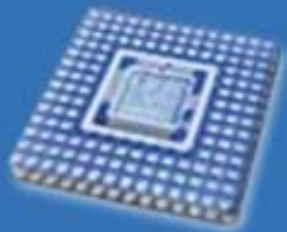


Informática
E.P.S. Universidad de Sevilla

TEMA 1:

INTRODUCCIÓN A LA INFORMÁTICA



Departamento de
Arquitectura y
Tecnología de Computadores
UNIVERSIDAD DE SEVILLA



- 1. Representación de la Información**
- 2. Estructura de Computadores**
- 3. Redes de Ordenadores**

1. REPRESENTACIÓN DE LA INFORMACIÓN

¿Cómo se almacenan en un computador?

1.1.- Números

1.2.- Texto

1.1 NÚMEROS:

■ Sistema de numeración

- Conjunto de símbolos y reglas de generación que permiten construir todos los números válidos.

■ Pueden clasificarse en:

- **No-posicionales:** el valor del símbolo no depende de la posición que ocupan dentro del número.
 - Por ejemplo, el sistema de numeración Maya, Egipcia, Chino o Romano (I, II, III, IV, V, VI, VII, VIII, IX, X, ...)
- **Posicionales (ponderados):** el valor de un dígito depende tanto del símbolo como de la posición que ese símbolo ocupa en el número.
 - **Base de numeración:** Número de símbolos diferentes utilizados en el Sistema para representar los números.
 - **Dígito:** Cada uno de los símbolos del sistema

Sistema de numeración	Base de numeración	Nº de dígitos	Dígitos usados
Decimal	Base 10	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9

1.1 NÚMEROS:

- ¿Qué sistema de numeración usa un computador?
 - Los computadores se componen de circuitos digitales los cuales se construyen de transistores.
 - Estos transistores trabajan con dos valores de voltaje: Uno de 0 voltios (o tierra) y otro cercano al valor dado por la fuente de alimentación del circuito.
 - Estos dos voltajes representan a los dos estados o niveles lógicos y se identifican como '0' y '1'.
- Puesto que toda la información en un computador se almacena, se procesa o se transmite empleando esos dos niveles 0 y 1, su sistema de numeración natural es el sistema binario.
 - Ej: el número 987 se representa en base binaria: 001111011011
- Los números en binario son menos 'compacto' que la base decimal (requiere más dígitos para representar el mismo número), en ciencias de la computación suele emplearse también otras bases potencia de 2: Hexadecimal y la Octal(8)

Sistema de numeración	Base de numeración	Nº de dígitos	Dígitos usados	Ejemplo
Binario	Base 2	2	0, 1	001111011011 ₍₂₎
Hexadecimal	Base 16	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	3DB ₍₁₆₎
Octal	Base 8	8	0,1,2,3,4,5,6,7	1733 ₍₈₎

- Para evitar confusiones, identificaremos la base de numeración como se muestra en los ejemplos de la tabla anterior.

1.1 NÚMEROS SIN SIGNO: Binario natural

■ Base 10 (10 dígitos): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

- Cada dígito tiene un valor diferente (peso) según su posición:

...	Centenas	Decenas	Unidades
...	100	10	1
...	10^2	10^1	10^0

Más significativo ← Menos significativo

Potencias de 10

- Ejemplo: $324 = 3 \times 10^2 + 2 \times 10^1 + 4 \times 10^0 = 300 + 20 + 4$
- Operaciones aritméticas: fáciles de realizar siguiendo unas reglas

■ Base 2 (2 dígitos): 0, 1

Binario	0	1	10	11	100	101	110	111	...
Decimal	0	1	2	3	4	5	6	7	...

- Cada dígito tiene un valor diferente (peso) según su posición:

...	2^3	2^2	2^1	2^0
...	8	4	2	1

Más significativo ← Menos significativo

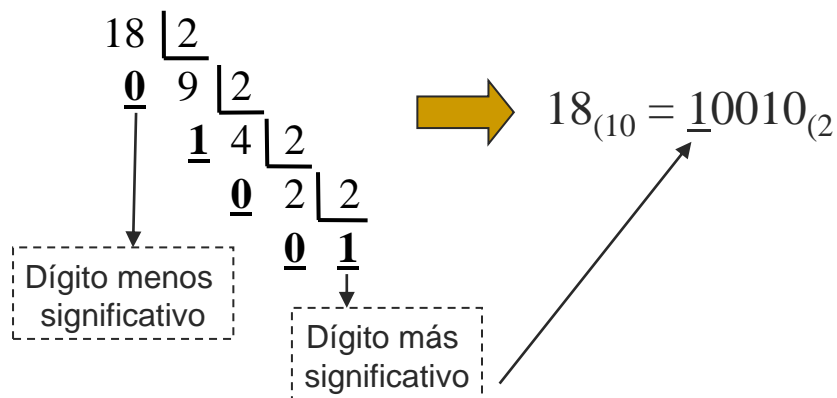
Potencias de 2

- Ejemplo (de Binario a Decimal):

$$10110_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 16 + 4 + 2 = 22_{10}$$

1.1 NÚMEROS SIN SIGNO: Binario natural

- ¿De Decimal a Binario? → divisiones sucesivas entre dos



- Suma: y Producto:

$$0 + 0 = 0$$

$$0 \times 0 = 0$$

$$0 + 1 = 1$$

$$0 \times 1 = 0$$

$$1 + 1 = 10$$

$$1 \times 1 = 1$$

- Operaciones aritméticas binarias → similares a las del sistema decimal

Acarreo →

$$\begin{array}{r}
 \\
 \\
 + 1 \\
 \hline
 1
 \end{array}$$

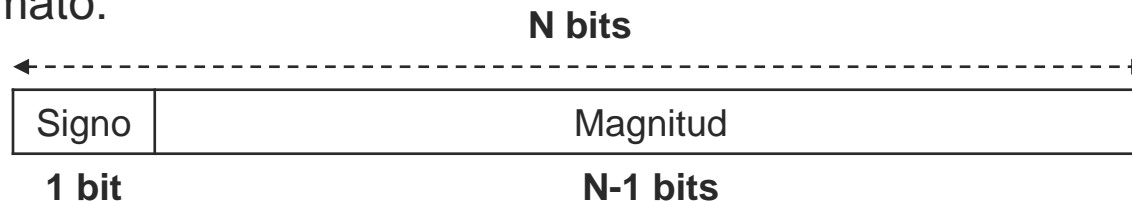
$$\begin{array}{r}
 1 \\
 \times 1 \\
 \hline
 1 \\
 1 \\
 \hline
 1 \\
 1
 \end{array}$$

1.1 NÚMEROS CON SIGNO:

- Para representar números negativos se utilizan diferentes codificaciones en binario que permiten indicar el signo del número.
 - Signo-Magnitud
 - Complemento a 2 (C-A2)
- **Bit de signo:** En binario no existen los símbolos '+' y '-' (sólo ceros y unos) por lo que se utiliza **un dígito binario para especificar el signo:**
 - '0' → Valor positivo
 - '1' → Valor negativo
- Antes de codificar el número es necesario especificar cuántos bits van a utilizarse para representarlo.
 - Ejemplo: Número en complemento A2 de 16bits.

1.1 NÚMEROS CON SIGNO: Signo- Magnitud

- Formato:



- El campo 'Magnitud' corresponde al **valor absoluto del número en binario natural**.

- Ejemplos:

-3 → 1011	-4 → 1100	-7 → 1111
+3 → 0011	+4 → 0100	+7 → 0111

- **Rango de valores:** el rango de números que se pueden representar en signo-magnitud con N bits es:

$$[-(2^{N-1}-1), 2^{N-1}-1]$$

- Por ejemplo, con 8 bits podemos representar $[-(2^7-1), 2^7-1] \rightarrow [-127, 127]$

- **Ventaja:**

- Representación intuitiva y fácil.

- **Inconvenientes:**

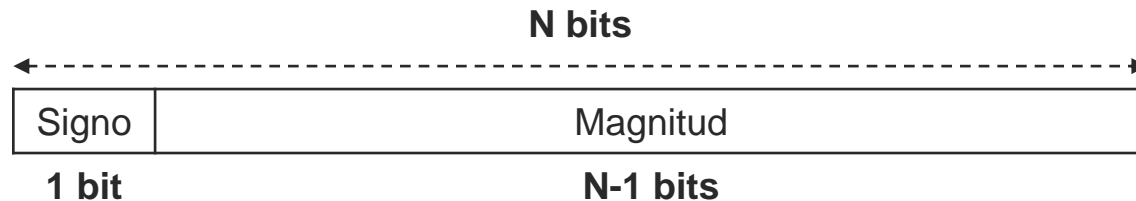
- Existen dos representaciones del 0

+0	0	00 ... 0
-0	1	00 ... 0

- El signo se trata de un modo diferente en las sumas y restas → Mayor complejidad del Hardware del computador (por ello no suele usarse).

1.1 NÚMEROS CON SIGNO: Complemento A2

- Empleado por la mayoría de los computadores para representar los números enteros con signo.
 - Ejemplo en C: `int temperatura = -5;` → 'temperatura' se guarda en Ca2 de 32 bits.



- La representación en Ca2 depende del signo:
 - **Positivo (bit de signo = 0):** Idéntico a **Signo-Magnitud**
 - **Negativo (bit de signo = 1):** Requiere una conversión
- Conversión de decimal a Ca2 para n^{os} negativos (dos procedimientos).
 - **A. Teórico mediante fórmula matemática**
 - Sea k el valor negativo que queremos convertir a Ca2 de N bits
 - Consiste en la representación en binario natural de $2^N - |k|$
 - Ejemplos:
 - 3 en C2 con 5 bits → binario natural de $2^5 - 3 = 32 - 3 = 29 \rightarrow 11101$
 - 16 en C2 con 5 bits → binario natural de $2^5 - 16 = 32 - 16 = 16 \rightarrow 10000$

1.1 NÚMEROS CON SIGNO: Complemento A2

○ B. Práctico mediante la Inversión:

- Sea k el valor negativo que queremos convertir a Ca2, utilizando N bits
- Representar $|k|$ en binario natural con N bits.
- Invertir cada bit (unos por ceros y viceversa) del número anterior.
- Sumar 1 y tomar los N bits menos significativos (no tomar el acarreo)
- Ejemplo: -3 en Ca2 con 5 bits

$$\begin{array}{r}
 -3 \rightarrow 3 \rightarrow 00011 \rightarrow 11100 \rightarrow \begin{array}{r} 11100 \\ + 1 \\ \hline 11101 \end{array} \leftarrow \text{Resultado}
 \end{array}$$

■ Conversión de Ca2 a decimal para números negativos:

- Invertir cada bit (unos por ceros y viceversa) del número en Ca2.
- Suma 1 al resultado y tomar los N bits menos significativos.
- Convertir a decimal como tratándolo como un binario natural
- Negar el número decimal obtenido.
- Ejemplo: Conversión a decimal del n° 11011

$$\begin{array}{r}
 11011 \rightarrow 00100 \rightarrow \begin{array}{r} 00100 \\ + 1 \\ \hline 00101 \end{array} \rightarrow 5 \rightarrow -5
 \end{array}$$

1.1 NÚMEROS CON SIGNO: Complemento A2

■ Rango de valores:

- Con N bits en Ca2 puede representarse números del rango:

$$[-2^{N-1}, 2^{N-1}-1]$$

- Ejemplo con 8 bits: puede representarse los valores $[-2^7, 2^7-1] \rightarrow [-128, 127]$

■ Extensión de bits:

- Para convertir el Ca2 de N bits en otro de M bits siendo $M > N$ se añaden bits de signo (**extensión de signo**) hasta que su tamaño sea de M bits.
 - **Ejemplo:** Extender a 8 bits el número '1001' en Ca2 \rightarrow se obtiene '**1111**1001'

■ Ventajas de la representación Ca2

- Una sola representación del cero (a diferencia del signo magnitud)
- Tanto las sumas como restas se realizan como sumas por lo que la complejidad del circuito es menor y su velocidad es mayor.

1.1 NÚMEROS: Hexadecimal

- En ciencias de la computación, la base hexadecimal se utiliza para representar números binarios (**de cualquier formato**) de forma compacta.
 - La sucesión de bits representada por el número en hexadecimal puede ser interpretada como cualquier codificación binaria: binario natural, signo-magnitud, Complemento A2,...

Símbolo Hexadecimal	Binario	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7

Símbolo Hexadecimal	Binario	Decimal
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

111010101 \rightarrow 1000 1101 0101 \rightarrow 8D5
8 D 5

$$\begin{array}{c}
 3982 \begin{array}{|l} 16 \\ \hline \end{array} \\
 \leftarrow \mathbf{14} \quad 248 \begin{array}{|l} 16 \\ \hline \end{array} \\
 \mathbf{8} \quad \mathbf{15} \rightarrow \mathbf{F}
 \end{array}
 \quad \xrightarrow{\text{Orange Arrow}} \quad
 3982_{(10)} = \mathbf{F8E}_{(16)}$$

$$-182_{(10)} \rightarrow 1\ 0100\ 1010_{\text{en Ca2}} \rightarrow \underbrace{1111}_F \underbrace{0100}_4 \underbrace{1010}_A_{\text{en Ca2}} \rightarrow \text{F4A}_{(16)}$$

1.1 Números: Unidades de medida

- Debido al elevado número de bits que almacenan, procesan o transmiten los sistemas digitales se emplean múltiplos para cuantificar la información.
- **Byte (Octeto):** es el mínimo conjunto de bits ordenados con los que el computador puede trabajar. Por lo general, **un byte equivale a 8 bits**.
- **Para almacenamiento en memoria:**
 - Utilizan **múltiplos del byte**, al ser el byte la unidad mínima almacenable en memoria.
 - Emplean **múltiplos de 2^{10}** ($= 1024$), pues se utilizan la base binaria siendo el número 2^{10} la potencia de 2 más próxima a 1000. *(Lo veremos en el próximo tema).*
- **En comunicaciones para referirse a la velocidad de transmisión:**
 - Utilizan **múltiplos del bit** siendo estos **múltiplos de 10^3** ($= 1000$) pues utilizan la base 10.

En Almacenamiento (bytes)	En comunicaciones (bits por segundo, bps)
Kibi byte (KiB) = 2^{10} bytes	Kilobits/seg (Kbps) = 10^3 bps
Mebibytes (MiB) = 2^{20} bytes = 1024 KB	Megabits/seg (Mbps) = 10^6 bits = 1024 Kbps
Gibibytes (GiB) = 2^{30} bytes = 1024 MB	Gigabits/seg (Gbps) = 10^9 bits = 1024 Mbps
Tebibytes (TiB) = 2^{40} bytes = 1024 GB	Terabits/seg (Tbps) = 10^{12} bits = 1024 Gbps
Pebibytes (PiB) = 2^{50} bytes = 1024 TB	Petabits/seg (Pbps) = 10^{15} bits = 1024 Tbps

1.2 TEXTO

■ **Carácter:**

- Unidad de información mínima e indivisible de la escritura como pueden ser letras, números, signos de puntuación, símbolos especiales, códigos de control...
- Ejemplos de caracteres: a A ? ñ : ú ë 9 ℒ = € ∞ گ [Ret]

■ **Codificación de caracteres:**

- Cada carácter se corresponde en un valor numérico (código) que lo identifica según el formato de codificación empleado.
- Los estándares de codificación de caracteres disponen de tablas de caracteres que relacionan cada carácter con su código.

■ Un texto es guardado en un computador almacenando la lista de valores numéricos de los caracteres que componen el texto.

- Ejemplo: la palabra “**Hola**” se representa con la lista de valores {72, 111, 108, 97} y se guardará como: 01001000 01101111 01101100 01100001

■ Los estándares de codificación de caracteres más conocidos son:

- **ASCII**
- **ASCII Extendido**
- **Unicode**

1.2 TEXTO: Codificación ASCII

- El código ASCII (*American Standard Code for Information Interchange*) surge en 1967 como un estándar a partir de los códigos utilizados en telegrafía.

- Los caracteres de este estándar se codifican con 7 bits por lo que está compuesto de sólo 128 caracteres.

- Dispone de 33 caracteres no imprimibles (caracteres de control), de letras mayúsculas y minúsculas, números y algunos símbolos.

- Las letras aparecen en orden alfabético. Ejemplo:

$$\text{ASCII}('a') + 1 = 97 + 1 = 98 = \text{ASCII}('b')$$

- Las letras mayúsculas aparecen antes que las letras minúsculas (sus códigos ASCII son menores)

- Los dígitos del 0 al 9 aparecen en orden creciente. Ejemplos:

- El código del carácter '0'+3 es el código del carácter '3':

- $\text{ASCII}('0') + 3 = 48 + 3 = 51 = \text{ASCII}('3')$

- El código de cualquier dígito menos el '0' da el valor al que representa dicho dígito:

- $\text{ASCII}('2') - \text{ASCII}('0') \rightarrow 50 - 48 \rightarrow 2$

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.asciitable.com

1.2 TEXTO: ASCII Extendido

- El ASCII estándar está orientado a representar los caracteres del inglés por lo que no incluye caracteres acentuados ni diacríticos.
- Surgieron otros códigos de 8 bits (256 caracteres) creados para representar otras lenguas además del inglés compatibles con ASCII → ASCII Extendido
- El ASCII extendido se refiere a cualquier sistema de codificación de 8 bits en el que los primeros 128 códigos coinciden con los del ASCII estándar y los 128 restantes pertenecían a la lengua en cuestión.
- Existen muchas codificaciones de ASCII extendido. La codificaciones más conocidas que incluyen el alfabeto español son:
 - **Página de códigos 850 (Europa Occidental):** Muy utilizaba en PC bajo MS-DOS.
 - **Norma ISO 8859-1:** contiene el alfabeto latino incluyendo los diacríticos y letras especiales.
 - **Norma ISO-8859-15:** una revisión de la 8859-1 que incluye, entre otros, el símbolo del euro

ASCII Estándar de la ISO-8859-12

Parte extendida de la ISO-8859-15

(en verde las modificaciones respecto la ISO 8859-1)

ISO-8859-15																
	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2x	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
8x	AD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SS2	SS3
9x	UCS	PU1	PU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	SI	OSC	PM	APC
Ax	NBSP	i	ç	£	€	¥	Š	š	š	©	ª	«	¬	SHY	®	—
Bx	°	±	²	³	Ž	µ	¶	·	ž	¹	º	»	CE	œ	Ÿ	¿
Cx	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Dx	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
Ex	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	ò	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

1.2 TEXTO: Unicode

- Surge con el propósito de reemplazar todos los estándares de codificación de caracteres por un sistema que contenga todos los símbolos de cualquier lengua.
- Es el sistema de codificación **más utilizado actualmente y completo** que abarca más de 100 mil caracteres procedentes de multitud de alfabetos actuales (griegos, latín, cirílico, árabe, chino...), de lenguas muertas (fenicio, rúnico...), sistemas de escritura como braille, multitud de símbolos (matemáticos, musicales, iconos...) y su número sigue creciendo en cada recisión del estándar.
- Cada código Unicode (conocido como punto de código) se representa con el prefijo 'U+' seguido un valor en hexadecimal de 4 o más dígitos.
- Posee un mecanismo para la **composición de caracteres**: Un carácter base se complementa con marcas: signos diacríticos, de puntuación o marcos.

$$\underset{\text{U+00F1}}{\tilde{n}} \equiv \underset{\text{U+006E}}{n} + \underset{\text{U+0303}}{\circ}$$

- Este estándar reconoce 3 posibles formatos según el número de bits que emplee el computador para representar los enteros: **UTF-8, UTF-16, UTF-32**
 - Los formatos UTF-8 y UTF-16 son más complejos de utilizar que los anteriores emplean códigos de longitud variable (de 1 a 4 bytes).
 - UTF-8 es compatible con ASCII estándar.

[2. ESTRUCTURA DE COMPUTADORES]

1.- INTRODUCCIÓN

2.- EL PROCESADOR (CPU)

3.- LA MEMORIA

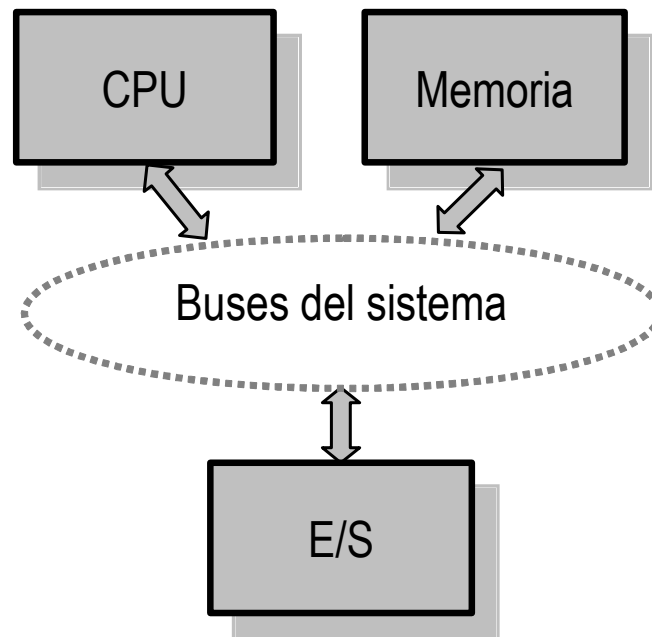
4.- EL SISTEMA DE E/S

5.- LOS BUSES

2.1. INTRODUCCIÓN

Organización básica de un computador

Modelo de *Von Neumann*



2.2 EL PROCESADOR

■ Conceptos generales:

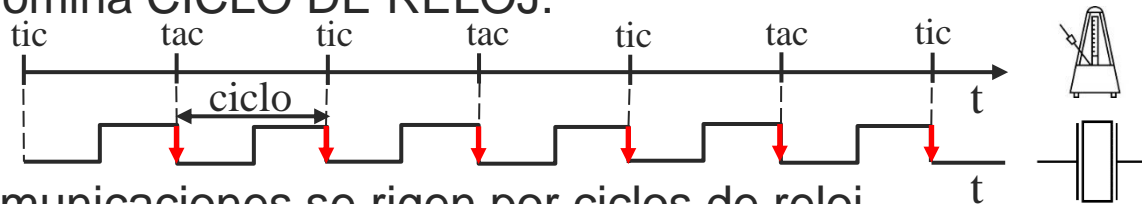
- Elemento principal y ACTIVO de un computador.
- Ejecuta las instrucciones de los programas una a una.
- **PALABRA:** tamaño de dato con el que opera o trabaja (en bits, y potencia de 2)
 - Aportado por el **BUS DE DATOS** (se verá luego)
- Formado por una serie de componentes internos que controlan la ejecución de las instrucciones paso a paso.
 - Uno de esos elementos internos es el **reloj**.

¿ES EL RELOJ LA VELOCIDAD DEL PROCESADOR?

Veamos qué es el reloj...

2.2 EL PROCESADOR

- ¿Qué es el reloj? → Símil del metrónomo
 - Los elementos internos deben comunicarse entre ellos.
 - ¿Cómo sabe cada uno cuándo mandar/recibir datos?
 - **Metrónomo**: utilizado para indicar el compás de una composición musical → reloj: metrónomo del procesador.
 - El reloj oscila (al igual que el metrónomo) de forma periódica e ininterrumpida, entre los valores 0 y 1.
 - Cada ciclo completo desde que comienza con el 0 hasta que vuelve a 0) se denomina **CICLO DE RELOJ**.



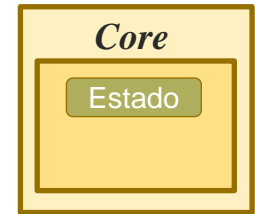
- Las comunicaciones se rigen por ciclos de reloj.
 - Por ejemplo: un componente A comienza a realizar una tarea con un dato y, posteriormente, se lo dará al componente B. Ambos saben que A tarda 3 ciclos de reloj en terminar su tarea, así que B se queda esperando “contemplando” el reloj hasta que pase dicho tiempo y, posteriormente, recibirá el dato.
- La velocidad o frecuencia del reloj se mide en Hercios (Hz) → $f = 1/T$, siendo T (periodo) el tiempo (en segundos) que dura un ciclo de reloj.

2.2 EL PROCESADOR: Tipos de procesadores

■ Mononúcleo (monocore)

- Compuesto por un sólo core o microprocesador dentro del chip procesador.
- Ejemplo: Pentiums

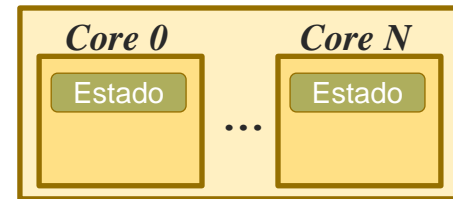
Proc. Monocore



■ Multinúcleo (multicore):

- Varios microprocesadores (núcleos) en el mismo chip.
- Cada núcleo ejecuta un programa diferente o, si el programa lo permite, divide la carga de trabajo del programa en subprocesos ejecutando cada uno en un núcleo distinto.
- Un multicore de N núcleos en condiciones ideales con un programa paralelo (perfectamente divisible entre sus cores) es N veces más rápido que el mononúcleo.
- Ejemplos: Core 2 dúo, quad-core, i5, i7...

Proc. Multicore



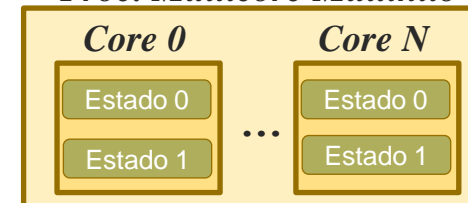
■ Multihilos (multithread):

- Core con algunos componentes internos duplicados permitiendo cambiar rápidamente de subproceso.
- Mejora leve respecto a monocore pero muy inferior a tener otro Core.

Proc. Multihilo



Proc. Multicore Multihilo



2.2 EL PROCESADOR: Rendimiento.

- La métrica más adecuada para cuantificar el rendimiento de un computador es el **tiempo de ejecución**, esto es, el tiempo necesario para ejecutar un programa concreto que actúa como patrón de medida.
- En el cálculo de este tiempo de ejecución se tiene en cuenta muchos aspectos de la CPU como la frecuencia de reloj, el repertorio de instrucciones (que afecta a la duración de las instrucciones, número de instrucciones...) o el paralelismo.

$$\text{TiempoEjecución}_{(\text{seg})} = N^{\circ}\text{Inst} * \text{CPI}_{\text{medio}(\text{Ciclos/Instr.})} * T_{(\text{seg})} / N^{\circ}\text{Cores}$$

NºInst = número de instrucciones ejecutadas (no es el que nº de instr. que compone el programa).

T = Periodo del reloj en segundos.

CPI_{medio} = duración media de las instrucciones del programa en ciclos.

NºCores = Núcleos del procesador (suponiendo que el programa puede paralelizarse completamente).

- Para comparar dos procesadores se emplea la aceleración:

$$\text{Aceleración} = \text{TiempoEjec}_{(\text{lento})} / \text{TiempoEjec}_{(\text{rápido})} \rightarrow \text{debe resultar} \geq 1$$

- **Ejemplo:** Sea el procesador A moncore a 2'5MHz con CPI = 2 y el Procesador B de 2 cores a 2MHz con CPI = 3. ¿Cuánto es más rápido uno respecto del otro?

$$\text{TiempoEjec}_A = N * 2 * (1/(2,5 * 10^6)) / 1 = 0,8 * 10^{-6} * N$$

$$\text{TiempoEjec}_B = N * 3 * (1/(2 * 10^6)) / 2 = 0,75 * 10^{-6} * N \quad \leftarrow \text{Más rápido B pues tarda menos}$$

$$\text{Aceleración} = \text{TiempoEjec}_A / \text{TiempoEjec}_B = 0,8 * 10^{-6} * N / 0,75 * 10^{-6} * N = 1,0667$$

El procesador B es 1,0667 veces más rápido que A.

2.3 MEMORIA

- Elemento de un Computador destinado a almacenar información de los programas que se están ejecutando.
- Elemento pasivo al que acceden los restantes elementos para leer y escribir información.
- ¿Cómo se comunican con él? → Recibe una posición (dirección) de memoria y devuelve lo que hay en ella (dato).
 - Similar a un casillero con múltiples posiciones (o a un vector)
- Conceptos generales:
 - UBICACIÓN: dónde se sitúa físicamente (CPU, Interna, Externa)
 - CAPACIDAD: Cantidad de información que se puede almacenar en ella. Expresada en Kbytes, Mbytes, Gbytes o Tbytes.
 - UNIDAD DIRECCIONABLE: Cantidad de información que se almacena en una celda de memoria. Cada celda posee una dirección que la identifica. Su tamaño suele ser de un byte o una palabra del procesador.
 - CAPACIDAD DE DIRECCIONAMIENTO: Número total de celdas de memoria que puede direccionar el procesador.
 - Viene dado por el Bus de Direcciones (se verá luego)
 - CAPACIDAD MÁXIMA: Capacidad de direccionamiento * Unidad direccionable
 - TECNOLOGÍA: HW específico empleado para la construcción de la memoria. Ej: semiconductora, magnética, óptica...

2.3 MEMORIA

- La memoria es siempre más lenta que la CPU.
- ¿Qué buscamos de la Memoria?
 - Alta Velocidad
 - Alta Capacidad
 - Bajo Coste
- Solución: organización de memorias de forma que el rendimiento global se acerque al ideal → Jerarquía de memoria.

¡Son excluyentes!
No hay tecnología que cumpla las tres

¡LA MEMORIA NO ES SOLO LA "RAM"! (Memoria principal)



2.3 MEMORIA

- Según su tecnología, hay varios TIPOS:
 - Semiconductoras: almacenan la info. en circuitos electrónicos.
 - Registros, M. Caché, M. Principal, Disco Duro de Estado Sólido (SSD)
 - También... Memorias flash, tarjetas de memoria...
 - Magnéticas: hacen uso de propiedades magnéticas para almacenar y leer info.
 - Discos duros, cintas magnéticas, discos flexibles...
 - Ópticas: hacen uso de las propiedades de la luz (reflexión) para la lectura de info.
 - CDs, DVDs, HD-DVD, Blu-Ray...
 - Hay algunos híbridos...
 - Minidisc...
- Nos centraremos en → Semiconductoras (implementadas en los principales niveles de la jerarquía de memoria)

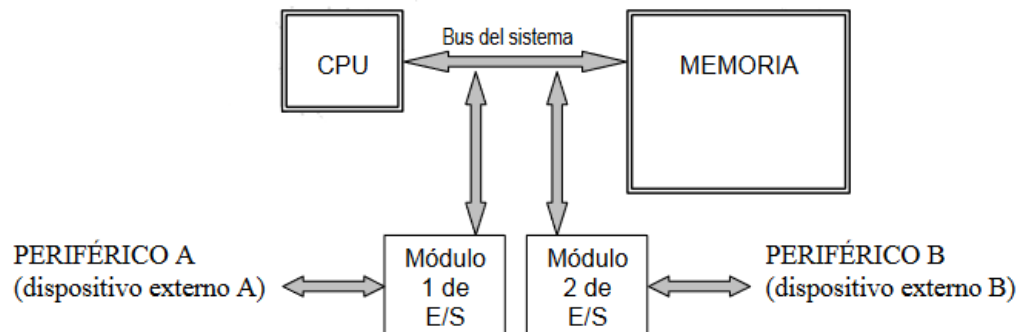
2.3 MEMORIA

■ Tipos de memorias semiconductoras:

- Volátiles: Contenido desaparece tras quitar alimentación (tipo RAM).
 - Estática (SRAM):
 - 1 bit: se almacena en un circuito con puertas lógicas que realimentan su contenido.
 - Utilizadas en memorias caché → rapidez y eficiencia.
 - Dinámica (DRAM):
 - Circuito basado en un condensador → NECESITAN QUE RECARGUEMOS LOS CONDENSADORES CADA CIERTO TIEMPO!!
 - Más lentas que SRAM por el refresco de los condensadores
 - Mayor capacidad/superficie porque cada bit necesita menos transistores.
 - Subtipos a lo largo de su evolución: EDO-DRAM, BEDO-DRAM, FPM-DRAM, RAMBUS (RDRAM), SDRAM (síncrona), DDR-SDRAM, [GDDR-SDRAM]
- No volátiles: Contenido permanece (tipo ROM).
 - ROM: programadas (escritas) en fábricas.
 - PROM: programables (escribibles) una sola vez.
 - EPROM: borrables mediante luz ultravioleta.
 - EEPROM: borrado electrónico (BIOS).
 - FLASH EPROM: iguales que EEPROM con más fácil borrado.

2.4 EL SISTEMA DE E/S

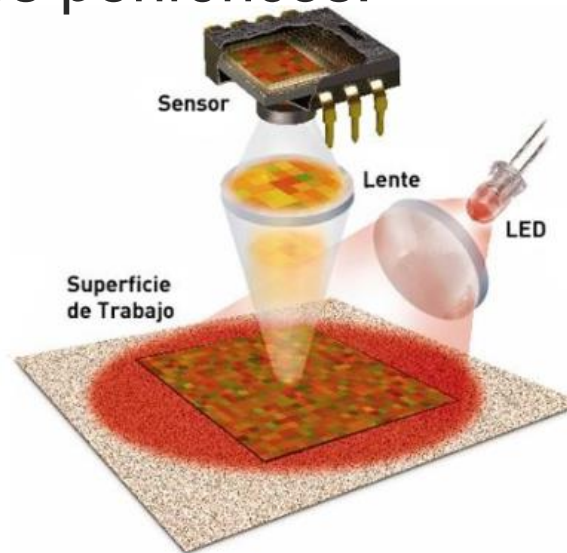
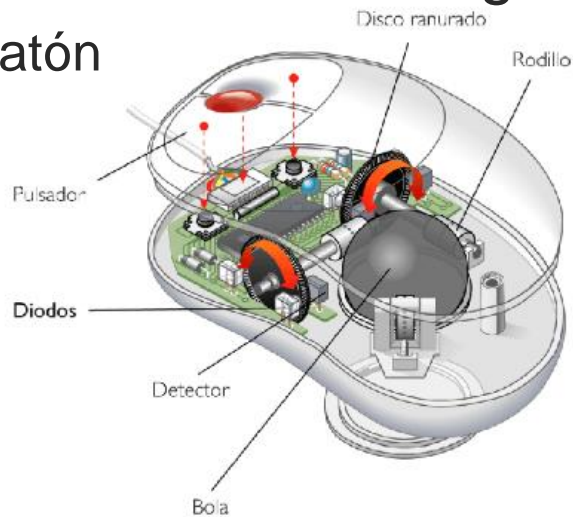
- El ordenador se comunica con el mundo exterior mediante periféricos.
- Estos se componen de:
 - Dispositivo externo o de E/S (ej. monitor, teclado, ratón, ...)
 - Módulo de E/S (ó controlador) a modo de adaptador/interfaz.
- ¿Por qué no conectar directamente?
¡Problema!: control y velocidades diversas, tamaño de datos diferente → Necesidad de hacer una adaptación.
- MÓDULO DE E/S ó Controlador (adaptador):
 - Permite comunicación CPU-Periféricos.
 - Genérico: puede gestionar dispositivos externos de varios tipos.
 - A veces pueden ser incluso auténticos procesadores.



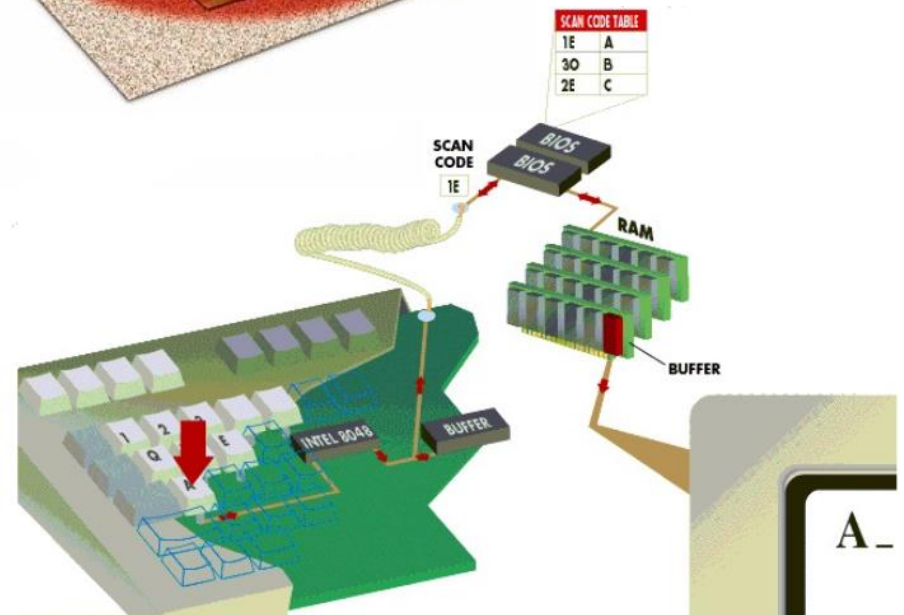
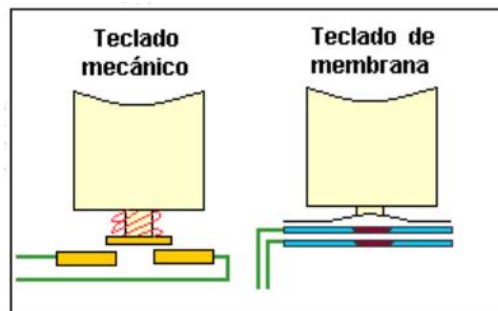
2.4 EL SISTEMA DE E/S

■ Funcionamiento de algunos periféricos:

○ Ratón



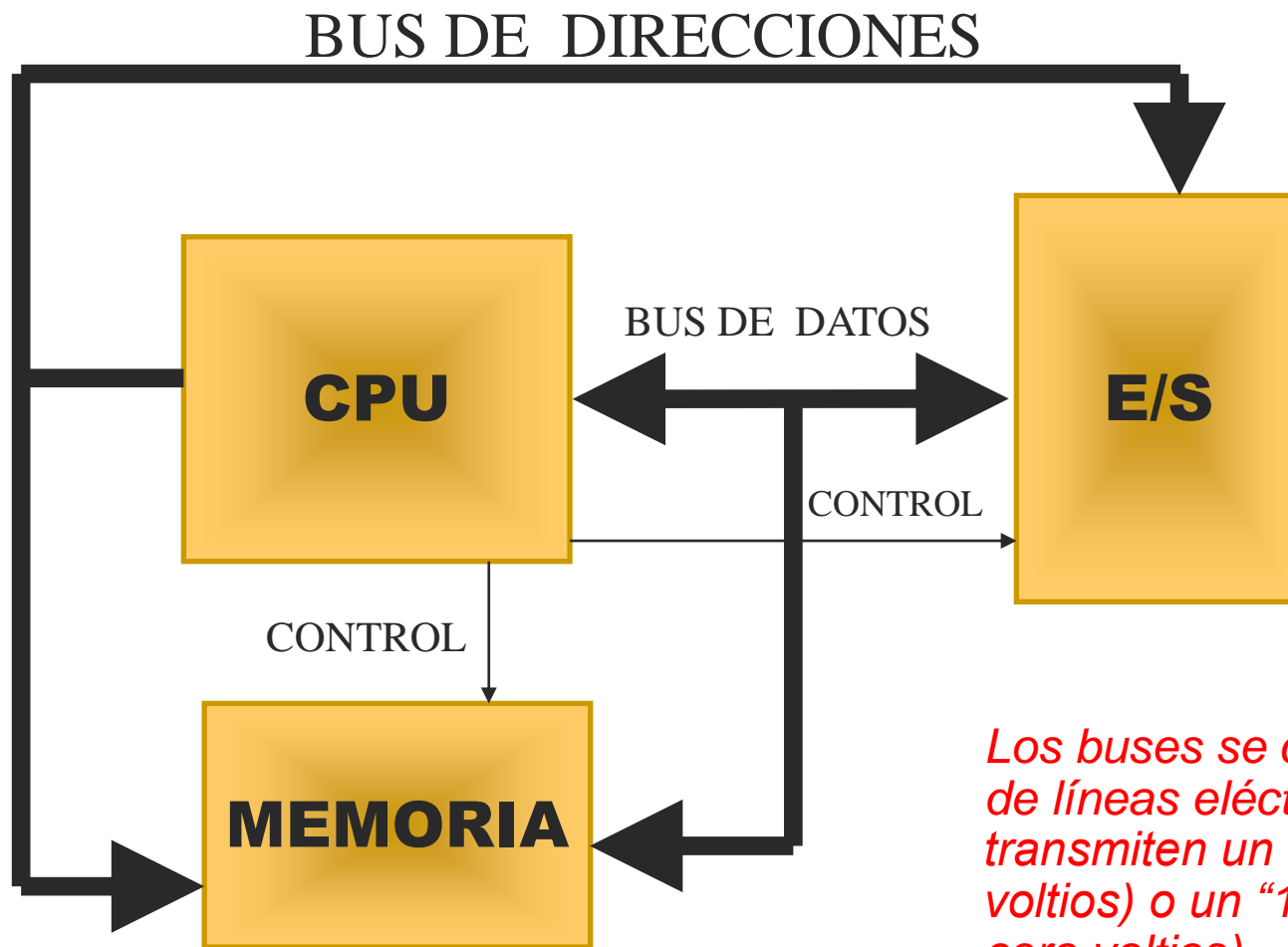
○ Teclado



2.5 LOS BUSES

- Los componentes de un computador (CPU, Memoria, E/S) se conectan entre sí mediante un conjunto de líneas que transmiten señales con funciones específicas.
- Hay 3 tipos de señales que constituyen un bus: direcciones (de memoria o E/S), datos y control.
 - **Bus de datos:** camino para transferir datos entre los elementos de un computador.
 - Su anchura (número de líneas eléctricas) suele ser una potencia de dos ($8=2^3$, $16=2^4$, $32=2^5$, $64=2^6$, ...).
 - El tamaño del bus de datos suele indicar el tamaño de palabra del procesador.
 - **Bus de direcciones:** identifican una **posición de memoria** o bien un **dispositivo externo** (direcciones de memoria o de E/S).
 - Son administradas por el procesador y su tamaño determina la capacidad de direccionamiento del procesador.
 - Con N líneas de dirección, la capacidad de direccionamiento del procesador es de 2^N celdas de memoria.
 - Ejemplo: Capacidad de máxima de direccionamiento del procesador con un bus de direcciones de 16 líneas y un bus de datos de 32 líneas.
$$\text{Capac. Máxima} = \text{N}^\circ \text{Celdas} * \text{Unidad Direc.} = 2^{16} \text{celdas} * 32 \text{ bits/celda} = 2^{16+2} \text{Bytes} = 256 \text{KBytes}$$
 - **Bus de control:** controlan el acceso y uso de los buses anteriores.
 - La mayoría de las líneas de control son gestionadas por la CPU.
 - Ejemplo: El procesador controla la línea de control RW de la memoria para ordenar una lectura o una escritura en memoria.

2.5 LOS BUSES



Los buses se componen de líneas eléctricas que transmiten un "0" (cero voltios) o un "1" (más de cero voltios).

3. REDES DE ORDENADORES

1. Introducción

2. Componentes

3. Tipos

- **Según la extensión que abarcan**

4. Modelos

1. OSI

2. TCP/IP

- **Capa de aplicación**
- **Capa de transporte**
- **Capa de internet (Direccionamiento IP)**
- **Interfaz de red (Direcciones MAC)**

BIBLIOGRAFIA

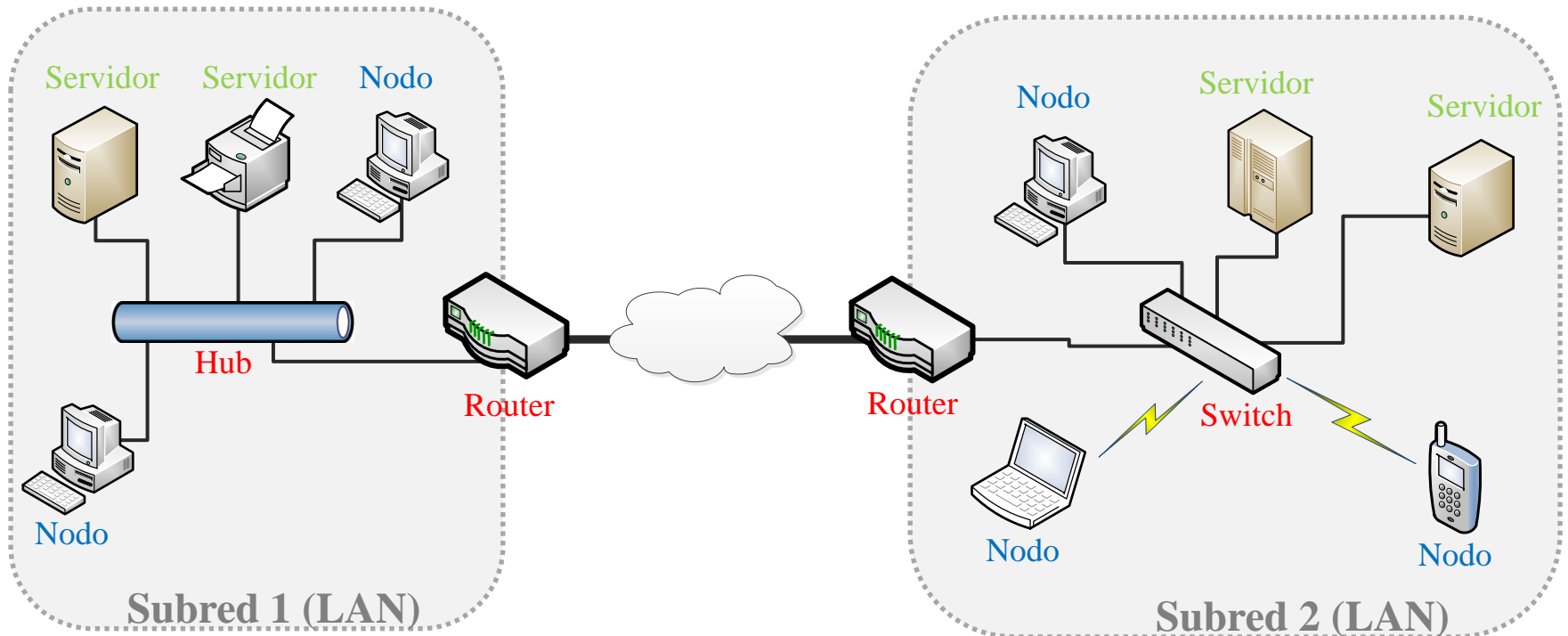
- **Bibliografía básica de referencia:**
 - **“Fundamentos de Informática para Ingeniería Industrial”, Fernando Díaz del Río et al.**
 - **“Computer Networks”, 4ª ed., A. S. Tanenbaum. Pearson/Prentice-Hall.**
 - **"Comunicaciones y Redes de Computadores", 7ª ed., W. Stallings, Pearson/Prentice-Hall.**

3.1 Introducción

- **Red:** conjunto de ordenadores conectados entre sí, que pueden comunicarse sin importar la localización física de los distintos dispositivos.
- **Objetivos de una red**
 - **Compartir recursos:** hacer que todos los datos y servicios de un equipo estén disponibles para cualquier elemento conectado a la red con autorización para ello.
 - **Fiabilidad:** Mantener datos importantes en varios computadores (actualizando las copias a través de la red) de forma que si una de ellas queda inoperativa pueda utilizarse otra copia.
 - **Supercomputación:** Grandes computadores formados por varios ordenadores pequeños conectados en red. Este diseño se debe a que los ordenadores pequeños tienen una mejor relación coste/rendimiento respecto a máquinas grandes (que son decena de veces más rápidas pero miles de veces mayor) y a que permiten aumentar el rendimiento gradualmente añadiendo más ordenadores.

3.2 Componentes de una red

- Se distinguen fundamentalmente 4 elementos:
 - Servidores:
 - Nodos:
 - Elementos de conmutación:
 - Líneas de comunicación:
- Ejemplo:



3.2 Componentes de una red

■ Servidor:

- Elementos de la red que gestionan:
 - La propia red, el acceso de los usuarios, la seguridad.
 - Los **recursos** de la red ofreciendo servicios y/o datos a los usuarios.
 - Ejemplo: Servidor web, de correos, de impresión, de base datos, de ficheros... muchos y cada día más. *¿Internet de las cosas?*
- Pueden ser desde un pequeño computador para uso doméstico o pequeña oficina hasta miles de computadores (Ej. servidores Google), pudiéndose emplear modelos de sobremesa, en rack dispuestos en armarios o 'miniPCs' muy compactos (Ej. un servidor NAS)
- Pueden clasificarse en:
 - **Servidores dedicados:** si no puede emplearse a la vez como estación de trabajo. Destinados exclusivamente a administrar los recursos de la red.
 - **Servidores no dedicados:** los que, además de ofrecer servicios, actúan como estaciones de trabajo.



■ Nodo o Estación de trabajo:

- Dispositivos como computadores, móviles, tablets, etc. conectados a la red que actúan de clientes haciendo uso de los recursos y servicios ofrecidos por los servidores.

3.2 Componentes de una red

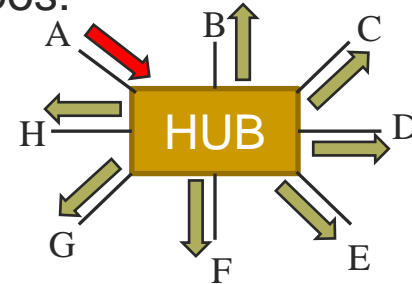
- **Línea de comunicación:** Medio de transmisión por el que se transportan los mensajes entre los elementos de una red. Diversos tipos: cables, wi-fi, satélites, etc.
 - Las líneas de comunicación pueden ser cableadas (medio de transmisión guiado o inalámbricas (medio de transmisión NO guiado)).
 - Medios de transmisión no guiados:
 - Conexión sin necesidad de conexión física pues la comunicación se realiza por ondas electromagnéticas (radiofrecuencia, microondas, infrarrojos...)
 - Ventajas: eliminación de cableado y facilidad de añadir nodos
 - Inconvenientes: seguridad y colisiones entre nodos.

3.2 Componentes de una red

■ Elementos de conmutación:

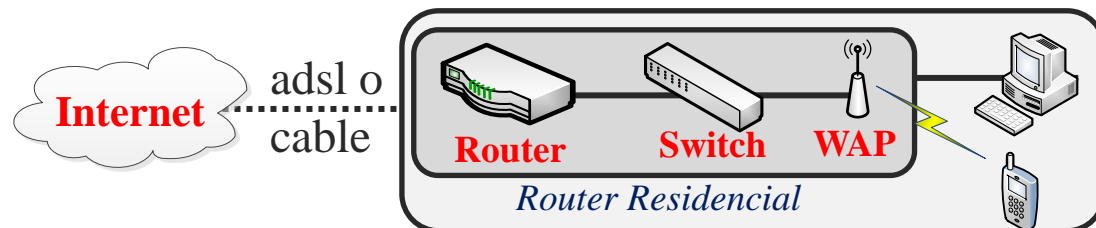
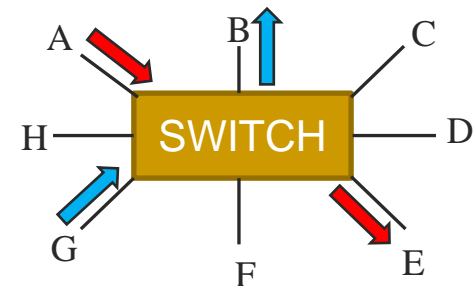
- Redirigen los mensajes que circulan en la red. Hay varios tipos:

- **Concentradores (hub):** reenvían por todos los puertos (dentro de una subred). Actualmente están en **desuso**.
- **Conmutadores (switch):** solo reenvía por el puerto correspondiente (dentro de una subred).
- **Enrutadores (routers):** interconectan subredes de **distintas tecnologías** reenviando los mensajes.



- ¿Qué instalan los proveedores de servicio de internet (ISP)?

- Un **Router residencial**, y se compone de:
 - Router de acceso
 - Switch
 - Punto de acceso inalámbrico.



3.3. Tipos: Según la extensión que abarcan

■ Según la extensión que abarcan

- **Redes de área personal** (PAN/WPAN, “Personal Area Network”):
 - *Abarca el entorno personal y cercano al usuario.*
 - *Emplean una comunicación rápida, eficaz y puntual, como el Bluetooth o infrarrojos*
- **Redes de área local** (LAN/WLAN, “Local Area Network”):
 - Su ámbito puede ser desde una casa o pequeña oficina con pocos equipos (varios metros) hasta un gran edificio con miles de computadores en la misma área de trabajo (unos pocos kilómetros).
 - Transmiten a alta velocidad con pocos errores pero limitadas en extensión y número de nodos.
- **Redes metropolitanas** (MAN/WMAN, “Metropolitan Area Network”):
 - Cubren desde un polígono industrial o campus hasta toda una ciudad.
 - Utilizado por entidades de servicios públicos como bancos.
 - Mayor velocidad. Más errores.
- **Redes de gran alcance** (WAN/WWAN, “Wide Area Network”):
 - Conectan desde ciudades, países, o continentes mediante satélites, cables interoceánicos, etc.. de infraestructuras telefónicas públicas o privadas.
 - Utilizadas por grandes compañías para uso propio o por proveedores de servicio de internet (ISP) para ofrecer el servicio de Internet a sus clientes.
- **A las que le precede la ‘W’ son versiones inalámbrica (Wireless)**

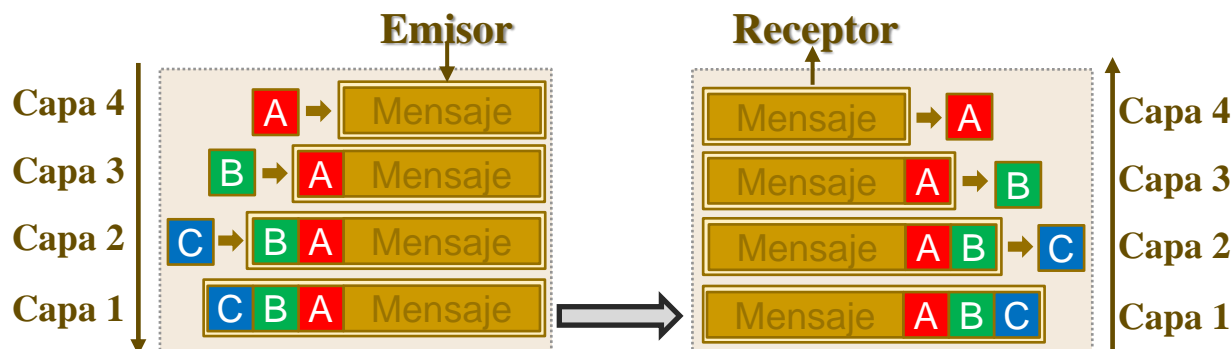
3.4 Modelos de Red

¿Qué es un protocolo?

- Conjunto de reglas usadas para la comunicación entre computadores a través de una red por medio de intercambio de mensajes.

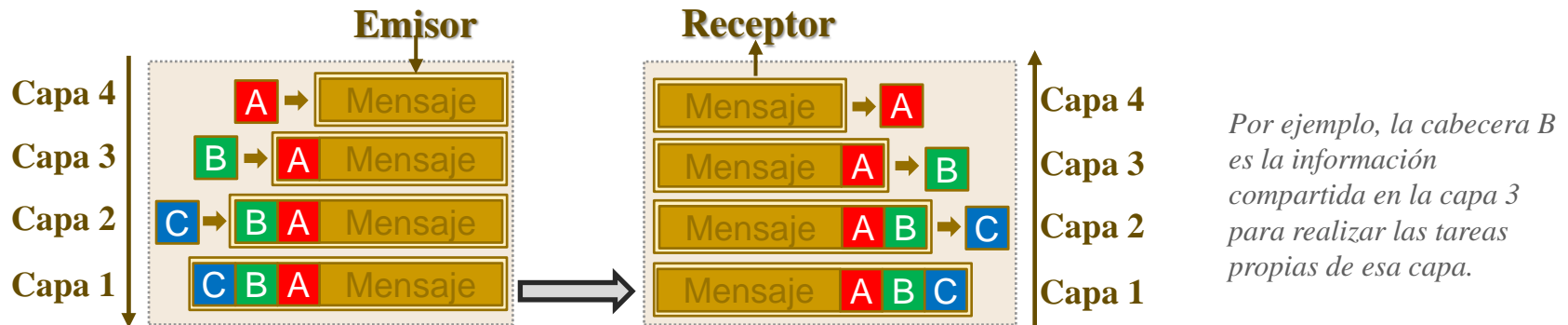
Modelo de red:

- Formado por una pila de protocolos (organizados en niveles/**capas**) → cada protocolo se encarga de una tarea en la comunicación.
- Las tareas a realizar por cada elemento de la pila se ejecutan en orden descendente en el emisor, y de forma ascendente en el receptor. Cada capa ofrece servicios a la capa inmediatamente superior y hace uso de los servicios de la inferior.
- Cada nivel de la pila de protocolos en el nodo emisor se comunica “virtualmente” con el mismo nivel del destinatario.



Por ejemplo, la cabecera B es la información compartida en la capa 3 para realizar las tareas propias de esa capa.

3.4 Modelos de Red



- Al enviar un mensaje
 - **En el emisor:** El paquete circula desde la capa superior a la inferior **añadiendo una cabecera**. La cabecera encapsulada por cada capa contiene la información que será interpretada por esa misma capa en el receptor para realizar las tareas asignadas.
 - El nivel inferior se encarga de transferir el paquete por el medio físico.
 - **En el receptor:** El paquete circula desde la capa inferior a la superior **desensamblando** la información que le corresponde en cada capa.
- *Por ejemplo: para mandar un correo electrónico a un destinatario, tiene que existir un protocolo que encapsule el contenido del correo, añadiéndole el destinatario, si es prioritario o no, el tipo de caracteres que utilizamos (ASCII, Unicode...)... un protocolo rige qué es lo que hay que añadir antes de mandarlo. Si el protocolo no añadiera, por ejemplo, el tipo de caracteres, el destinatario recibiría el correo pero solo podría ver un conjunto de información binaria sin poder ver caracteres*

3.4.1 Modelos de Red: Modelo OSI

- OSI (Open Systems Interconnection) es un modelo teórico utilizado como referencia para crear modelos reales divide la funcionalidad de una red en 7 capas

MODELO OSI



3.4.2 Modelos de Red: TCP/IP

- Modelo de red usado mundialmente para **internet** y **redes locales** y debe su nombre a los dos protocolos más importantes que implementa (TCP e IP).
- Nació en el Departamento de Defensa de EE.UU con el objetivo de crear una red que soportara cualquier circunstancia (como un conflicto bélico) utilizando varios tipos de medios y rutas alternativas para que lleguen siempre al destino.
- Tiene sólo 4 capas por lo que agrupa algunas capas del OSI.
 - Ejemplo: la capa de Aplicación del TCP/IP se encarga de las funciones de las capas Aplicación, Presentación y Sesión del modelo OSI.



3.4.2 Modelo TCP/IP: Capa de Aplicación

- Agrupa las 3 capas superiores del modelo OSI por lo que se encarga de funciones como dar soporte a las aplicaciones de usuario, representación de la información, su formato y codificación, o control de diálogo.
- Esta capa define las especificaciones para las aplicaciones más comunes a través de muy diversos protocolos. Alguno de ellos son:

OSI

Aplicación

Presentación

Sesión

TCP/IP

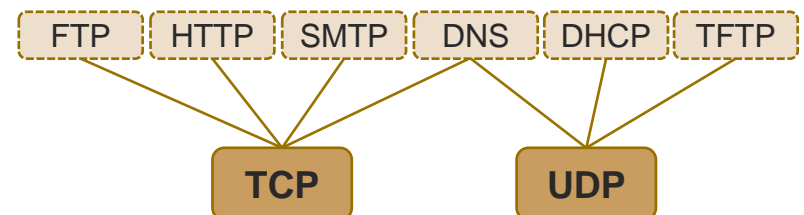
Aplicación

Protocolo		Puerto	Descripción
HTTP	<i>Hyper-Text Transfer Protocol</i>	80	Transmisión de páginas web
FTP	<i>File Transfer Protocol</i>	21	Transferencia de ficheros
SMTP	<i>Simple Mail Transfer Protocol</i>	25	Envío de correo a través del un servidor remoto
POP	<i>Post Office Protocol</i>	110	Descarga de correo de un servidor remoto
IMAP	<i>Internet Message Access prot.</i>	143	Acceso al servidor de correo sin necesidad de descarga
DNS	<i>Domain Name System</i>	53	Asigna nombres a cierta información (por ejemplo la IP)
DHCP*	<i>Dynamic Host Config. Prot.</i>	67/68	Asigna IP a los nodos clientes y registra quien la tiene.
TELNET	<i>Telecommunication Network</i>	23	Controla remotamente otra máquina

** Los proveedores de Internet envían por DHCP la dirección IP pública al router residencial y éste envía a su vez por DHCP direcciones IP privadas a todos los dispositivos de la red interna (intranet).*

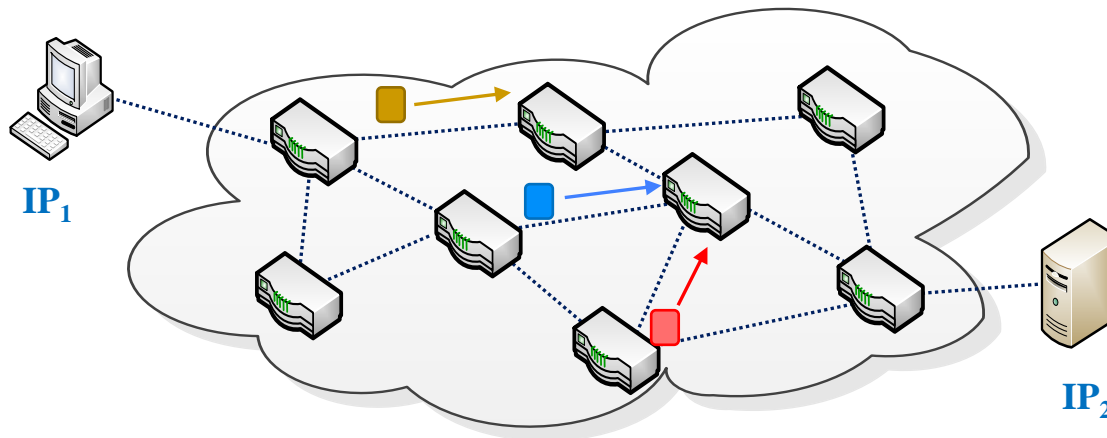
3.4.2 Modelo TCP/IP: Capa de Transporte

- Se encarga de aspectos relacionados con la calidad del servicio, la confiabilidad, el control de flujo y la corrección de errores.
- Protocolos empleados en la capa de transporte:
 - **TCP**: establece una conexión entre origen y destino (**protocolo orientado a la conexión**) permitiendo así crear un flujo de datos entre ambos.
 - Garantiza la entrega de datos sin errores y en el mismo orden en que se transmitieron (los datos se trocean en paquetes y se reconstruyen en el destino).
 - **UDP**: realiza envíos sin establecer una conexión previa (**protocolo NO orientado a la conexión**), por lo que no garantiza confiabilidad (no confirma la llegada, ni control de flujo, los paquetes pueden llegar desordenados...)
 - Utilizados en protocolos para la transmisión de video y audio en streaming, tareas de control, o cuando se envía poca información.
- Cada protocolo del nivel de aplicación empleará un protocolo de la capa de transporte según el servicio que precise.



3.4.2 Modelos TCP/IP: Capa de Internet

■ Se encarga de encaminar los paquetes hacia el destino a través de distintas redes físicas con independencia de la ruta elegida. Todos elementos de la red está identificado con una dirección IP.



■ Prácticamente toda la funcionalidad de esta capa la cubre el **protocolo IP**:

- A. **Direccionamiento IP**: método por el que se **identifica cada elemento de la red** y que consiste en un valor numérico distinto para cada uno.
- B. **Enrutamiento**: los paquetes deben ser dirigidos hacia el nodo destino identificado con una IP a través de los enrutadores (*routers*)

Veremos a continuación estas 2 funcionalidades

■ Esta capa puede contener otros protocolos como ICMP, RIP...

3.4.2 Capa de Internet: Direccionamiento IP

- **Dirección IP:** consiste en una etiqueta numérica asignada a cada nodo de la red que identifica, de manera **lógica** y **jerárquica**, a un dispositivo dentro de la red.
- Según la versión del protocolo IP pueden ser:
 - **Direcciones IPv4:** direcciones de 4 bytes.
 - Se representan expresando cada byte en notación decimal (por tanto cada byte está entre 0 y 255) y separadas por punto:
 - **Ejemplos:** 150.214.9.142 (de *www.us.es*) 216.58.214.163 (de *www.google.es*)
 - Posibilidad de **2^{32} direcciones = 4.294.967.296 direcciones**
 - Las más utilizadas actualmente pero ya no hay direcciones IPv4 libres.
 - Una dirección IPv4 se divide 2 partes

Identificador de red	Identificador de host
----------------------	-----------------------
 - Los bytes dedicados a identificar la red y host depende de la clase de red:

Clase	Tamaño	Direcciones de red	Ejemplo
A	Grande	0.x.x.x a 127.x.x.x	
B	Mediano	128.0.x.x a 191.255.x.x	150.214.x.x → Univ. Sevilla
C	Pequeño	192.0.0.x a 223.255.255.x	

Hay más clases, y además, cada clase tiene direcciones especiales.

- **Direcciones IPv6:** direcciones de 16 bytes
 - Se representan en 8 grupos de 2 bytes en hexadecimal y separadas por ‘:’
 - Ejemplo: 2001:0123:0004:00ab:0cde:3403:0001:0063
 - Posibilidad de **2^{128} direcciones > $3,4 * 10^{38}$ direcciones**
 - Se está implantando desde 2012.

3.4.2 Capa de Internet: Direccionamiento IP

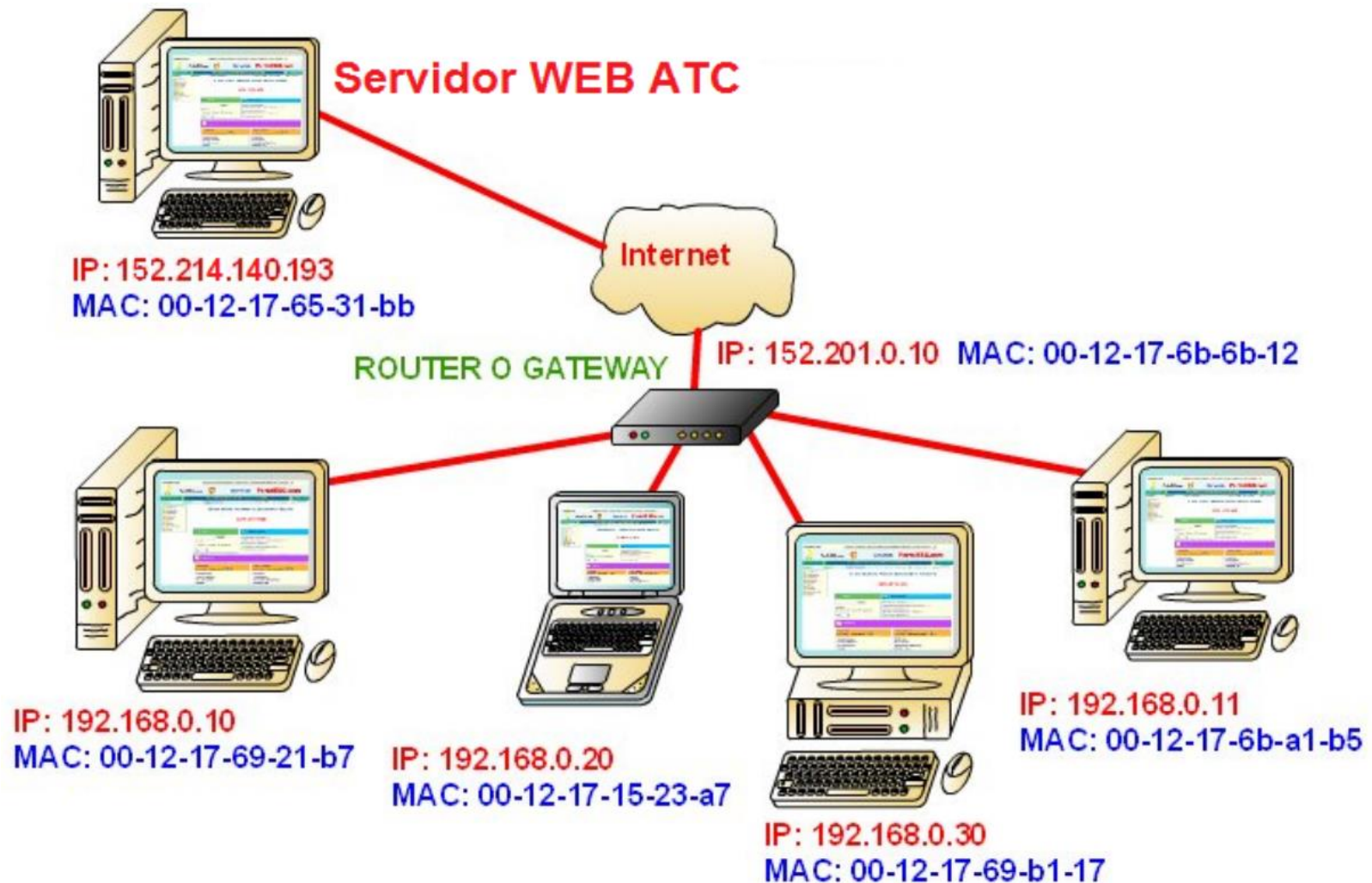
- Tipos de direcciones IP según su visibilidad(funcionalidad):
 - **Direcciones públicas:**
 - Conjunto de direcciones asignadas a los elementos **conectados directamente a internet** para comunicarse con el exterior.
 - Los proveedores de internet, empresas y organismos que precisan de estas direcciones deben solicitarlas a IANA, entidad que se encarga de asignarlas.
 - No puede haber dos elementos en internet con la misma IP.
 - Los routers residenciales tienen una IP pública a través de la cual son accesibles.
 - **Direcciones privadas:**
 - Conjunto de direcciones reservadas para ser asignadas a elementos de una red privada (una red cuyos elementos no se conectan directamente a internet).
 - En estos casos, los nodos de la red privada suelen acceder a internet a través de un router que hace de puente entre internet y la red privada.
 - No puede haber 2 elementos con la misma IP en una red privada pero sí entre redes privadas diferentes ya que la redes privadas no son visibles desde fuera.
 - Hay un rango de direcciones IP de cada clase que están reservadas como direcciones privadas.

Clase	Rango de Direc. privadas	Ejemplo
A	10.x.x.x a 10.x.x.x	
B	172.16.x.x a 172.31.x.x	
C	192.168.0.x a 192.168.0.x	192.168.0.1

Con **x** a 0 → Dirección de red

Con **x** a 255 → Dirección “broadcast”
(envía a todos los nodos de la red)

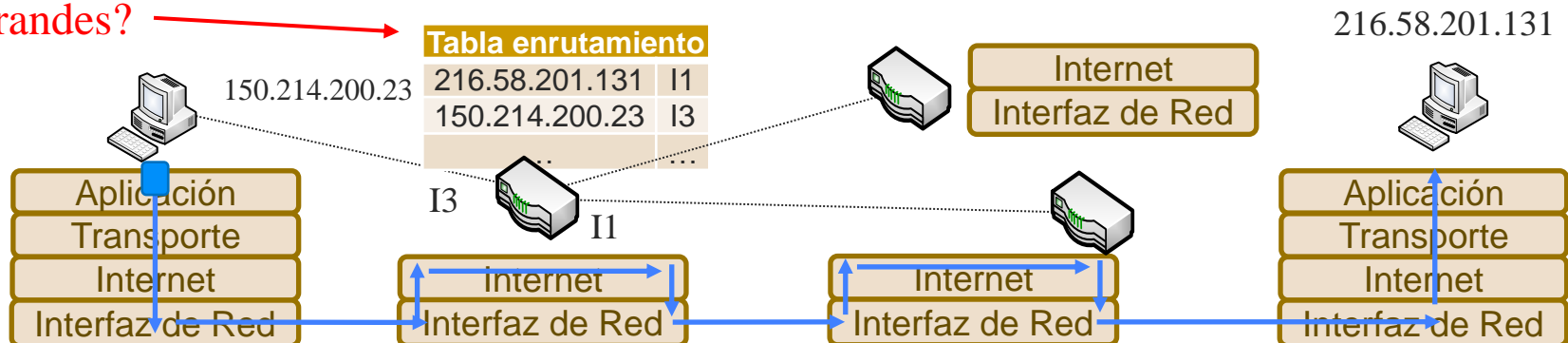
3.4.2 Capa de Internet: Direccionamiento IP



3.4.2 Capa de Internet: Enrutamiento

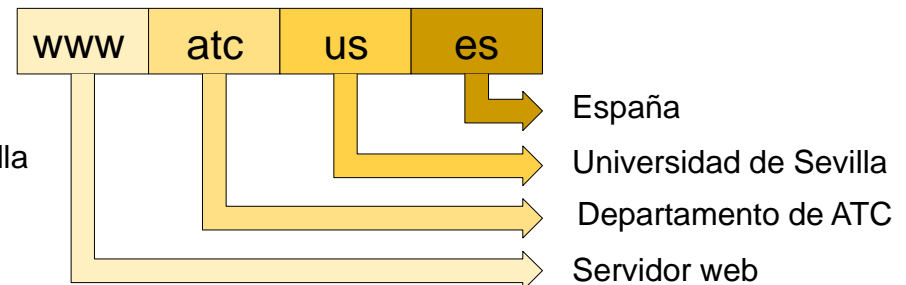
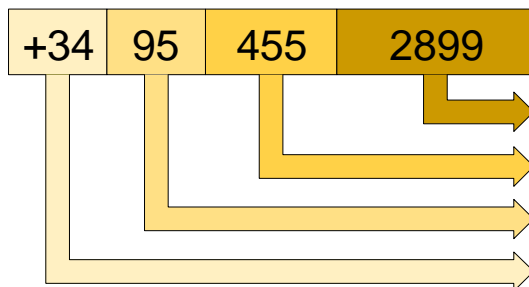
- El enrutamiento consiste en la búsqueda de un camino entre todos los posibles para que los paquetes viajen entre los enrutadores hacia el destino.
- No hay garantía de que llegue (protocolo no orientado a la conexión) ni si lo hará correctamente aunque lo hará lo mejor posible (mecanismo del mejor esfuerzo).
 - Símil del correo postal: al enviar una carta puede haber varias rutas posibles e intentan que llegue lo antes posible... pero lo más importante es que llegue.
- ¿Cómo decide la ruta? Los nodos envían cada paquete por una u otra interfaz empleando como criterio de mejor ruta alguna métrica: retraso entre nodos vecinos, distancia, ancho de banda,... (según el protocolo de enrutamiento).
 - Cada nodo posee una tabla de enrutamiento que relaciona las IPs con el interfaz de red que mejor dirige el paquete hacia el destino.

¿Muy grandes?



3.4.2 Capa de Internet: Direcciones IP

- Para conectarse a un servidor... ¿Necesitamos conocer la dirección IP? ¡¡NO!!
- Se utiliza el sistema de nombres de dominio **DNS** (Domain Name System)
 - Sistema de nomenclatura **jerárquica** para computadores, servicios o cualquier recurso conectado a Internet o a una red privada.
 - Asocia información variada con nombres de dominios asignado a cada uno de los participantes.
 - Su función más importante es **traducir** nombres inteligibles para las personas en direcciones IP, asociados con los equipos conectados a la red.
 - Su finalidad es localizar y direccionar todos los equipos mundialmente.
 - Actúa como un listín telefónico



3.4.2 Modelos TCP/IP: Capa Interfaz de red

- Abarca los niveles capa física y de enlace de datos del modelo OSI
- Define las especificaciones para transmitir a través de la red física (realiza un enlace físico real con los medios)
- Cada elemento de una red posee un **adaptador de red** por cada red a la que está conectada y que será específico según la tecnología de la red
- En muchas tecnologías, como la Ethernet, el adaptador de red posee una dirección denominada **dirección MAC** (Media Access Control) o dirección física.

- Es una dirección **no jerárquica** formada por 6 bytes:

PID (3 bytes)	UID (3 bytes)
Identifica al fabricante	Identifica el adaptador

- Se representan en hexadecimal separado por ':' (por ejemplo: 0A:18:FF:34:71:93)
- Es un identificador que se utiliza para comunicar los elementos que están conectados directamente.
- Los switches (conmutadores) utilizan esta dirección para enviar el paquete al destino.

