrow labeled '3'.

The final product is 1001001101111.

## OPERACIONS ARITMÈTIQUES A BINARI:

Divisió:



### Representació interna de la informació

La informació alfanumèrica conté caràcters alfabètics, signes especials (signes de puntuació, parèntesi, etc.) i caràcters numèrics. La representació de tots aquests símbols es realitza assignant a cada un d'ells una única combinació d'uns i zeros.

Hi ha una taula de correspondència que assigna a cada caràcter una combinació binària. Aquesta taula rep el nom de codi. Hi ha diversos codis de representació de caràcters normalitzats, és a dir, que es pretén que tots els ordinadors presentin cada caràcter amb la mateixa combinació binària.

Els sistemes de **codificació alfanumèrica** més importants són:

El codi **ASCII** (Codi estàndard americà per el intercanvi de informació) originalment utilitzava 7 bits, és a dir, permetia representar  $2^7 = 128$  caràcters. Posteriorment es va ampliar a **8 bits** per poder representar els caràcters especials de cada idioma, conservant com a estàndard els caràcters de el 0 a 127. A continuació es mostra la taula ASCII de 8 bits:

# UNITAT 1. Introducció als Sistemes Informàtics

Código ASCII																
0	0	1 ☺	2 ☹	3 ♥	4 ♦	5 ♣	6 ♠	7 ●	8 ◼	9 ○	10 ◼	11 ♂	12 ♀	13 ♪	14 ♪	15 ☼
1	16 ▶	17 ◀	18 ⬆	19 !!	20 ¶	21 §	22 —	23 ⬇	24 ⬆	25 ⬇	26 ➡	27 ⬅	28 ⬇	29 ↔	30 ▲	31 ▼
2	32	33 !	34 "	35 #	36 \$	37 %	38 &	39 ' ,	40 (	41 )	42 *	43 +	44 ,	45 -	46 .	47 /
3	48 0	49 1	50 2	51 3	52 4	53 5	54 6	55 7	56 8	57 9	58 :	59 ;	60 <	61 =	62 >	63 ?
4	64 @	65 A	66 B	67 C	68 D	69 E	70 F	71 G	72 H	73 I	74 J	75 K	76 L	77 M	78 N	79 O
5	80 P	81 Q	82 R	83 S	84 T	85 U	86 V	87 W	88 X	89 Y	90 Z	91 [	92 \	93 ]	94 ^	95 _
6	96 ,	97 a	98 b	99 c	100 d	101 e	102 f	103 g	104 h	105 i	106 j	107 k	108 l	109 m	110 n	111 o
7	112 p	113 q	114 r	115 s	116 t	117 u	118 v	119 w	120 x	121 y	122 z	123 {	124	125 }	126 ~	127 ☐
8	128 Ç	129 ü	130 é	131 â	132 ä	133 à	134 å	135 ç	136 ê	137 ë	138 è	139 ï	140 î	141 ï	142 Å	143 Å
9	144 È	145 æ	146 Æ	147 ô	148 ö	149 õ	150 ú	151 û	152 ÿ	153 ö	154 ü	155 ç	156 £	157 ¥	158 Pt	159 f
A	160 á	161 í	162 ó	163 ú	164 ñ	165 Ñ	166 ã	167 õ	168 ð	169 ñ	170 ñ	171 ½	172 ¼	173 ½	174 «	175 »
B	176 ☐	177 ☐	178 ☐	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192 L	193 L	194 L	195 L	196 L	197 L	198 L	199 L	200 L	201 L	202 L	203 L	204 L	205 L	206 L	207 L
D	208 L	209 L	210 L	211 L	212 L	213 L	214 L	215 L	216 L	217 L	218 L	219 L	220 L	221 L	222 L	223 L
E	224 α	225 β	226 γ	227 π	228 Σ	229 σ	230 μ	231 τ	232 Φ	233 Θ	234 Ω	235 δ	236 ∞	237 φ	238 €	239 ∩
F	240 ≡	241 ±	242 ≥	243 ≤	244 ∫	245 J	246 ÷	247 ≈	248 °	249 .	250 .	251 √	252 n	253 2	254 ■	255 ■
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

## UNITAT 1. Introducció als Sistemes Informàtics

---

El codi **UNICODE**, és un codi estàndard internacional utilitzat als sistemes operatius. Utilitza **16 bits** per símbol, el que permet representar  $2^{16} = 65536$  caràcters. Amb 16 bits ja es poden representar tots els caràcters internacionals, però sorgeix un altre problema: com s'emmagatzema la informació a l'ordinador ?, a **low-endian** o en **big-endian**, en paraules simples és la forma en què el processador d'un computador llegeix la informació; d'esquerra a dreta o de dreta a esquerra. Les architectures x86 de Intel i els processadors Alpha de DEC fan servir low-endian, Mentre que les plataformes Sun 's SPARC, Motorola, i IBM PowerPC utilitzen la convenció big-endian.

Actualment l'estàndard **UTF-8**, Dins dels sistemes de codificació de UNICODE recull la codificació de la major part dels signes d'escriptura dels llenguatges occidentals i també dels orientals, i fins i tot de símbols matemàtics. per a això UTF-8 utilitza **entre un i 6 bytes** per descriure cada signe

## Representació d'imatges

Per representar imatges es distingeixen dos tipus d'imatges:

- imatges vectorials

La imatge es construeix a partir de vectors, que són objectes formats matemàticament com segments, polígons, arcs i altres figures, emmagatzemant diferents atributs matemàtics dels mateixos.

- Imatges de mapes de bits

S'emmagatzema la informació de cada píxel descomponent-lo en tres colors binaris (RGB), amb valors de mida depenent de el nombre de colors que admeti, el que es coneix com **profunditat de color**.

# UNITAT 1. Introducció als Sistemes Informàtics

---

Quantitat d'informació que ocupa una imatge matricial:

Hi ha diferents formats de imatges matricials, molts d'ells molt populars utilitzen compressió de la informació com jpeg, Gif, tiff, ...

No obstant això una imatge matricial sense comprimir i sense perdre per tant res d'informació ve marcada per tres paràmetres:

- Resolució vertical: nombre de píxels de la imatge en una línia vertical
- Resolució horitzontal: nombre de píxels de la imatge en una línia horitzontal.
- Profunditat de color: quantitat de bits que s'utilitza per definir el color de cada píxel.

Amb aquests paràmetres la mida que ocuparia una imatge matricial sense comprimir vindria marcat per la següent fórmula:

$$\text{Mida total} = \text{bits per a cada color} * \text{resolució horitzontal} * \text{resolució vertical}$$



# UNITAT 1. Introducció als Sistemes Informàtics

---

Exemple de càlcul de mida en una imatge matricial:

Imatge amb els següents paràmetres:

- Resolució: 640 x 480 (horitzontal x vertical)
- Profunditat de color: 24 bits

El càlcul seria:

$$\text{Mida} = 640 \times 480 \times 24 = 7.372.800 \text{ b}$$

Passant-lo a múltiples:

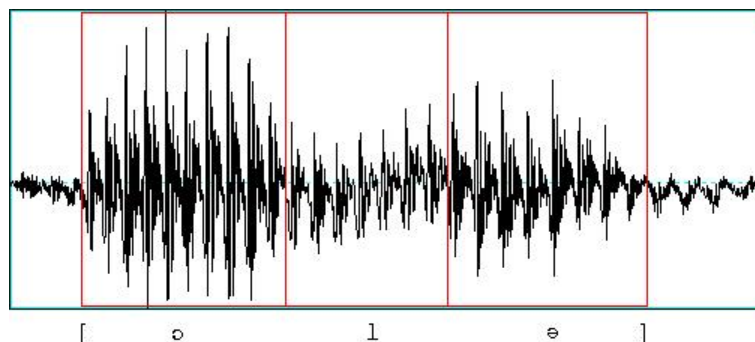
$$\text{Mida} = 7.372.800 / 8 = 921.600 \text{ B}$$

$$\text{Mida} = 921.600 / 1.024 = 900 \text{ kB}$$

## Representació de l'àudio

Per naturalesa, una ona analògica que transcorre durant un temps. Per emmagatzemar aquest so caldrà representar d'alguna manera aquesta ona perquè després es pugui emmagatzemar el senyal adequada a dispositius de sortida d'àudio.

L'ona de so sol tenir un aspecte similar a el següent:



## UNITAT 1. Introducció als Sistemes Informàtics

---

Per guardar el so d'aquesta ona es pren el valor de l'amplitud en binari amb un nombre de bits, anomenat **qualitat de l'mostreig**, Que determinarà la qualitat de la mateixa, habitualment 16 o 32 bits. Aquesta operació es fa de tant en tant, prenent-se un nombre de punts per segon al que s'anomena **freqüència** que es mesura en hertzs.

Per reproduir el so es reconstrueix l'ona a partir dels valors emmagatzemats.

La mida que ocuparia un so sense comprimir vindria marcat per a l' fórmula següent:

Mida = nombre de canals \* qualitat de mostreig \* freqüència (Hz) \* durada (s)

# UNITAT 1. Introducció als Sistemes Informàtics

---

## Exemple de càlcul de grandària en un àudio:

Audio amb els següents paràmetres:

- So estèreo (2 canals).
- Freqüència de mostreig: 44 kHz
- Qualitat de mostreig: 32 bits
- Durada: 1 minut

El càlcul seria:

$$\text{Mida} = 2 \times 44.000 \times 32 \times 60 = 168.960.000 \text{ b}$$

Passant-lo a múltiples:

$$\text{Mida} = 168.960.000 / 8 = 21.120.000 \text{ B}$$

$$\text{Mida} = 21.120.000 / 1.024 = 20.625 \text{ kB}$$

$$\text{Mida} = 20.625 / 1.024 = 20,14 \text{ MB}$$

## Representació de el vídeo

Un video és la seqüència de moltes imatges molt continuades en el temps acompanyat d'un audio.

Cadascuna d'aquestes imatges es denomina frame.

La mida d'un vídeo sense cap format de compressió es podria obtenir amb la següent fórmula:

Mida = Mida de la imatge \* nombre d'imatges + mesura de el so

### 3. Lògica de circuits

**Aquests circuits poden visualitzar-se com màquines que contenen un o més dispositius d'entrada i exactament 1 dispositiu de sortida.**

**A cada instant cada dispositiu d'entrada té exactament un bit d'informació, sigui un **0** o **1**; aquestes dades són processades pel circuit per donar un bit de sortida o sigui un **0** o **1**.**

Als dispositius d'entrada se'ls pot assignar successions de bits (totes les successions amb el mateix nombre de bits) Que són processats pel circuit bit per bit per produir una successió amb el mateix nombre de bits.

Es pot interpretar un bit com un voltatge a través d'un dispositiu d'entrada / sortida, una successió per exemple **1000110** es representa com: **1** **000** **11** **0** se suposa que el circuit sempre processa a la successió d'esquerra a dreta tret que s'especifiqui una altra cosa.

Els **circuits lògics** es construeixen a partir de circuits elementals anomenats **Portes lògiques**.

OR



Valor I:

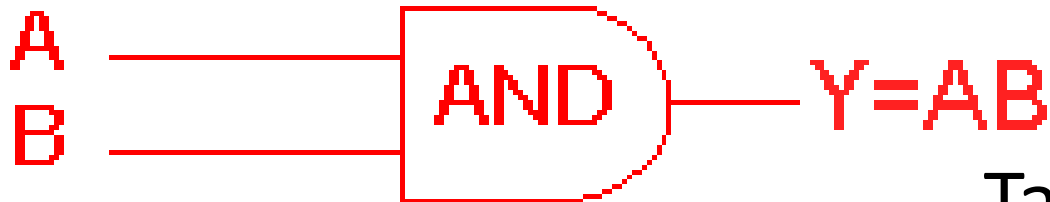
+	1	0
1	1	1
0	1	0

Taula de Veritat:

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1



## AND



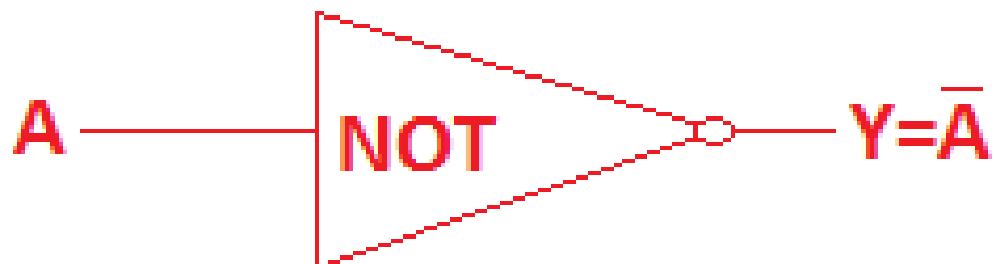
Valor de Y:

*	1	0
1	1	0
0	0	0

Taula de Veritat:

A	B	I = AB
0	0	0
0	1	0
1	0	0
1	1	1

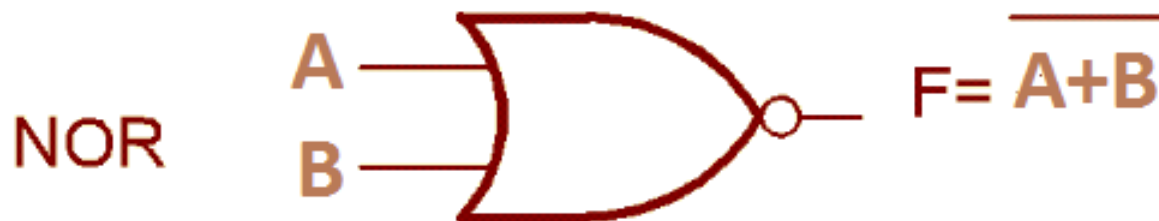
NOT



Valor de Y:

A	sortida
0	1
1	0

## UNITAT 1. Introducció als Sistemes Informàtics



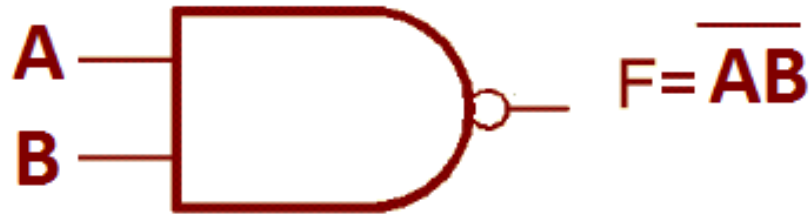
taules de Veritat: per a dos i tres variables

A	B	$I = \overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

A	B	C	$I = \overline{A+B+C}$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

# UNITAT 1. Introducció als Sistemes Informàtics

NAND



Taules de veritat: Per a 2 i 3 variables

A	B	$I = AB$
0	0	1
0	1	1
1	0	1
1	1	0

A	B	C	$I = ABC$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

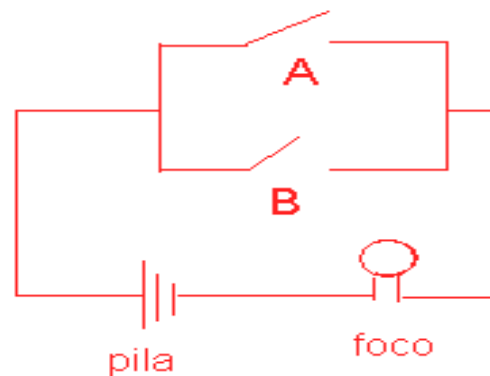
### **Les portes lògiques i la seva estreta relació amb els circuits elèctrics.**

La porta OR. Un circuit elèctric d'interruptors normalment conté alguna font d'energia, en aquest cas una pila, un dispositiu de sortida per exemple: una llum i un o més interruptors tots ells connectats per filferros.

Un interruptor és un dispositiu de dos estats que està **tancat** (**encès**) o **obert** (**apagat**) i el corrent pot passar a través de l'interruptor només quan l'interruptor estigui tancat.

### Porta OR i el seu circuit elèctric.

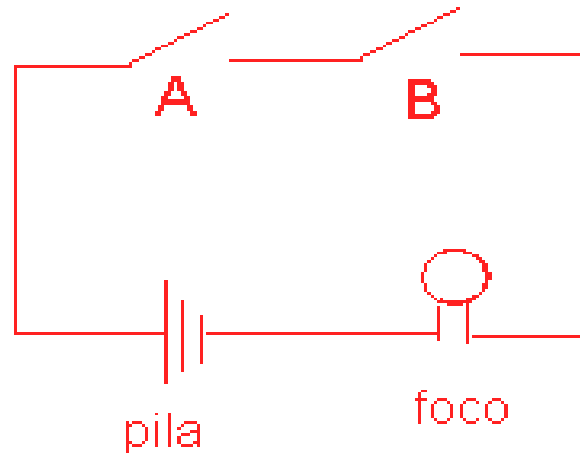
A la figura, els interruptors A i B estan connectats a **paral·lel**, S'observa que la llum s'encendrà si l'interruptor A està tancat o si el interruptor B està tancat, O si tots dos interruptors estan tancats. Aquesta és precisament la propietat de la taula de veritat per la comporta OR, en què 1 denota que l'interruptor (A, B) o el focus (A + B) aquesta encès i un zero denota que està apagat.



### Porta AND i el seu circuit elèctric.

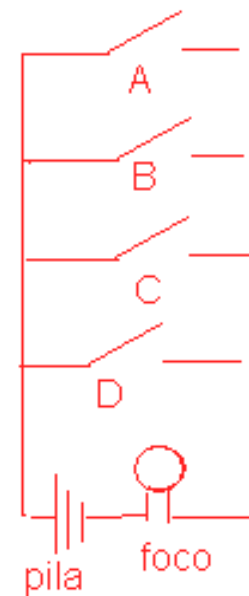
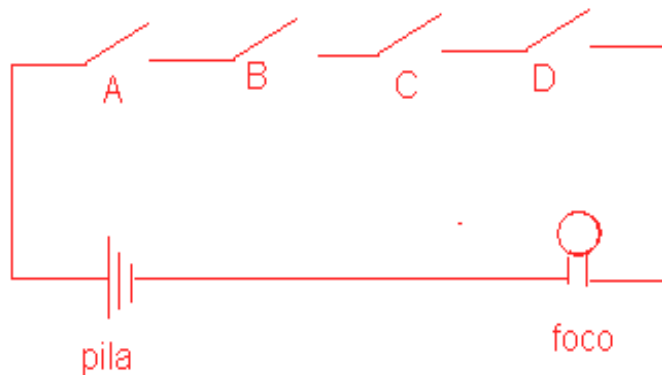
La figura mostra un circuit d'interruptors A i B connectats en sèrie.

En la qual s'observa que la llum s'encendrà només quan estan tancats tant A com B.



## UNITAT 1. Introducció als Sistemes Informàtics

les portes **OR**, **AND** i els seus respectius circuits elèctrics poden tindre més de dos entrades.





exercicis:

1.- Donada l'expressió  **$F = ABC + A \bar{B}C + \bar{A}B$**

- a) determinar el seu circuit lògic
- b) Determinar el seu circuit d'interruptors
- c) Determinar la seva taula de veritat

### 4. Components físics dels sistemes informàtics: Arquitectures.

#### L'arquitectura d'ordinadors

- L'arquitectura d'ordinadors és **l'estudi de la seva estructura, operació i disseny.**
- Fonamentalment això es refereix a aspectes de hardware, encara que pot també afectar a qüestions de software.

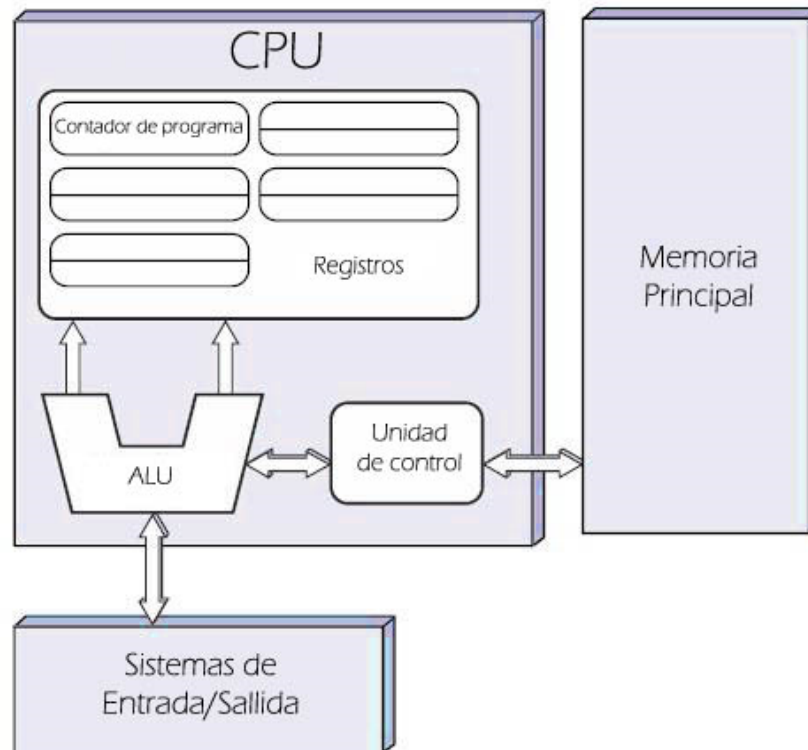
### L'arquitectura de von Neumann

- ✓ A 1944 **John von Neumann** va descriure un model de computador amb programa emmagatzemat en memòria elèctrica.
- ✓ El que pretenia Von Neumann era **construir un ordinador amb programa emmagatzemat**, per al que necessitava connectar permanentment les unitats de l'ordinador **i coordinar el funcionament sota un control central**.

## L'arquitectura de von Neumann

- ✓ En els primers ordinadors, les dades i el programa no estaven emmagatzemats en memòria. Aquest concepte va ser tan important i decisiu que va donar lloc a el concepte de l'arquitectura de Von Neumann.
- ✓ Segons l'arquitectura de Von Neumann, **un ordinador està format per:**
  - Unitat aritmeticològica (ArithmeticLogicUnit) ALU.
  - Unitat de Control (UC)
  - Memòria
  - Unitat d'entrada / Sortida

## L'arquitectura de von Neumann



## arquitectura von Neumann

### ✓ Unitat aritmètic-lògica (ALU)

Porta a terme **operacions elementals** com sumes, restes i **operacions lògiques** (AND, OR i NOT entre d'altres). Les dades amb què s'operen, procedeixen de la memòria principal i temporalment s'emmagatzemen en uns registres de l'ALU.

Les operacions lògiques s'implementen mitjançant dispositius electrònics anomenats portes lògiques, que són la base de la implementació dels circuits de commutació integrats en un xip.

**AND:**  $A \cdot B$

**OR:**  $A + B$

### L'arquitectura de von Neumann

#### ✓ Unitat de Control (UC)

**Interpreta cadascuna de les instruccions** del programa en llenguatge màquina i genera els senyals perquè es realitzin les modificacions sobre els registres i / o les posicions de la memòria principal.

Per conèixer la posició de memòria de les instruccions que es van a executar, hi ha un registre apuntador anomenat **comptador de programa**.

## L'arquitectura de von Neumann

### ✓ Memòria Principal

Es correspon amb la memòria RAM (RandomAccess Memory). És un dispositiu totalment electrònic i que funciona a molt altes velocitats. No obstant això, té una capacitat reduïda i un preu relativament alt.

Està formada per un **conjunt de cel·les de la mateixa mida**. Cada cel sol estar **composta per un conjunt de bits, que es denominen punt de memòria**.

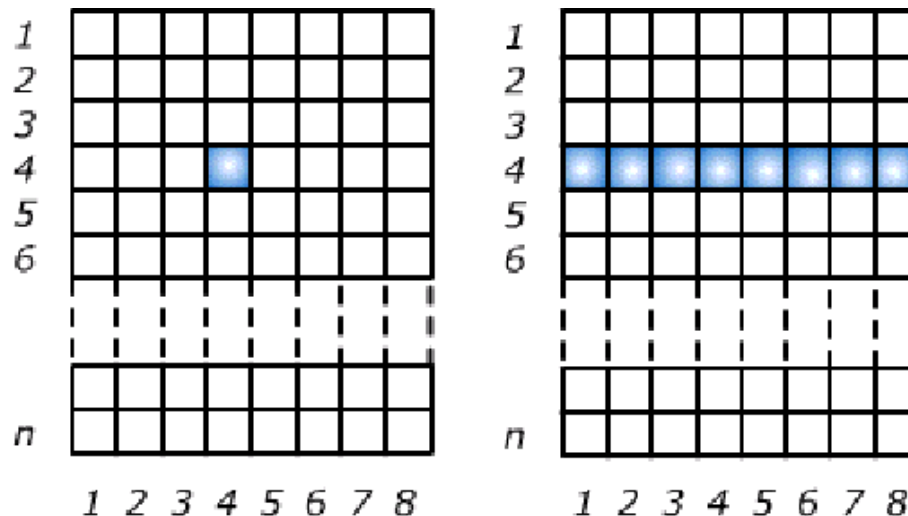
Aquests punts tenen dos possibles valors 0 o 1, que corresponen a estats de tensió diferents.

En les cel·les **s'emmagatzemen tant dades com instruccions màquina**.



## L'arquitectura de Von Neumann

- **Memòria Principal**



## L'arquitectura de Von Neumann

- **Memòria Principal**
  - la memòria principal o RAM guarda les dades que s'estan utilitzant en el moment actual, amb l'equip encès i operatiu.
  - Quan l'equip s'apaga, la informació que conté s'elimina (memòria volàtil)
  - La seva capacitat de emmagatzematge actualment es mesura a gigabytes (GB), sent comú avui en dia : 8 GB, 16 GB ...

### L'arquitectura de Von Neumann

- **Memòria principal**
  - A l'actualitat , els ordinadors tenen memòria RAM a molts components interns.
  - Per exemple, al processador (memòria caché, registres), a lectors òptics (buffer o caché) o a targetes gràfiques (memòria de vídeo o gràfica) encara que, quan es parla de memòria RAM, s'està parlant principalment dels mòduls de memòria que s'insereixen a la placa base.

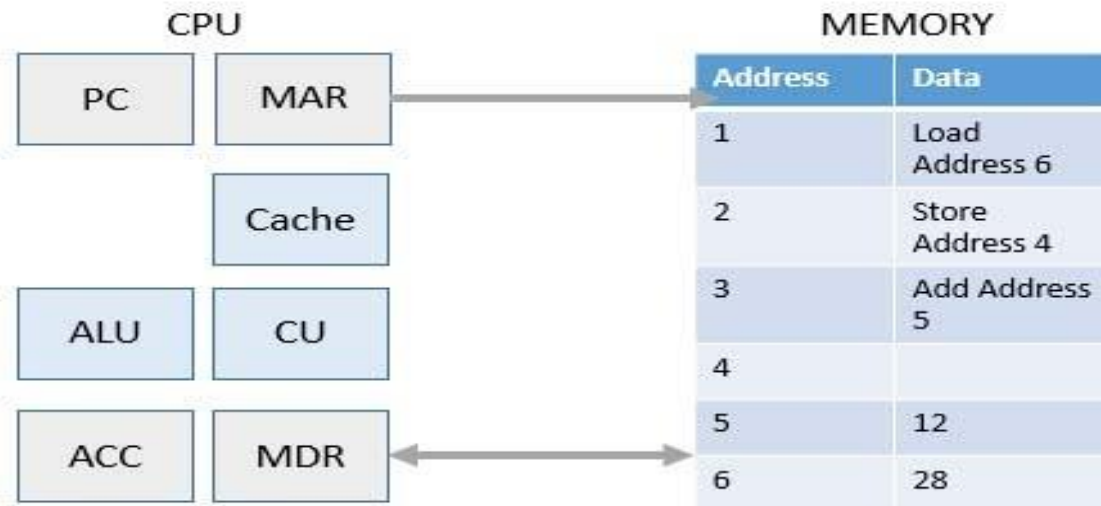
### L'arquitectura de Von Neumann

#### Memòria principal

- ✓ Les dades de la memòria principal s'organitzen en una mena de **casellers numerades** de manera que tota la informació es pugui localitzar ràpidament.
- ✓ El caseller rep el nom de posició de memòria, i la seva identificació o nombre d'ordre s'anomena **direcció de memòria**.
- ✓ Quan s'emmagatzema una dada en una posició de memòria, aquest roman en ella fins que s'apagui l'equip.

## L'arquitectura de Von Neumann

- Memòria Principal**



## L'arquitectura de Von Neumann

- Unitat Entrada / Sortida

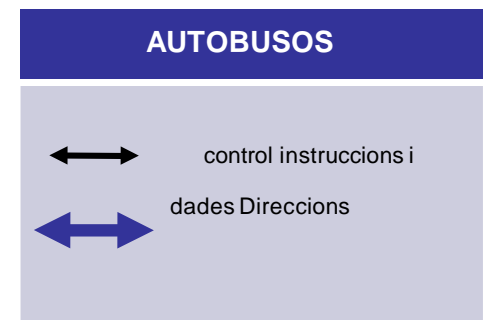
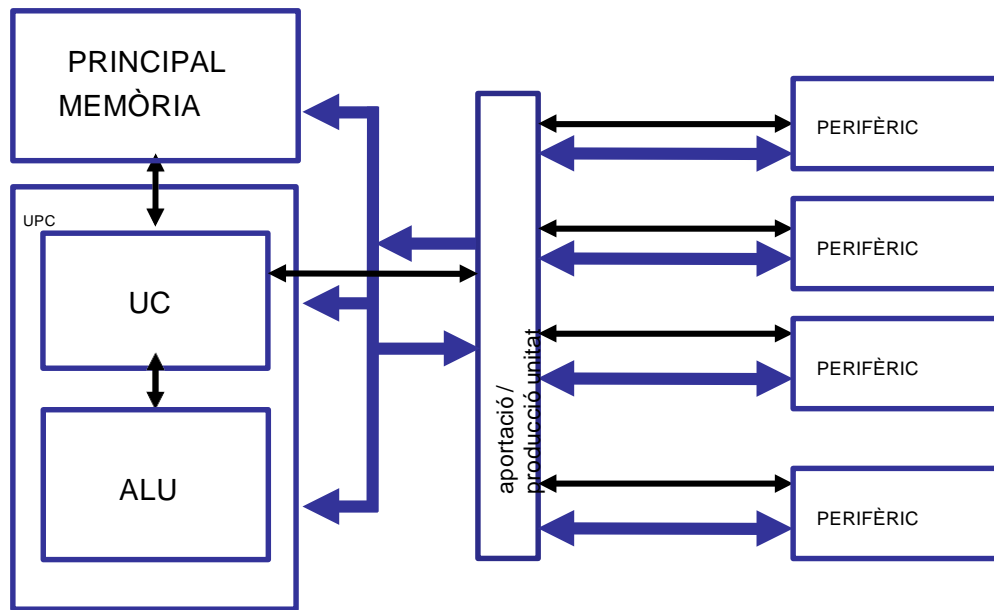
Les **unitats d'entrada i sortida**, també abreuïat I/O (Input / Output), són les interfícies que **utilitzen les diferents unitats funcionals** (subsistemes) d'un sistema de processament d'informació **per comunicar-se** unes amb altres, o els senyals (informació) enviades a través d'aquestes interfícies.

**Les entrades són els senyals rebuts per la unitat, mentre que les sortides són els senyals enviats per aquesta.**

Transfereix la informació a través de canals associats (busos) a aquestes unitats

# UNITAT 1. Introducció als Sistemes Informàtics

## L'Arquitectura de von Neumann



## La Unitat Central de Procés (CPU)

### Components

- **La Unitat Central de Procés (CPU)**

És l'autèntic **cervell del ordinador**, controla i governa tot el sistema.

La UCP o CPU (Central Process Unit), consisteix en un **circuit que interpreta i executa les instruccions dels programes emmagatzemats en memòria** i que a més pren les dades de les unitats d'entrada, les processa i les envia a les unitats o perifèrics de sortida.

Podem resumir que **s'encarrega del control i el procés de dades.**



## La Unitat Central de Procés (CPU)

### Components

#### La Unitat Central de Procés (CPU)

Està formada per:

- ✓ **Unitat de control (UC).** Interpreta i executa les instruccions màquina emmagatzemades en la memòria principal i genera els senyals de control necessàries per executar-les.
- ✓ **La unitat aritmètic-lògica (UAL o ALU).** Rep les dades, efectua operacions de càlcul, comparacions, presa de decisions lògiques (àlgebra de Boole) i retorna el resultat. Tot això sota la supervisió de la UC.
- ✓ **Registres.** On s'emmagatzema informació temporal (dades en execució), que constitueix l'emmagatzematge intern de la CPU.

## La Unitat Central de Procés (CPU)

### Execució d'instruccions

Tota instrucció màquina resident a memòria principal passa per una sèrie de fases que va de la seva captura fins la seva interpretació i execució. aquestes etapes són les següents:

- Càrrega, búsqueda o lectura.
- Decodificació.
- Execució.
- Increment del comptador de programa.

## La Unitat Central de Procés (CPU)

### Execució de instruccions

Una vegada conegudes les fases i els registres, anem a explicar el funcionament de les mateixes:

- **càrrega, recerca o lectura:** la UC envia a la memòria principal la direcció de la instrucció a executar, que està emmagatzemada en el comptador de programa (PC) i activa els senyals de control.
- **descodificació:** la UC rep la instrucció, la analitza i, on sigui apropiat, llegeix els operands de la memòria principal (RAM), enviant la seva direcció i activant les senyals de control.

## La Unitat Central de Procés (CPU)

### Execució de instruccions

- **Execució:** l'ALU, sota el control de la UC, realitza la operació sobre els operands i emmagatzena el resultat en els registres o a la memòria principal.
- **Increment del Comptador de Programa:** també denominat punter d'instrucció, de manera que es pot passar a executar la instrucció següent. Hi ha instruccions que poden modificar el registre CP, donant lloc a bifurcacions. Una d'aquestes bifurcacions, les provocades per interrupcions, externes o internes, s'anomenen traps.

## El sistema de memòria

### Jerarquia de memòria

Per optimitzar els temps en què el processador obté una dada de la memòria s'organitzen els diferents tipus de memòries en el que s'anomena **jerarquia de memòries**.

Aquesta es fa servir per millorar el rendiment dels diferents tipus de memòries.

La memòria quedarà organitzada de més ràpida a més lenta, que per regla general també mantindrà l'ordre de més petita a més gran (pel que fa a capacitat d'emmagatzematge).

## El Sistema de mMemòria

### Jerarquia de memòria

Les característiques que es demanen de les memòries són les següents:

- **Capacitat d'emmagatzematge**
- **Velocitat:** Ve donada pel temps d'accés. Com més baix temps d'accés tingui, més veloç serà.
- **Cost per bit:** A igualtat de capacitat, una memòria de 1GB tindrà un cost per bit més gran que una memòria de 250 GB.

La memòria ideal, serà aquella que tingui molta capacitat d'emmagatzematge, que treballi a grans velocitats (amb un temps d'accés molt baix) i que costi pocs diners.

## El sistema de Memòria

### Jerarquia de memòria

El problema és que quan es construeixen memòries amb un temps d'accés molt petit el cost s'incrementa molt.

Les memòries més ràpides, són més cares (el cost per bit és molt alt), per tant es construeixen memòries ràpides però amb poca capacitat. D'altra banda, quan es construeixen memòries amb molta capacitat, aquestes són molt lentes.

La solució és utilitzar cada memòria diferent per a un propòsit determinat, amb la qual cosa, es poden classificar les memòries de forma jeràrquica per nivells.

## el sistema de memòria

### Jerarquia de memòria

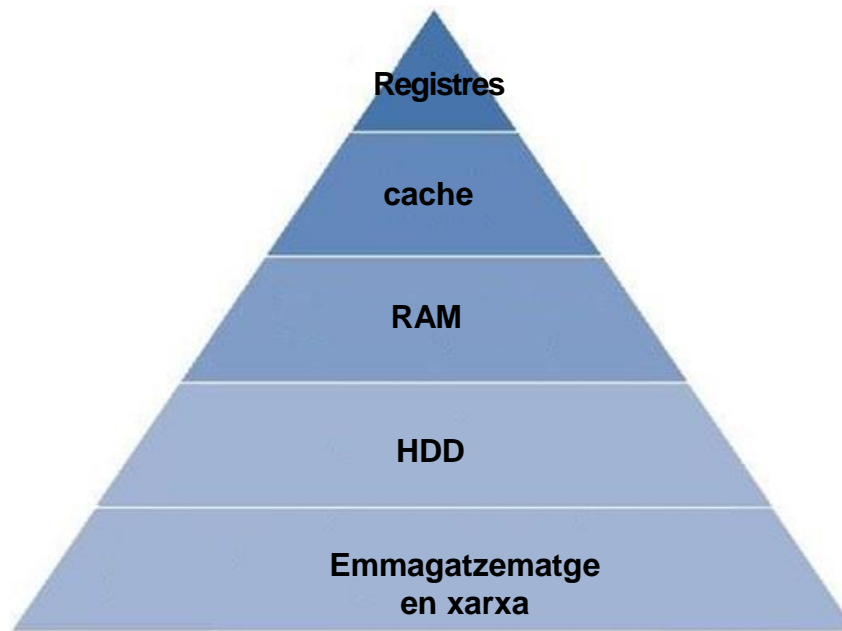
En general, cada nivell jeràrquic tan sols es pot relacionar amb un nivell superior i / o posterior.

La distribució jeràrquica de las diferents memòries depèn de la seva proximitat a l' processador.



## el sistema de memòria

### Jerarquia de memòria



## El sistema de memòria

### Nivell 0. Registres

Són memòries molt veloços però amb poca capacitat. Estan integrats en el processador.

Aquests registres s'emmagatzemen dades transitoris utilitzats pel processador, **generalment resultats d'operacions matemàtiques**.

Si el processador utilitza de manera repetitiva certs valors, aquests també s'emmagatzemen en els registres.

**Utilitza tecnologia SRAM.**

## El sistema de memòria

### Nivell 0. Registres

#### Tipus:

- Registres de dades
- Registres de memòria
- Registres de propòsit general
- Registres de coma flotant
- Registres de propòsit específic

## El sistema de memòria

### Nivell 1. memòria caché

És una **memòria més petita i ràpida**. Situada entre la memòria principal i el processador.

El seu objectiu és emmagatzemar còpia de les **dades** situats en memòria principal els quals o són **molt utilitzats** o es preveu que es **van a utilitzar en un futur immediat**.

Quan s'accedeix a una dada, aquest es col·loca a la memòria caché. La segona vegada que s'accedeix a aquesta dada, no caldrà accedir a la memòria principal ja que la dada ja és a la caché.

## El sistema de memòria

### Nivell 1. Memòria caché

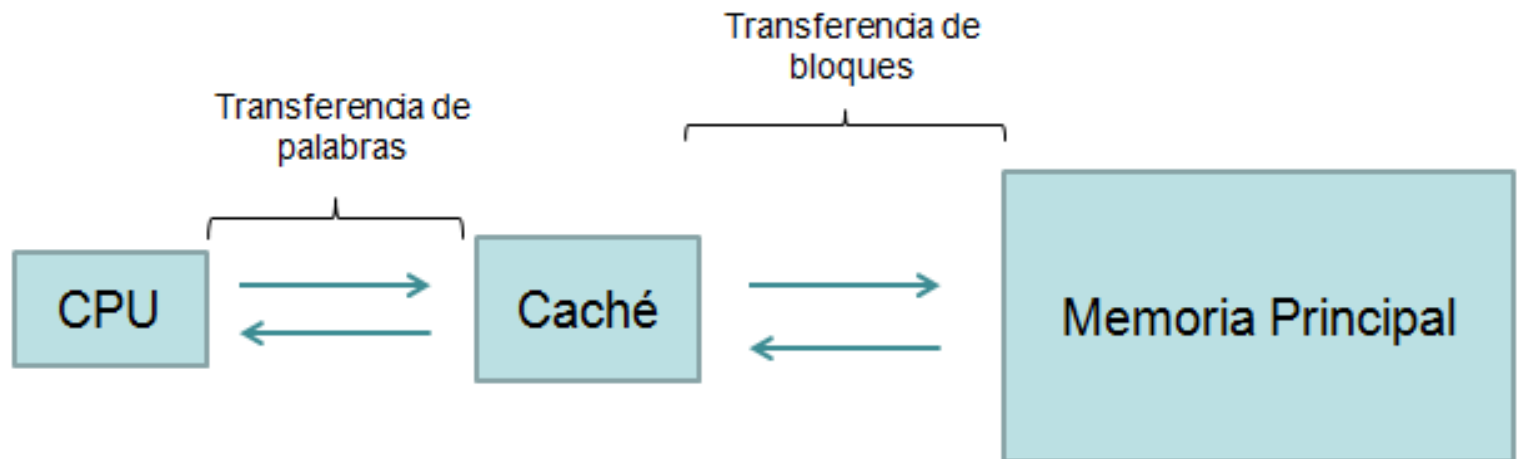
Un resum de l'operació de la memòria caché podria ser el següent:

- La CPU sol·licita continguts de la localització de memòria.
- Comprova la memòria caché per a aquestes dades.
- Si hi és, l'obté de la memòria caché (ràpidament).
- Si no hi és, llegeix el bloc requerit a partir de la memòria principal fins a la memòria caché.
- Després, de la memòria caché els lliura a la CPU.
- La memòria caché inclou etiquetes per identificar quin bloc de la memòria principal està en cada ranura de la memòria caché.

## El sistema de Memòria

### Nivell 1. memòria caché

Un resum del funcionament de la memòria caché podria ser el següent:



## El sistema de memòria

### Nivell 2. memòria principal o RAM

S'insereix **en l'slot de la placa base**.

És una unitat que **es divideix en cel·les que s'identifiquen per una direcció**. Està formada per blocs de circuits integrats o xips, capaços d'emmagatzemar, retenir valors binaris.

## El sistema de memòria

### Nivell 3. disc dur o memòria secundària

És un dispositiu d'emmagatzematge **d'informació no volàtil**, és a dir, conserva la informació encara que no disposi de corrent elèctric. Fa servir un sistema de gravació magnètica per emmagatzemar dades digitals.

En aquest nivell **s'inclou el mecanisme de memòria virtual**, el qual utilitza espai en disc per gestionar la memòria.

La memòria virtual permet als programes usar més memòria de la disponible físicament.



## El sistema de memòria

### Nivell 4. Emmagatzematge extern a xarxes

L'emmagatzematge en xarxa s'està popularitzat donat el creixement i millora de les xarxes pel que fa a velocitat i prestacions.

Actualment les necessitats pel que fa a la informació són que sigui accessible des de qualsevol dispositiu i qualsevol lloc el que implica que haguem de emmagatzemar les dades en la xarxa.

Fins ara, els sistemes més populars són els NAS (Network AttachedStorage), Encara que últimament s'està popularitzant molt l'emmagatzematge en el núvol.

Alguns dels sistemes més utilitzats utilitzades són Drive, onedrive o Dropbox.

## El sistema de busos

### Introducció

En informàtica, un bus és un **conjunt cablejat que serveix per que els dispositius hardware puguin comunicar-se entre si.**

Són rutes compartides per tots els dispositius i els permeten transmetre informació d'uns als d'altres, són, en definitiva, les autopistes de la informació interna, les que permeten les transferències de tota la informació manejada pel sistema.

A un bus, tots els nodes connectats a ell reben les dades que es bolquen, però només aquell dispositiu a què va dirigida la informació és qui la pren i la processa, el resta la ignora.

## El sistema de busos

### Introducció

Els busos **poden ser tant paral·lels com a sèrie**. quan és paral·lel és perquè hi ha diversos cables que condueixen la informació; sense això, si és sèrie només hi ha un canal per el que es transmet la informació.

Si el comparem amb una carretera, el bus paral·lel tindria diversos carrils, mentre que el bus sèrie, només té un carril per que circulin els vehicles.

Es denomina **ample de bus** a l' mida d'aquest nombre de fils o bits que es transmeten simultàniament per un d'aquests canals.

## El sistema de busos

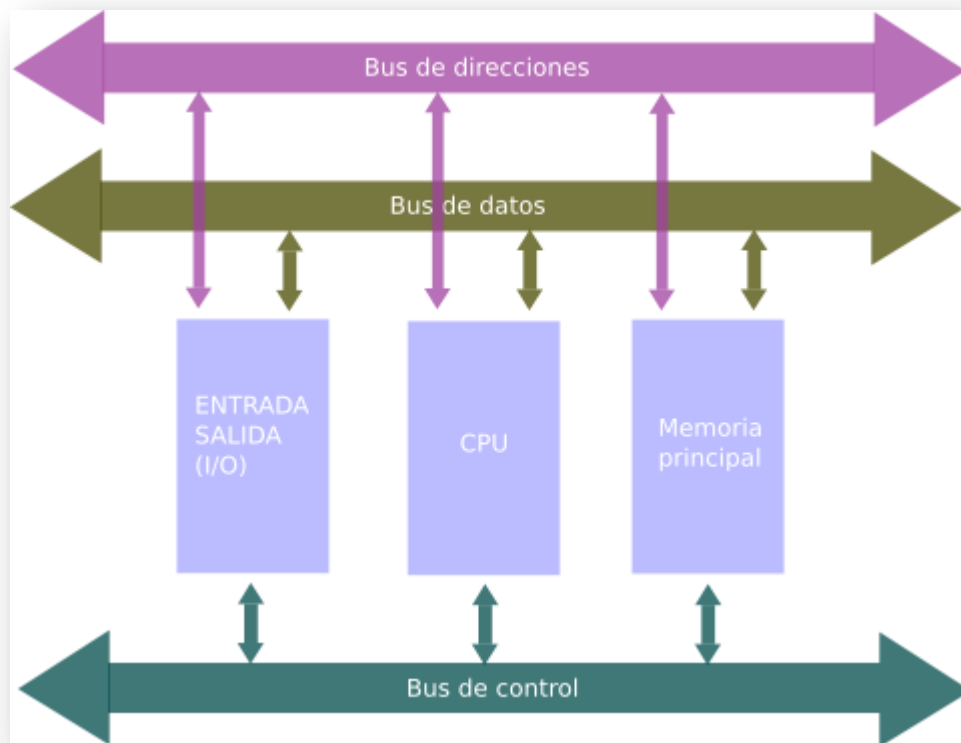
### Introducció

Es poden distingir 3 tipus de busos:

- **Bus de dades** (Bidireccional). Transporta dades procedents o amb destinació a la memòria principal i les unitats de Entrada / Sortida.
- **Bus de direccions** (Unidireccional). transporta les direccions de la unitat de control a la memòria principal o als perifèrics.
- **Bus de control** (Bidireccional). transporta les senyals de control (Microórdenes) generades per la unitat de control.

## El sistema de busos

### Introducció



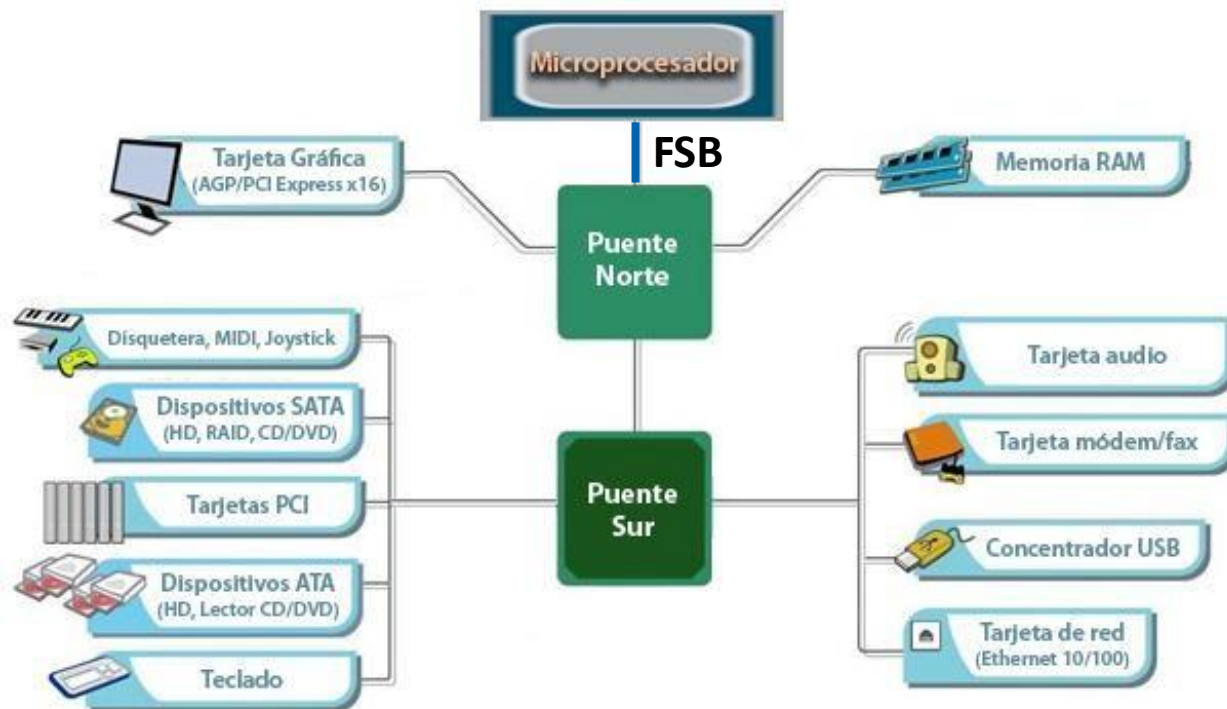
### El sistema de busos

#### Els busos de la placa base

- **El FSB** o *Front Side Bus* és la connexió entre el processador i el *Northbridge* (xip que controla la CPU, la RAM, la targeta gràfica i el *Southbridge*). En alguns llocs encara s'anomena bus d'el sistema.

## El sistema de buses

### Els busos de la placa base



## El sistema de busos

### Els busos de les plaques de expansió

- **Bus PCI** (*Peripheral Component Interconnect*). Transmet dades a paral·lel. serveix per connectar dispositius perifèrics directament a la placa base. Va reemplaçar a busos antics com els ISA i els VESA. el PCI va desapareixent donant pas a l'PCI Express.
- **Bus AGP** (*Accelerated graphics Port*). Està dedicada a connectar targetes de vídeo.
- **Bus PCI Express**. També pot trobés com PCI-E. és la evolució de l' bus PCI. suposa unamillora substancial, arribant a superar la taxes de transferència de l' bus AGP. Per aquest motiu, seva ús està especialment orientat a les targetes gràfiques.



### El sistema de Entrada / Sortida

Fins ara hem vist components necessaris per gestionar la informació que s'introdueix, però necessitem dispositius per introduir o treure aquesta informació.

El concepte de entrada / sortida fa referència a tota comunicació o intercanvi d'informació entre la CPU o la memòria amb l'exterior.

Aquestes operacions es solen portar a terme a través d'uns dispositius externs que es denominen **perifèrics** que proporcionen a l'ordinador les vies per intercanviar dades amb el exterior.

### El sistema de Entrada / Sortida

La part de l'equip que permet aquesta comunicació és la **unitat de entrada-sortida**, entenent per tal concepte a architectures reals a un conjunt de mòduls o **canals de entrada-sortida** encarregats de governar un o més perifèrics associats als quals subministra la intel·ligència necessària per el seu funcionament coordinant amb l'ordinador.

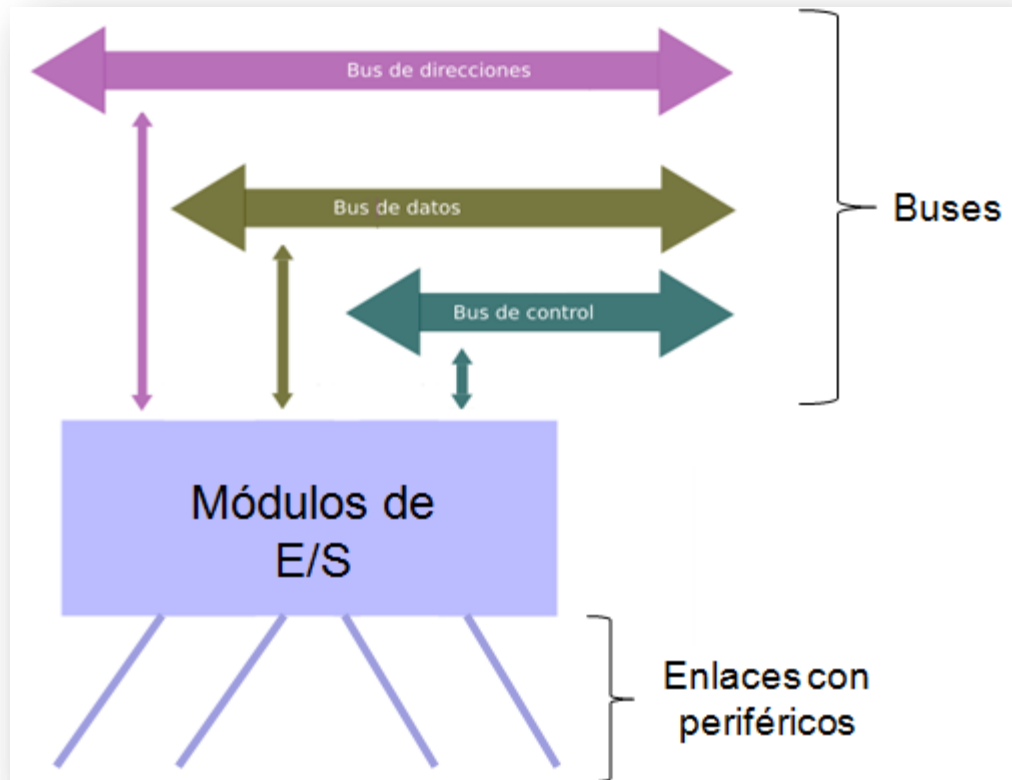
Aquests mòduls d'entrada-sortida estarien formats pels controladors de perifèrics, de forma que cada perifèric necessita el seu propi controlador per comunicar-se amb la CPU i amb els **ports de entrada-sortida**, que són els registres que es connecten directament a un de els busos de l'ordinador.

### El sistema de Entrada / Sortida

Cada port té associada una direcció o codi, de forma que el processador veu a l' perifèric com un port o unconjunt de ports.

Aquests mòduls o canals s'usen per resoldre les diferències que poden existir entre el processador i aquests perifèrics. Les seues funcions fonamentals són adreçament, transferència i sincronització.

### El sistema de Entrada / Sortida



### 5. Evolució dels sistemes informàtics

- 1<sup>a</sup> Generació (1940-1956)
  - Ús de vàlvules de buit
  - Fins militars i científiques
  - Màquines molt grans, pesades i lentes
- 2<sup>a</sup> Generació (1956-1963)
  - Aparició de transistor: màquines més petites i amb menor consum.
  - Fins comercials: sorgeixen ordinadors amb fins comercials.
  - Sèrie IBM 7090: primer ordinador comercial.
  - Perifèrics: apareixen els primers perifèrics.
  - Superordinador: ordinador amb capacitat de càlcul per sobre de la resta.
  - Primers llenguatges de programació

- 3<sup>a</sup> Generació (1964-1971)
  - Circuits integrats: en un sol xip s'integren tots els transistors i circuits.
  - Nous suports d'emmagatzematge: discos flexibles.
  - Nous conceptes: miniordinador, estació de treball
  - Llenguatges d'alt nivell (C, PASCAL, BASIC)

- 4<sup>a</sup> Generació (1971-1981)
  - Popularització de l'microordinador i ordinador personal o domèstic
  - Alta escala d'integració (LSI): milers de transistors en un microxip.
  - Microprocessador: Conté tots els elements de càlcul.
  - Proliferen els llenguatges de programació
  - S'estén l'ús d'ordinadors a casa.

- 5a Generació (1982-1991)
  - Es segueix evolucionant el microprocessador
  - Tecnologia VLSI: 10.000 a 100.000 transistors en un microxip.
  - Microprocessadors d'ús específic
  - Tecnologia multimèdia
  - Apareix la GUI (Interfície d'usuari gràfica)
  - Baixen els preus
  - Microprocessadors en paral·lel
  - Ús generalitzat de xarxes



- 6a Generació

- **ULSI** (100.000 a 1.000.000 d'transistors) i **GLSI** (Més de 1.000.000 de transistors)
- Múltiples microprocessadors que treballen al mateix temps.
- Aplicacions suportades per Internet
- Arquitectura paral·lel-vectorial
- **WAN** (Xarxa d'àrea mundial)

### 6. Virtualització

La virtualització és la creació d'una màquina hardware simulada dins un altre ordinador via software.

L'amfitrió és l'equip físic on s'instal·la la màquina virtual.

L'hoste és la màquina virtual dins d'un equip físic.

L'hipervisor és l'element (programa) encarregat de la virtualització.

El hipervisor pot ser:

- Natiu: S'executa directament l'hoste sobre el hardware de l'equip amfitrió.

Exemples: Microsoft HiperV, Citrix Server, KVM

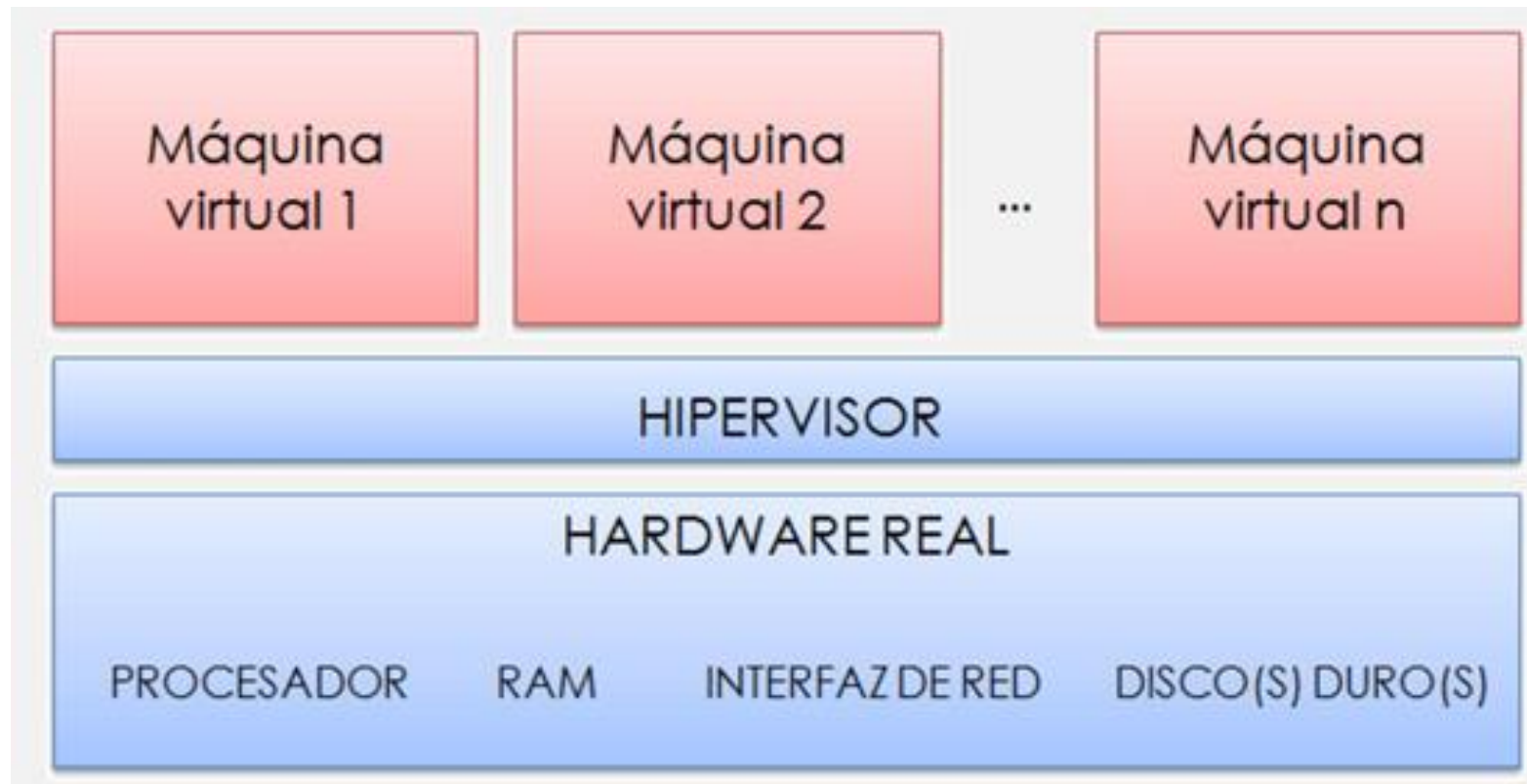
- Allotjat: S'instal·la l'hoste en una aplicació instal·lada en el Sistema Operatiu de l'equip amfitrió

Exemples: VirtualBox, VMWare

# Hipervisor nativo

## UNITAT 1. Introducció als Sistemes Informàtics

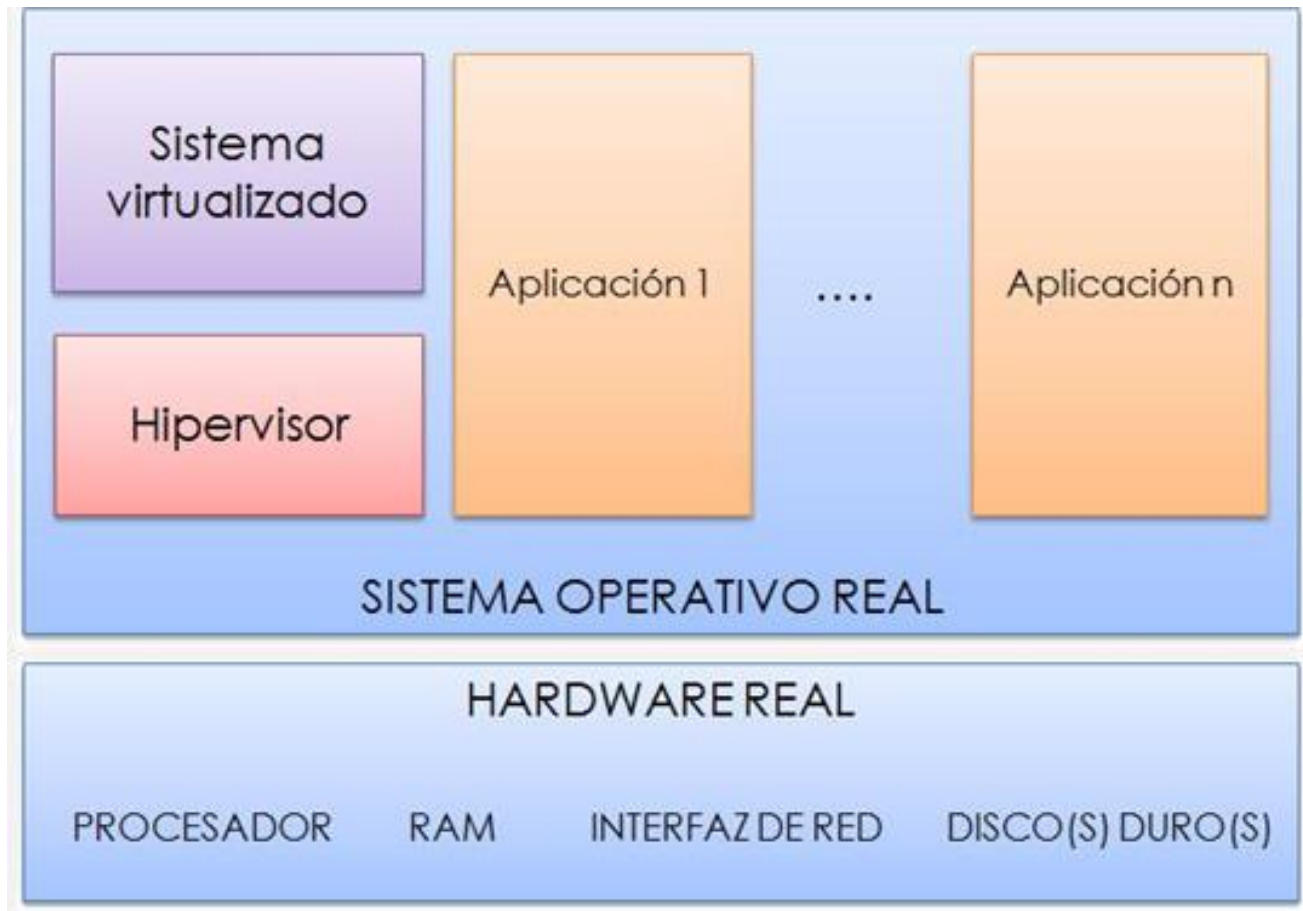
---



# Hipervisor alojado

## UNITAT 1. Introducció als Sistemes Informàtics

---



### Tipus de virtualització

- Virtualització de recursos:

Es simulen recursos hardware solts com pot ser volums d'emmagatzematge.

- Virtualització de plataforma:

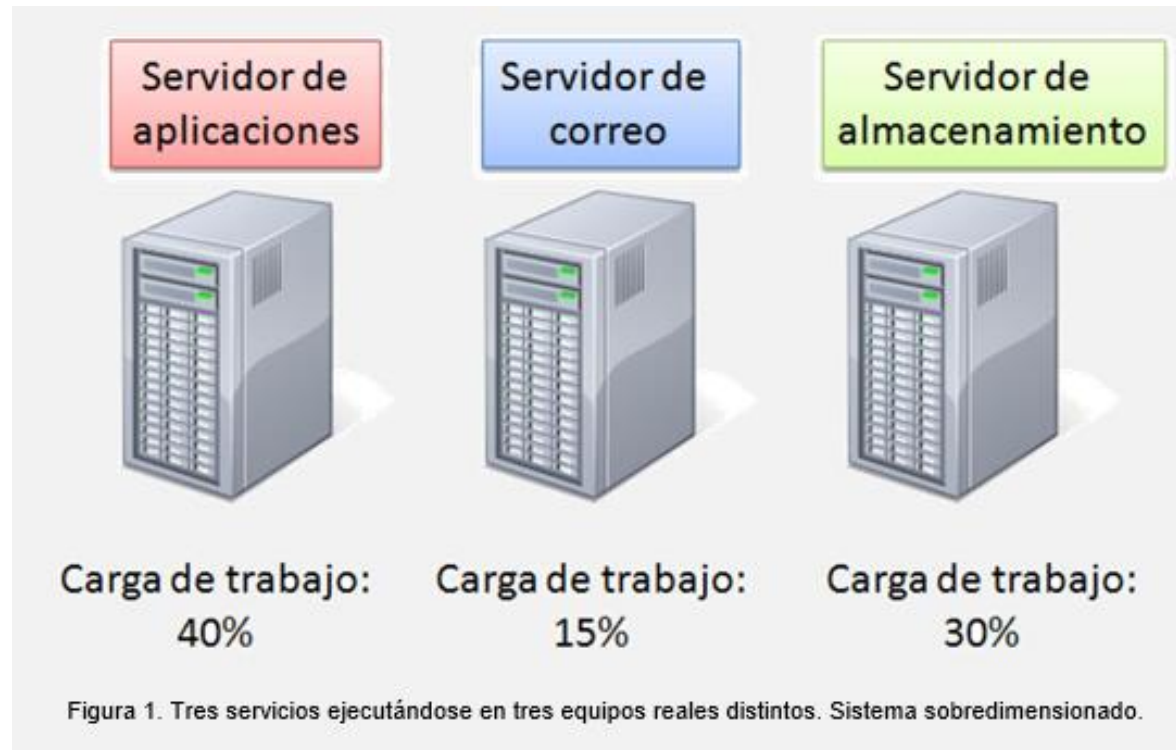
Es simula una màquina amb un Sistema Operatiu

### Avantatges de la virtualització

- Permet **executar diferents sistemes operatius simultàniament** sobre un únic hardware.
- Permet crear **instantànies**, les quals guarden un estat definit de la màquina i pot tornar a ell en cas que alguna modificació feta sobre el sistema guesst hagi causat algun dany en aquest.
- **En entorns de producció, on el hardware és molt potent, aprofiten la capacitat de l'equip estalviant costos**, Ja que no és el mateix tindre tres servidors físics dedicats, per exemple, un com a servidor de correu, un altre com a servidor d'aplicacions i un altre com a servidor d'emmagatzematge (figura 1) on probablement cadascú no aprofita més del 30% o 40% de les possibilitats del hardware, o tindre els tres servidors virtualitzats sobre un únic equip físic explotant el màxim la potència d'aquest equip real (figura 2).

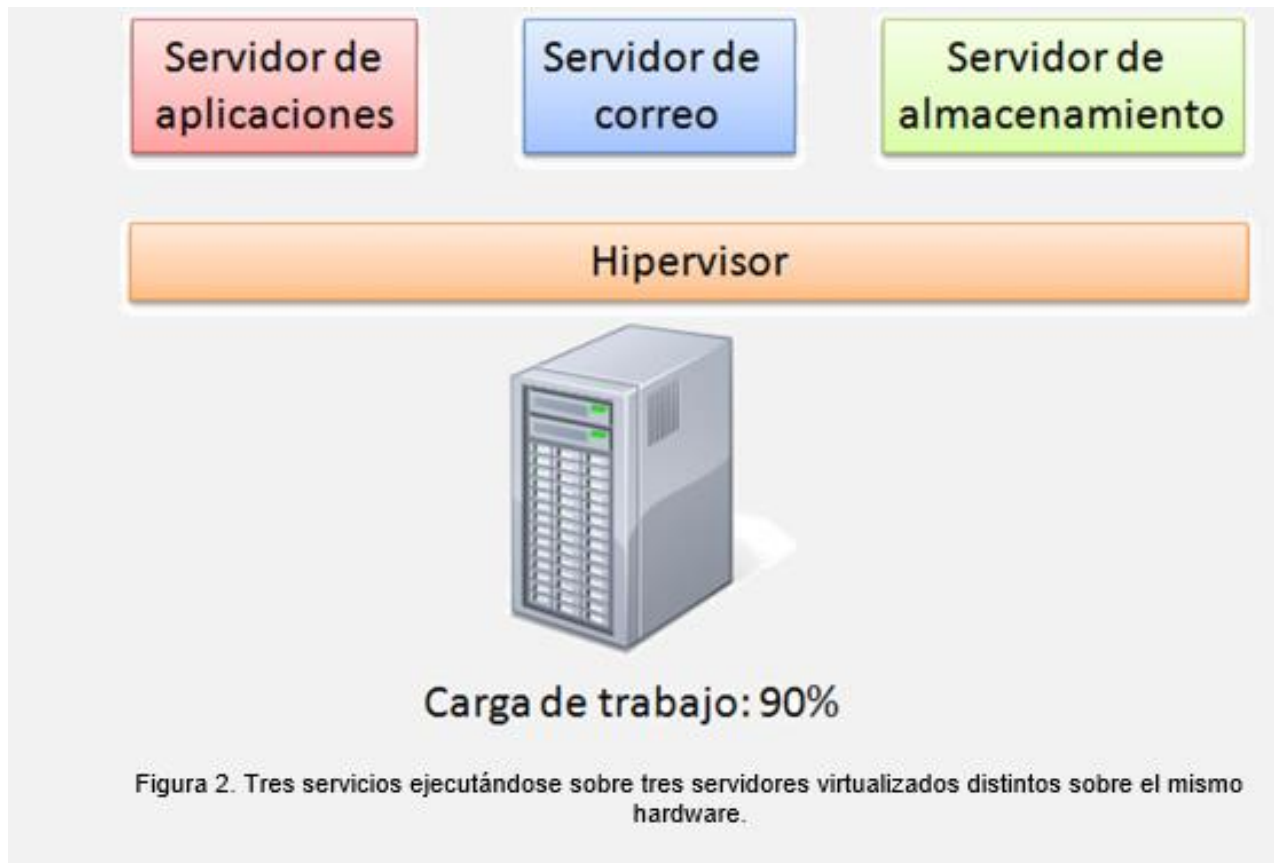
# UNITAT 1. Introducció als Sistemes Informàtics

---



# UNITAT 1. Introducció als Sistemes Informàtics

---





## UNITAT 1. Introducció als Sistemes Informàtics

---

- Les aplicacions executades en un sistema operatiu guest (o convidat) es troben aïllades de sistema operatiu host (o amfitrió), de manera que davant un atac per virus o algun altre malware, el sistema real està fora de perill, podent recuperar-se el sistema operatiu guest a partir d'una instantània 'sana'.
- Els sistemes virtualitzats poden ser 'portats' a un altre equip físic d'una manera molt senzilla. De fet, en entorns de alta disponibilitat es creen clústers de virtualització per poder executar les màquines virtuals en un equip físic o un altre en cas que hi hagi incidències en el hardware.

# Inconvenients de la virtualització

## UNITAT 1. Introducció als Sistemes Informàtics

---

- La complexitat afegida a l'execució de sistema operatiu convidat, ja que hi ha capes intermèdies fins arribar a el maquinari.
- La pèrdua de prestacions ocasionada per aquestes capes intermèdies entre el sistema convidat i el maquinari, a més de les limitacions imposades per la compartició dels recursos maquinari entre diferents sistemes operatius. No obstant això, els esquemes de virtualització baremetal 'Alleugereixen' aquestes capes intermèdies fins a valors que a penes alteren el rendiment de sistema

## VirtualBox

Per crear una nova màquina virtual premem en el botó "Nova":



Donem un nom a la màquina i seleccionem el Sistema Operatiu:

?

×

← Crear máquina virtual

### Nombre y sistema operativo

Seleccione un nombre descriptivo y una carpeta destino para la nueva máquina virtual y seleccione el tipo de sistema operativo que tiene intención de instalar en ella. El nombre que seleccione será usado por VirtualBox para identificar esta máquina.

Nombre:

Carpeta de máquina:

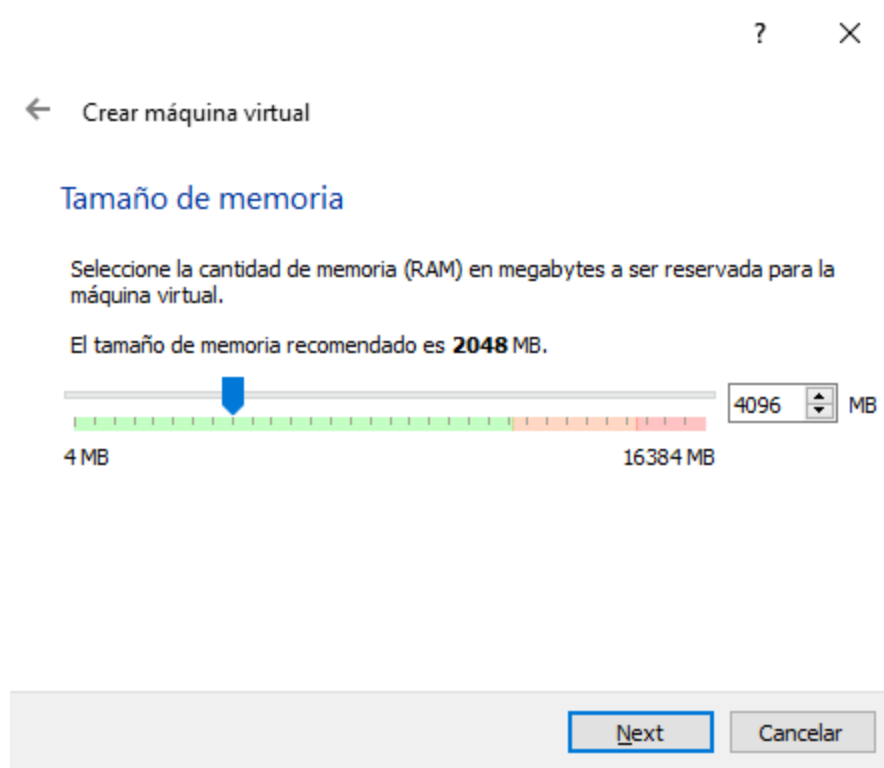
Tipo:

Versión:

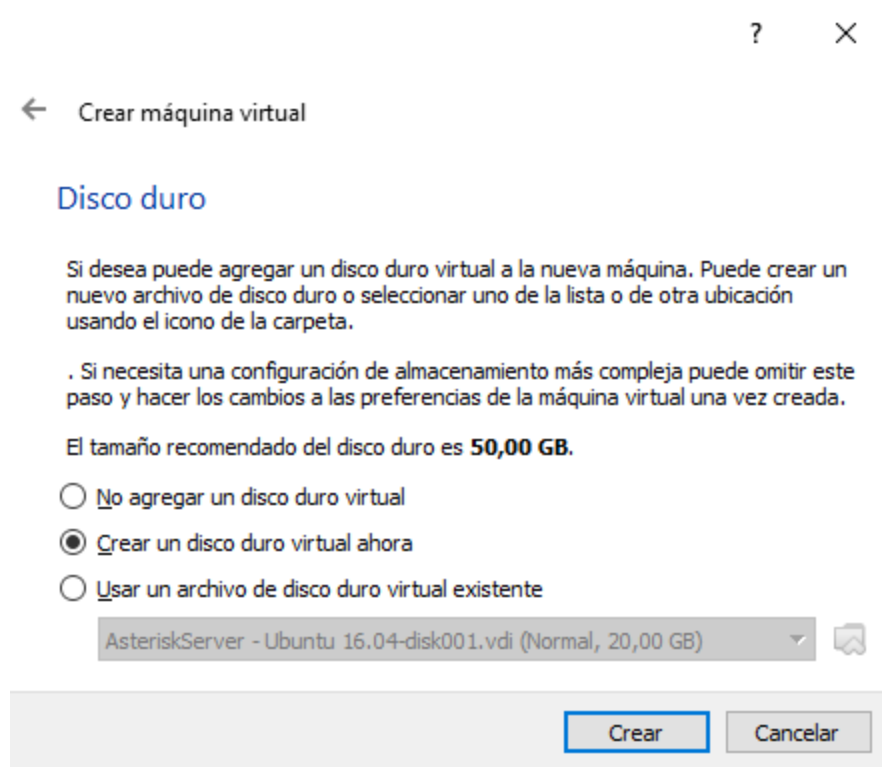
64 10

Modo experto Next Cancelar

# Seleccionem la quantitat de Memòria RAM:

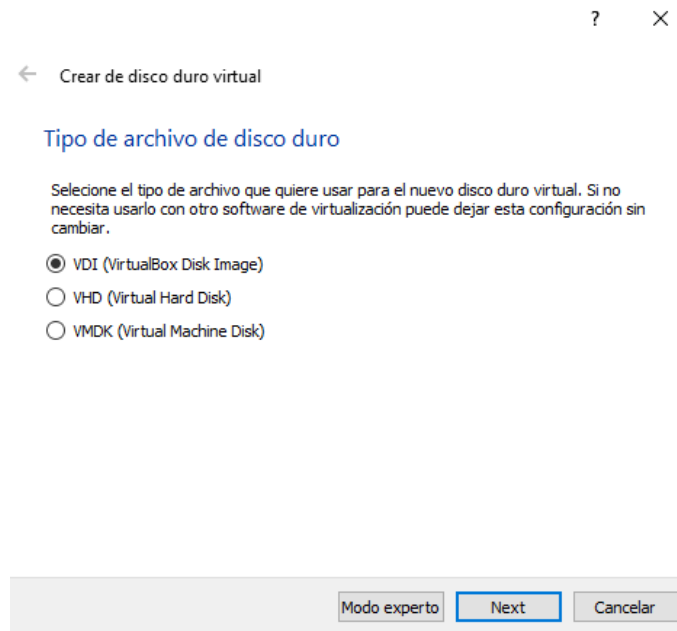


Triem el disc dur, si és la primera instal·lació escollirem "Crear un disc dur virtual ara":



## UNITAT 1. Introducció als Sistemes Informàtics

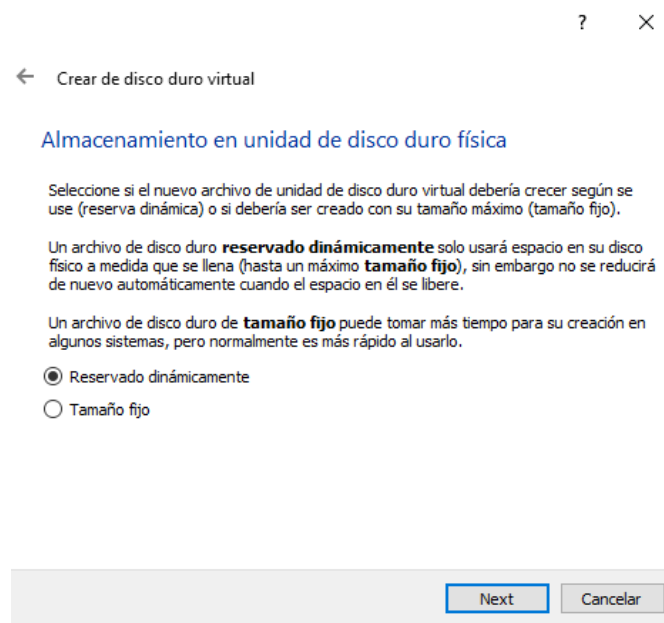
- Triem el tipus d'arxiu que allotjarà el disc dur virtual. Si només anem a usar-lo en VirtualBox millor seleccionar VDI perquè està més optimitzat però si després ho volem utilitzar en altres programes de virtualització haurem d'escollir VMDK.



# UNITAT 1. Introducció als Sistemes Informàtics

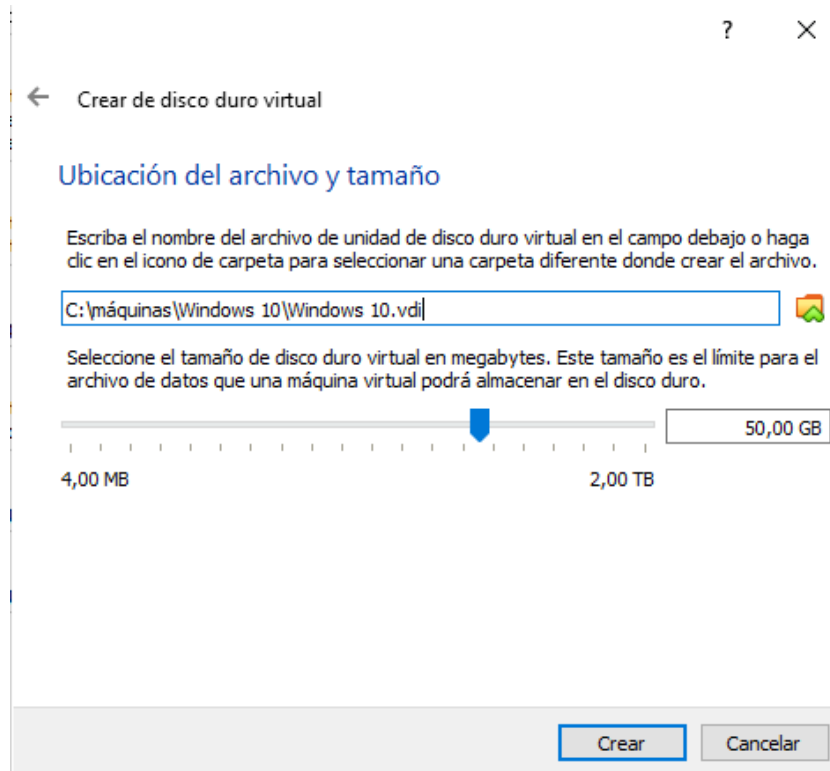
## Triem si el disc dur és dinàmic o fix:

- Dinàmic: L'arxiu que el representa en l'amfitrió ocuparà una mida inicial mínim i anirà incrementant aquest a mesura que es van introduint dades al disc dur virtual.
- Fix: L'arxiu que el representa en l'amfitrió ocuparà tot l'espai del disc dur des del principi però anirà més fluid.



## UNITAT 1. Introducció als Sistemes Informàtics

Triem arxiu que allotja el disc dur i la mida d'aquest disc



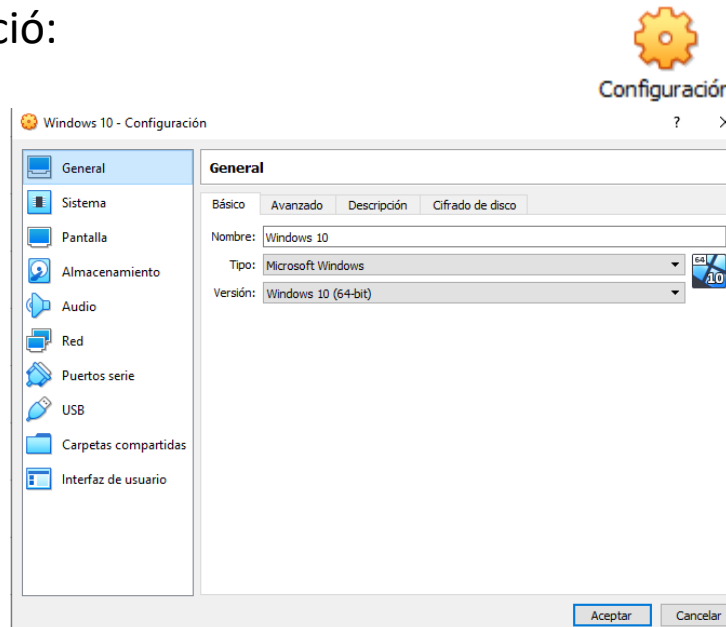


## UNITAT 1. Introducció als Sistemes Informàtics

Un cop creada la màquina apareixerà al panell esquerre de l'VirtualBox:



Per acabar de configurar-la, la seleccionem al panell esquerre i premem en el botó de configuració:

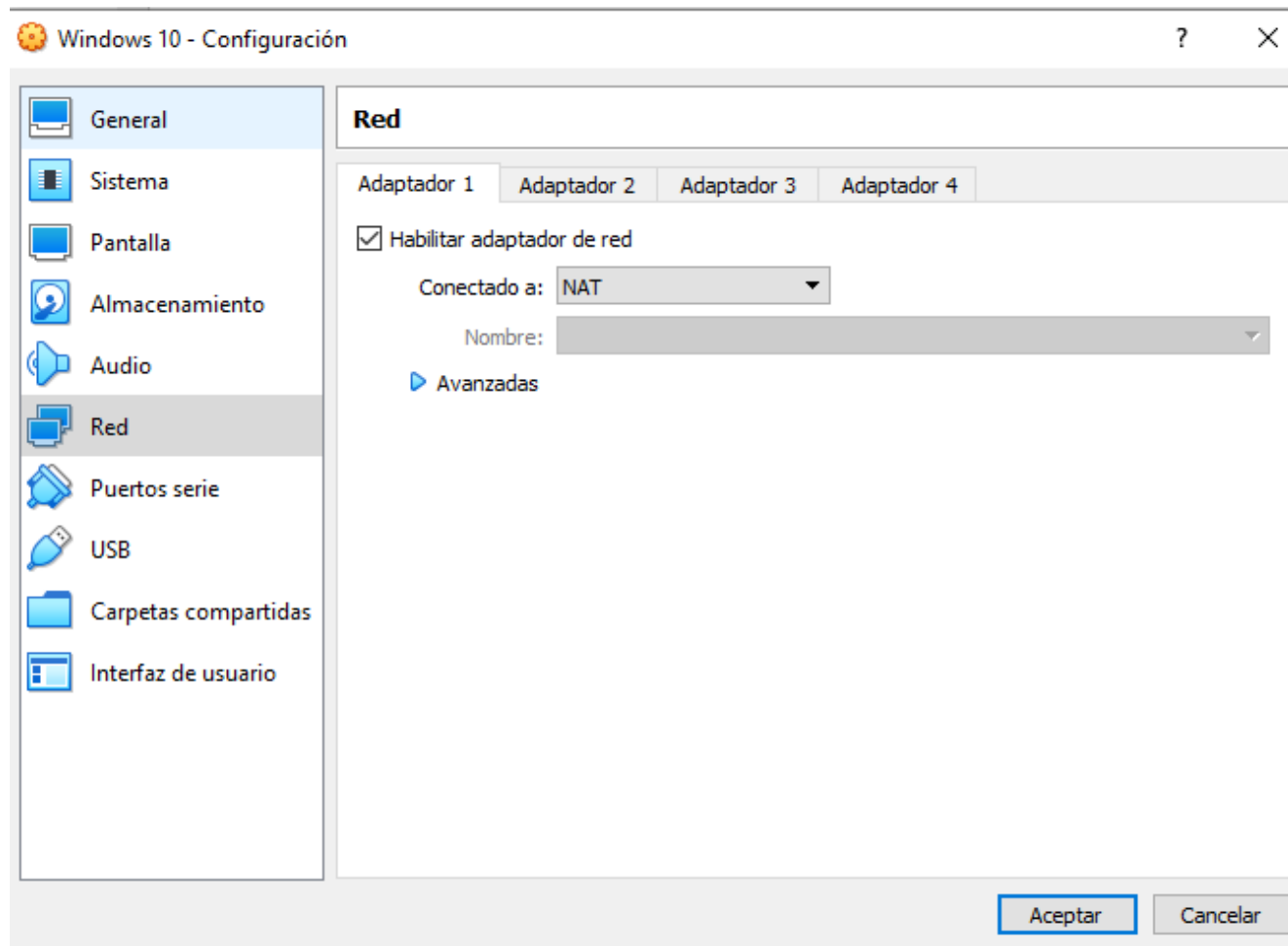


# Configuració adaptador de Xarxa

Escollim l'adaptador de xarxa i el tipus. Els tres principals són:

- Xarxa interna: L'ordinador es podrà comunicar amb la resta de màquines en el mateix equip amfitrió però no pot veure equips fora d'aquest amfitrió i per tant no pot accedir a Internet.
- NAT: Adquireix adreçament privat. Els equips es comuniquen entre si i poden accedir a l'exterior de l'amfitrió (tenen accés a Internet) però no es veuen directament com un equip independent des de fora de la xarxa.
- Adaptador pont: L'equip funciona com un equip independent a l'amfitrió i per tant des de fora de l'equip amfitrió es veu com un ordinador més. Té accés a Internet.

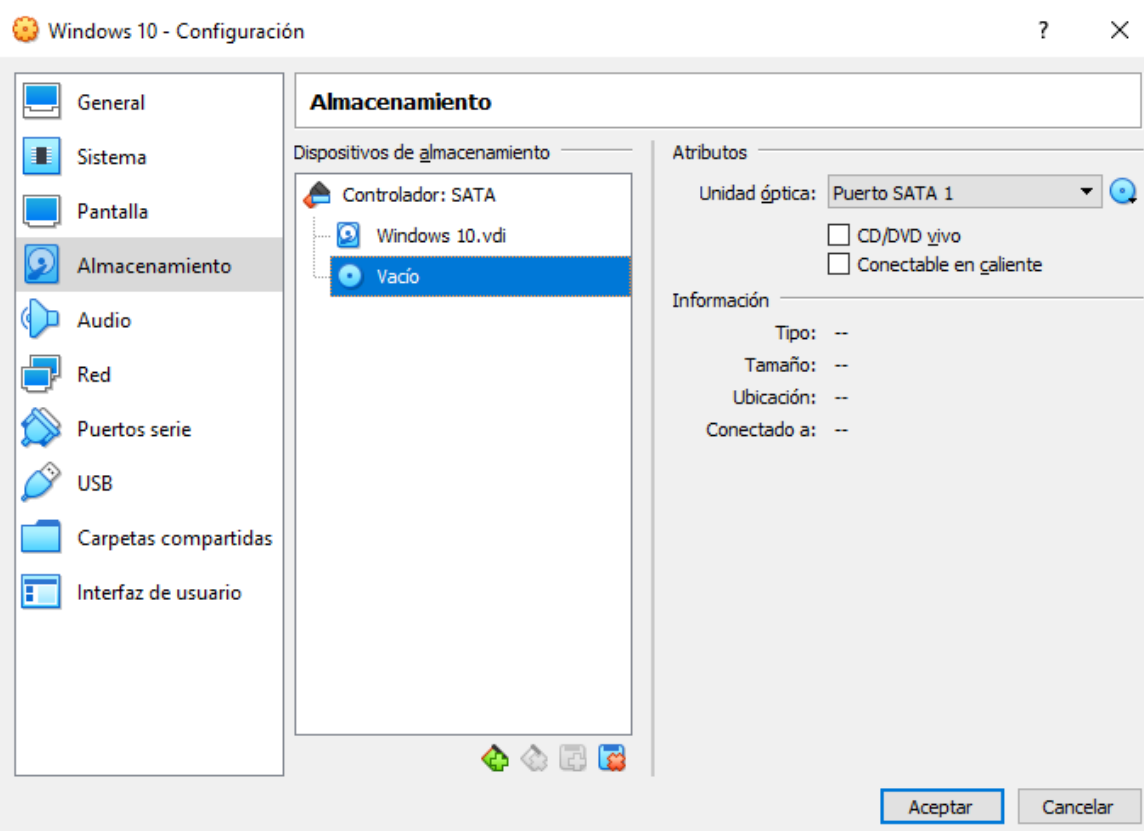
# UNITAT 1. Introducció als Sistemes Informàtics



## UNITAT 1. Introducció als Sistemes Informàtics

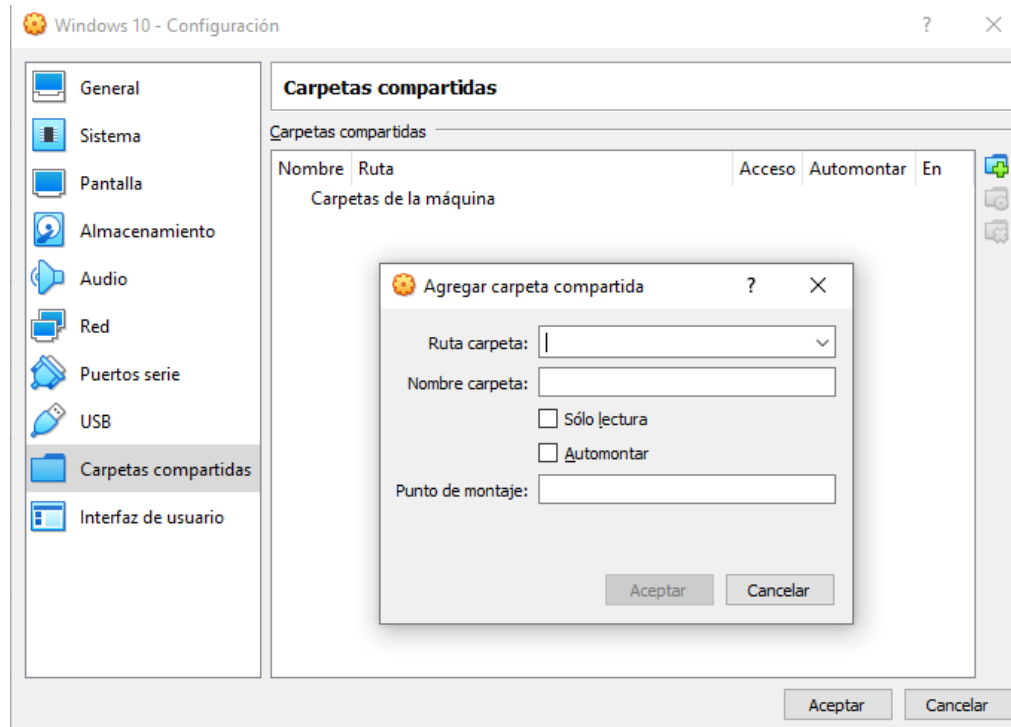
En l'apartat emmagatzematge podem inserir més discs durs o inserir unitats òptiques.

A les unitats òptiques podem utilitzar la de l'amfitrió o virtualitzarla seleccionant un arxiu ISO.



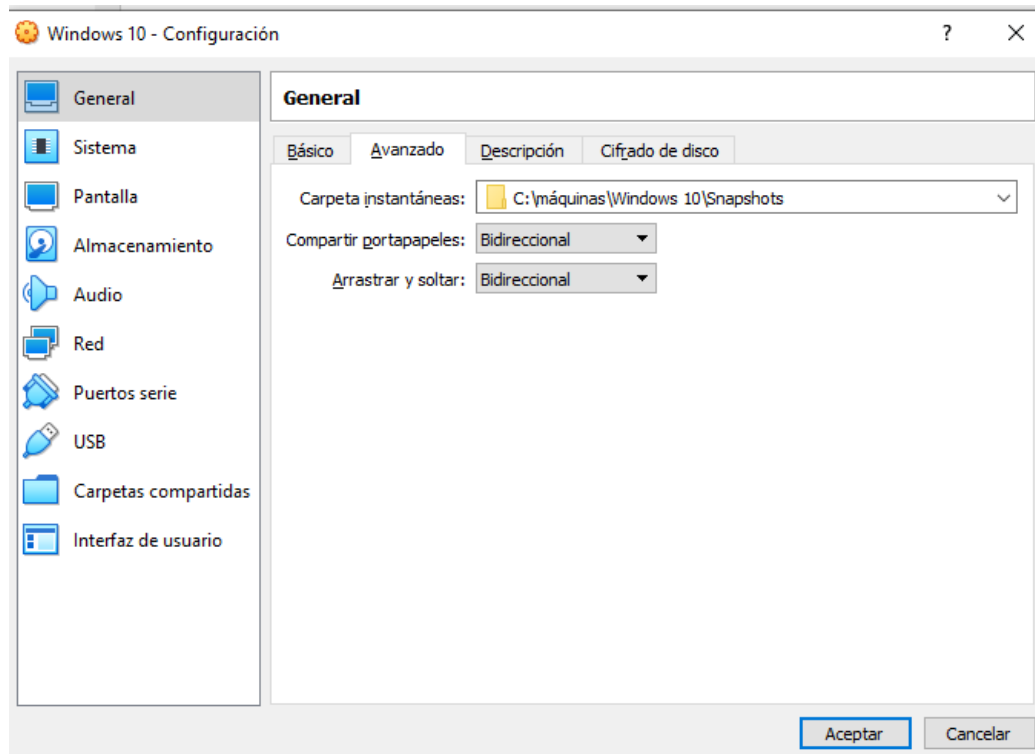
## Carpets compartides

Les carpetes compartides permet compartir un espai d'emmagatzematge entre l'equip amfitrió i la màquina virtual a través d'una carpeta existent en l'equip amfitrió.



# Copiar arxius entre amfitrió i màquina

Per poder copiar aquests arxius cal marcar des de l'apartat "General" i pestanya "Avançat" tant en l'apartat "compartir porta-retalls" com en el de "arrossegar i deixar anar" l'opció bidireccional.



### 6. Seguretat informàtica

#### Principals objectius de la seguretat informàtica:

**Detectar els possibles problemes i amenaces a la seguretat, minimitzant i gestionant els riscos.**

**Garantir l'adequada utilització dels recursos i de les aplicacions dels sistemes.**

**Limitar les pèrdues i aconseguir l'adequada recuperació de sistema en cas d'un incident de seguretat.**

**Complir amb el marc legal i amb els requisits imposats a nivell organitzatiu.**

### 2. Fiabilitat, confidencialitat, integritat i disponibilitat

#### Què és la seguretat?

**Un conjunt de tècniques encaminades a obtenir alts nivells de seguretat en els sistemes informàtics.**

**La seguretat absoluta no és possible.**

Per tant parlarem de **probabilitat**:

probabilitat que un sistema es comporti tal com s'espera d'ell

**Parlarem de sistemes fiables en lloc de sistemes segurs.**



**Un sistema fiable consisteix a garantir 3 aspectes:**

- **Confidencialitat**
- **Integritat**
- **Disponibilitat**

## CONFIDENCIALITAT

**És la qualitat d'una informació perquè només pugui ser comprensible pels usuaris autoritzats.**

**Per exemple en un àmbit d'historials o informació estratègica d'empreses és molt important que aquesta informació no pugui ser comprensible per personal no autoritzat encara que puguin arribar accés.**

# INTEGRITAT

**Qualitat de la informació que no pugui ser manipulada per usuaris no autoritzats.**

**Per exemple en un àmbit bancari seria molt important que un usuari no autoritzat fos capaç de manipular les dades dels comptes dels clients.**

## DISPONIBILITAT

**És la qualitat que una informació o servei estigui disponible el major temps possible.**

**S'associa sobretot a serveis crítics que requereixen d'estar actius constantment com pot ser un control aeri.**

decideixin

## **Grup de característiques i objectius de la seguretat.**



**Sense el nivell inferior no es pot aplicar l'exterior.**

### Autenticació i No repudi

**Autenticació:** Es tracta de verificar que un usuari és qui diu ser, generalment es fa a través d'un nom d'usuari i un password, encara que existeixen altres mètodes.

**No repudi:** Es tracta de no poder negar una cosa que s'ha realitzat o rebut i es divideix en:

No repudi en origen: l'usuari no pot negar que ha enviat alguna cosa o realitzat alguna cosa.

No repudi en destí: l'usuari no pot negar que ha llegit o rebut alguna cosa.

## Seguretat física

Les mesures de seguretat física tracten de protegir els actius tangibles i físics de l'organització, així com a les persones. Així doncs, en la seguretat física podem tenir mesures per protegir-nos o minimitzar l'impacte davant les següents amenaces:

- Incendis: mobiliari ignífug, murs i parets tallafocs, extintors ...
- Inundacions: evitar ubicació a plantes baixes, impermeabilitzar parets i sostres ...
- Robatoris: Cambres de seguretat, vigilants, codis de seguretat, sistemes biomètrics ...
- Senyals electromagnètiques: evitar ubicació a lloc amb gran radiació, protegir de les emissions mitjançant filtres o cablejat especial ...
- Apagades: SAI, estabilitzadors, grups electrògens.
- Sobrecàrregues elèctriques: SAI professionals amb filtres per evitar pics de tensió.
- Desastres naturals: ubicació, CPD s de respatller ...

## SAIs

Pel que fa a la protecció de els sistemes informàtics és convenient disposar de sistemes de alimentació ininterrompuda (SAI o UPS)

El SAI o UPS millora la qualitat del subministrament elèctric minimitzant problemes com:

- Petits talls de energia
- Sobretensió o disminucions de tensió
- Soroll elèctric
- Distorsió harmònica

La potència del SAI es mesura en VA (voltampere). Generalment, encara que també és habitual en W (watts) i cal seleccionar un SAI que permeti suportar la càrrega (Nombre d'equips informàtics) que li anem a connectar. Per exemple, un SAI de 1500W, no pot suportar teòricament la connexió de 4 servidors si cadascú d'ells té una potència de 500W a la seva font d'alimentació.

També cal indicar que la funció d'un SAI no és proporcionar alimentació continua quan cau el subministrament, sinó permetre un apagat correcte dels equips quan queda poca bateria. Per proporcionar energia permanent s'utilitzen generadors o grups electrògens, però són més habituals a instal·lacions crítiques com hospitals, data centers, etc.



# UNITAT 1. Introducció als Sistemes Informàtics

---

Existeixen diversos tipus de SAI:

- Standby (Offline): En funcionament normal, la càrrega crítica s'alimenta directament de la tensió de la xarxa. Quan hi ha una fallada de subministrament, aquest passa a funcionar amb els seus bateries. Es fa servir per a entorns domèstics. Preu baix.
- Interactiu (Inline): Proporciona alimentació de la seva bateria quan l'energia de la xarxa baixa, fent funcions de regulador de voltatge (AVR). Té un temps de commutació a bateria menor que els offline. Preu mitjà.
- On-line de doble conversió: Contínuament alimenta els equips connectats mitjançant la tensió que proporciona l'inversor, totalment aïllada de la tensió de la xarxa. Es diu de doble conversió perquè converteix de corrent altern (AC) a corrent contínua (CC) i després a corrent altern de nou. Són els que s'utilitzen habitualment als CPD i són els més cars.

## Seguretat lògica

La seguretat lògica protegeix els actius intangibles de l'organització, com són el software, les dades o els processos, així com dels accessos no autoritzats als sistemes informàtics. Per tant, a la seguretat lògica podem tenir mesures per protegir-nos o minimitzar l'impacte davant les següents amenaces:

- Robatoris: Xifrar l'informació emmagatzemada, utilitzar contrasenyes, sistemes biomètrics ...
- Pèrdua d'informació: Còpies de seguretat, sistemes tolerants a fallades, discos redundants RAID ...
- Pèrdua d'integritat: Programes de revisió de l'equip, signatura digital, eines d'integritat ...
- Software maliciós: Antimalware (Antivirus, Antirootkit, etc.)
- Atacs en xarxa: Firewall, programes de monitoratge, servidors proxys, detectors d'intrusos ...
- Accessos no autoritzats: Contrasenyes, llistes de control d'accés, xifrar documents ...

# Funcions Hash

Són funcions resums per comprovar la integritat d'un arxiu.

Es a dir es crea un codi en funció del contingut d'un arxiu de tal forma que si se modifica l'arxiu el codi resultant d'aplicar la funció serà distinta i per tant es detectarà aquesta modificació.

Antigament s'utilitzava md5 però a causa de la longitud de la contrasenya i a l'avanç de la potència computacional aquest algoritme ha deixat de ser segur, a l'igual que sha1. Actualment podem utilitzar sha128, sha256, sha3, ...

# Funcions Hash

- Provar el sha1 de linux amb qualsevol arxiu i observar que ens retorna el sha256 calculat.

`sha256sum arxiu`

- Crear un arxiu amb el codi sha256:

`sha256sum arxiu> archivohash.sha`

- Comprovar si el fitxer ha estat modificat:

`sha256sum -c archivohash.sha`

- Modificar l'arxiu original i tornar a fer una comparació.
- Per a la resta de versions és similar

# Encriptació

El xifrat d'arxius de dades, ens permet garantir la confidencialitat dels mateixos, ja que només podran accedir a les dades aquells usuaris que puguin desxifrar el missatge utilitzant el mateix algoritme de xifrat i la clau de desxifrat.

Hi ha criptografia simètrica, asimètrica i híbrida.

Criptografia simètrica:

S'utilitza una mateixa clau per xifrar i desxifrar missatges.

Les dues parts de la comunicació s'han de posar **d'acord per endavant sobre la clau a utilitzar.**

Un bon sistema de xifrat **posa tota la seguretat en la clau i cap en l'algoritme**

## Principals problemes dels sistemes de xifrat simètrics

### **L'intercanvi de claus: com es realitza de forma segura?**

En una comunicació hauria de compartir la clau en text pla abans de començar a xifrar i si algú l'intercepta podria desxifrar tota la comunicació.

### **El nombre de claus que es necessiten.**

Cada comunicació hauria de tenir una clau diferent.

**Per solucionar aquests problemes es fa servir **criptografia asimètrica** i **criptografia híbrida**.**

### PRÀCTICA XIFRAT SIMÈTRIC

#### **PGP (Pretty good Privacy):**

**Programa més popular d'encryptació i de creació de claus públiques i privades**

#### **GPG (GNU Privacy guard):**

Igual que l'PGP però de software lliure, utilitzant algoritmes no registrats per patents com l'IDEA.

Utilitza algoritmes no patentats com ElGamal, CAST5, 3DES, AES i Blowfish.

Ve preinstal·lat en Linux però hi ha versions gratuïtes per a Windows.

### PRÀCTICA XIFRAT SIMÈTRIC II

A Linux la comanda a executar és:

```
gpg -c archivoorigen
```

Ens demanarà una clau.

Crea un arxiu com l'original afegint l'extensió GPP.

L'arxiu de sortida és binari.

Si es tracta d'un arxiu de text ASCII cal afegir l'extensió -a, donant un arxiu amb extensió asc

Per desxifrar l'arxiu:

```
gpg -d archivo.asc
```



# UNITAT 1. Introducció als Sistemes Informàtics

---

```
miguel@miguel-virtual-machine: ~
miguel@miguel-virtual-machine:~$ cat prueba
Esto solo es un texto de prueba
miguel@miguel-virtual-machine:~$ gpg -c -a prueba
miguel@miguel-virtual-machine:~$ cat prueba.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.11 (GNU/Linux)

jA0EAWMCFIe2CUE1DEBgyTcxBcKBi9lbhc8Ttpsi8NVFBSq+gTeaUHdUEQcI4u9T
tkPusgWwE8Dgv0vaW1VHSP1wvQLavNgV
=TX2U
-----END PGP MESSAGE-----
miguel@miguel-virtual-machine:~$ rm prueba
rm: ¿borrar el fichero regular «prueba» protegido contra escritura? (s/n) s
miguel@miguel-virtual-machine:~$ gpg -d prueba.asc
gpg: anillo «/home/miguel/.gnupg/secring.gpg» creado
gpg: datos cifrados CAST5
gpg: cifrado con 1 frase contraseña
Esto solo es un texto de prueba
gpg: AVISO: la integridad del mensaje no está protegida
miguel@miguel-virtual-machine:~$ cat prueba
cat: prueba: No existe el archivo o el directorio
miguel@miguel-virtual-machine:~$
```

# UNITAT 1. Introducció als Sistemes Informàtics

---

## Criptografia de clau asimètrica

Cada usuari de sistema criptogràfic ha de posseir una parella de claus:

**Clau privada:** Serà custodiada pel seu propietari i no es donarà a conèixer a cap altre.

**Clau pública:** Serà coneguda per tots els usuaris.

El que xifra una solament ho pot desxifrar l'altra i viceversa

És impossible conèixer una clau a partir de l'altra

Basades en funcions hash d'un sol sentit

Alguns algoritmes emprats com a funció resum o hash són MD5 i SHA.

## Xifrat simètric vs de clau pública

Davant d'un atac de força bruta el xifrat de clau pública és menys segur que el xifrat simètric.

### Avantatges de clau asimètrica:

Es xifra amb una clau i es desxifra amb una altra.

### Desavantatges de l'xifrat asimètric (De clau pública):

Major temps de procés.

La clau ha de ser més gran.

El missatge xifrat ocupa més espai.

### Criptografia híbrida

S'utilitza un algoritme de clau pública per al xifrat en l'enviament d'una clau simètrica.

A partir d'ací es fa servir clau simètrica.

Amb això **anem a aconseguir:**

- **Confidencialitat**
- **Integritat**

### PRÀCTICA XIFRAT ASIMÈTRIC

Utilitzarem gpg mitjançant `gpg --gen-key` a Linux.

Se'ns van a anar fent preguntes per a la construcció de l'USERID:

- Tipus de claus
- Mida de les claus
- Temps de validesa
- Passphrase

En la generació seria bo treballar amb l'ordinador perquè sigui el més aleatori possible.

### PRÀCTICA XIFRAT ASIMÈTRIC II

Les claus públiques es guarden en el fitxer publing.gpg i les privades en el fitxer secring.gpg.

Per veure les claus públiques disponibles: `gpg -list-keys` o `gpg -k`.

L'identificador de clau (ClaveID) és el nombre que apareix a continuació de 2048D.

Per a veure les claus privades: `gpg -list-secret-keys`.

Per esborrar les claus primer cal esborrar la privada i després la pública:

Eliminar clau privada: `gpg -delete-secret-key ClaveID`

Eliminar clau pública: `gpg -delete-key ClaveID`

## PRÀCTICA XIFRAT ASIMÈTRIC III Còpia i distribució de claus

Es pot realitzar de diverses formes:

- Pujant-se a un servidor de claus públiques:  
`gpg -send-keys -keyserver pgp.rediris.es ClaveID`
  - Per fer una recerca de claus en un ordinador:  
`gpg -keyserver nomdelservidor -search-keys ClaveID`
  - Per baixar-nos la clau:  
`gpg -keyserver nomdelservidor -recv-keys ClaveID`
- Enviant-les per correu o donant-la en un suport portable mitjançant un fitxer.  
`gpg -armor -output ficherodeclave -export ClaveID`
- És important tenir una còpia a part de la nostra clau privada:  
`gpg -armor -output ficherodeclave -export-secret-key ClaveID`
- Per importar una clau bolcada:  
`gpg -import ficherodeclaves`

## PRÀCTICA XIFRAT ASIMÈTRIC III Còpia i distribució de claus

Es pot realitzar de diverses formes:

- Pujant-se a un servidor de claus públiques:  
`gpg -send-keys -keyserver pgp.rediris.es ClaveID`
  - Per fer una recerca de claus en un ordinador:  
`gpg -keyserver nomdelservidor -search-keys ClaveID`
  - Per baixar-nos la clau:  
`gpg -keyserver nomdelservidor -recv-keys ClaveID`
- Enviant-les per correu o donant-la en un suport portable mitjançant un fitxer.  
`gpg -armor -output ficherodeclave -export ClaveID`
- És important tenir una còpia a part de la nostra clau privada:  
`gpg -armor -output ficherodeclave -export-secret-key ClaveID`
- Per importar una clau bolcada:  
`gpg -import ficherodeclaves`



### PRÀCTICA XIFRAT ASIMÈTRIC IV

- Encriptar un fitxer:

`gpg --armor --recipient ClaveID --encrypt mensaje`

- Desencriptar un fitxer:

`gpg --decrypt fichero.asc`

# Seguretat activa

Per seguretat activa s'entenen aquelles mesures que prevenen i intenten evitar els danys als sistemes informàtics. Són mesures preventives.

Com a mesures o tècniques de seguretat activa a seguretat informàtica podem citar els següents exemples:

- Clau: accés al sistema o aplicacions per part de persones no autoritzades.
- Permisos a fitxers: accés a fitxers per part de personal no autoritzat.
- Xifrat: accessos no autoritzats a informació confidencial.
- Antivirus: virus informàtics i en general, aplicacions malicioses
- Certificats digitals: atacs a la integritat i autenticitat de els dades.
- Quotes de disc: Fallada de l' sistema per esgotament de l'espai a disc.

Entre les mesures de seguretat activa que podem aplicar, hi ha moltes com poden ser:

- Protegir la BIOS o UEFI amb contrasenya
- Gestió d'usuaris i política de contrasenyes
- Autenticació de múltiples factors
- Gestió de permisos
- Quotes de disc

### Seguretat passiva

Per seguretat passiva, s'entenen aquelles mesures que s'apliquen després d'ocórrer un incident de seguretat i intenta minimitzar l'impacte del mateix. Són mesures correctores o pal·liatives.

Entre les mesures de seguretat passiva que podem aplicar, podem citar:

- Còpies de seguretat (restauració).
- Recuperació de dades en cas d'incident.
- Neteja de malware.

## Còpies de seguretat

En funció de la quantitat d'arxius que es salvaguarden a l'hora de fer la còpia de seguretat, podem distingir 3 tipus de còpia:

Còpia de seguretat total o íntegra: És una còpia de seguretat completa de tots els arxius i directoris seleccionats

Còpia de seguretat incremental: És una còpia de seguretat només dels arxius que han canviat des de l'última còpia de seguretat realitzada. Si fem còpia de seguretat total el dia 1 de cada mes i còpia de seguretat incremental la resta dels dies, cada còpia incremental només guardarà els arxius que s'hagin modificat aquest dia. Si hem de realitzar la restauració d'arxius davant un desastre, hem de disposar de la còpia total i de totes les còpies incrementals que hàgim realitzat des de la còpia total.

Còpia de seguretat diferencial: És una còpia de tots els arxius que han canviat des de l'última còpia de seguretat total que hàgim fet. Si fem còpia de seguretat total el dia 1 de cada mes i còpia de seguretat diferencial la resta dels dies, cada còpia diferencial guardarà els arxius que s'hagin modificat des de el dia 1. L'avantatge és que es requereix menys espai que la còpia total i que a el procés de restauració únicament necessitem l'última còpia total i l'última còpia diferencial. Per contra, es consumeix més temps a realitzar la còpia i també més espai que al cas de còpia incremental.

Recomanacions sobre el tipus de còpia de seguretat a utilitzar:

- Si el volum no és molt elevat:

Còpies totals

- Si el volum és molt elevat però el volum de dades que es modifiquen no és molt elevat:

Còpies diferencials

- Si el volum de dades és molt elevat i el volum de dades que es modifica també ho és:

Còpies incrementals

# RAID

Un RAID (Redundant Array of Independent disks, abans Inexpensive disks) és un sistema d'emmagatzematge que utilitza múltiples discos durs o SSD entre els quals es distribueixen o repliquen les dades per proporcionar tolerància a fallades. Té diverses implementacions i en funció de l'ús és més convenient una o altra.

Es poden realitzar per hardware o per software, sent per hardware l'opció més recomanada per rendiment. algunes controladores RAID de servidors permeten canviar els discos danyats a calenta sense apagar el servidor.

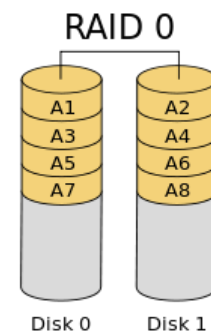
En els sistemes RAID de servidors és comú parlar del disc hot spare, Que és un disc de reserva que automàticament forma part activa del RAID si falla un disc, sense necessitat de intervenir l'administrador.

# UNITAT 1. Introducció als Sistemes Informàtics

## Alguns de els tipus RAID més usats.

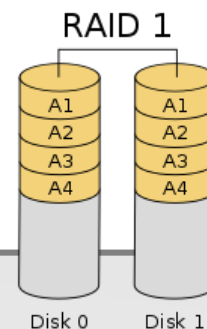
### RAID 0:

- Volum de dades lògic a partir de discos físics (es coneix també com stripping)
- No hi ha redundància de dades ni tolerància a errors: si falla un disc, perdem dades (llevat que tinguem còpia, òbviament)
- Millora el rendiment sobretot a arxius grans (ex: edició de vídeo)
- Cada bloc es va escrivint a un disc alternativament
- S'utilitzen 2 o més discos



### RAID 1:

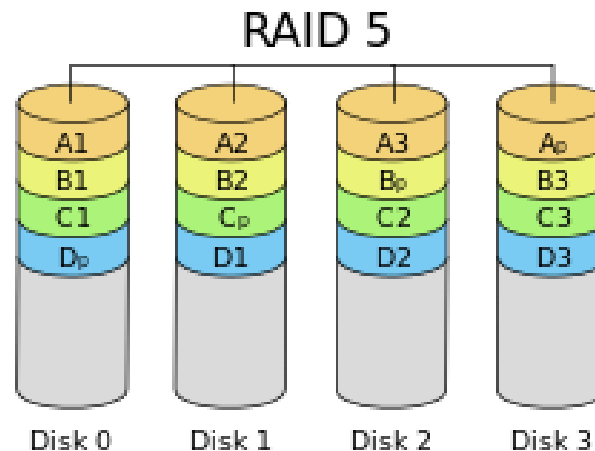
- Consisteix a una còpia exacta d'un disc físic a un altre (també es coneix com com mirall o mirroring)
- Hi ha tolerància a errors i es perd la mida d'un disc (50%)
- Millora el rendiment de lectura
- Es forma amb 2 discos





### RAID 5:

- Divideix les dades a blocs i fa servir paritat distribuïda
- La paritat permet recuperar les dades si es perd un disc
- Hi ha tolerància a errors i es perd la mida d'un disc (33% amb 3 discos)
- Mal rendiment amb escriptures petites
- Necessita mínim 3 discos



## UNITAT 1. Introducció als Sistemes Informàtics

A més existeixen els RAID niats, Que combinen diversos tipus de RAID com el 10 o el 0 +1. En tots dos casos es necessita 4 discos mínim i han de ser parells. En el cas de el 10 primer es fa el mirroring de dades cada dos discos, i després el stripping sobre els dos RAID1. Al 0 + 1 és a revés: primer stripping i després mirroring:

