

# Llenguatge de marques i sistemes de gestió de la informació

R.A 1 : Reconeix les característiques de llenguatges de marques analitzant  
i interpretant fragments de codi

# Dades

- Les dades són representacions d'aspectes del món real i se solen recollir per fer càlculs, mostrar-les, organitzar-les, etc., amb l'objectiu que posteriorment algú en pugui fer alguna cosa: prendre decisions, generar noves dades...
- Entre les característiques interessants sobre les dades en destaquen sobretot tres aspectes:
  - A qui van dirigides:
    - a. **Dades destinades als humans:** generalment les dades destinades al humans requeriran que tinguin alguna estructura concreta, amb uns formats determinats, amb textos decorats d'alguna manera. Hi apareixeran títols, caràcters en negreta, etc. Generalment no cal conèixer quin significat tenen les dades, ja que la interpretació es deixa al lector.
    - b. **Dades destinades als programes:** els programes generalment no necessiten que les dades tinguin informació sobre com s'han de representar, sinó que n'hi ha prou que siguin fàcilment identificables, que quedi clar de quin tipus són i que hi hagi alguna manera de determinar què signifiquen per poder-les tractar automàticament.
  - La possibilitat de reutilitzar-les:

Molt sovint les dades es voldran reutilitzar per poder fer tasques diferents. Un error corrent sol ser emmagatzemar-les específicament per fer una tasca concreta, ja que això pot provocar que posteriorment sigui molt més complicat fer-les servir per fer altres tasques. Per tant, és bàsic disposar d'un sistema d'emmagatzematge que permeti aconseguir que les dades puguin ser reutilitzades fàcilment i si pot ser que puguin ser reutilitzades tant per les persones com pels programes.
  - Que es puguin compartir:

En un sistema informàtic modern s'ha de tenir en compte la possibilitat, a l'hora d'emmagatzemar dades, el fet que aquestes dades siguin compartides i, per tant, cal emmagatzemar-les d'alguna manera que no tingui problemes per usar-les en sistemes diferents.

# Emmagatzematge de dades en ordinadors

Atesa la seva arquitectura, els ordinadors emmagatzemen la informació en binari i, per tant, tota la informació que s'hi pot emmagatzemar sempre es representarà en uns i zeros (1, 0). Això fa que per representar qualsevol tipus de dades (imatges, vídeos, text...) calgui fer algun tipus de procés que converteixi les dades a una representació en format binari.

Tradicionalment en els ordinadors les dades s'organitzen de dues maneres:

- Dades de text
- Dades binàries

# Dades binàries (I)

Emmagatzemar les dades de manera binària és la manera natural d'emmagatzemar dades en ordinadors. Estrictament parlant, les dades binàries estan en el format que fa servir l'ordinador, ja que només són una tira de bits un rere l'altre. Per tant, normalment, un ordinador no haurà fer cap procés especial per emmagatzemar i llegir dades binàries.

Les dades en format binari tenen una sèrie de característiques que les fan ideals per als ordinadors:

- Generalment estan optimitzades per ocupar l'espai necessari.
- Els ordinadors les llegeixen fàcilment.
- Poden tenir estructura.
- És relativament fàcil afegir-hi metadades.

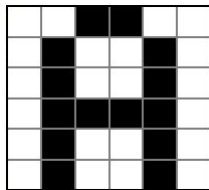
Per emmagatzemar el nombre 150 només cal convertir aquest valor decimal a la seva representació en binari i emmagatzemar-lo. És trivial comprovar que pot ser emmagatzemat en un sol byte (8 bits) com es pot veure

valor decimal	valor binari
150	10010110

# Dades binàries (II)

- Metadades:

- Molt sovint no s'emmagatzemen directament les dades tal com estan sinó que es processen per optimitzar-les, com ara emmagatzemant informació sobre el seu contingut o aplicant-hi procediments d'optimització. Aquestes optimitzacions són transparents per l'usuari final, que visualitzarà les dades normalment.
- Una de les maneres més senzilles de representar una imatge en un ordinador consisteix a representar cada un dels punts de color que la formen. O sigui, que només cal dir de quin color serà cada un dels punts per poder emmagatzemar la imatge en un fitxer.



```
001100
010010
010010
011110
010010
010010
```

Memòria interna:

```
001100010010010010011110010010010
```

Es pot representar de diferents formes, també podem guardar-ho en bits dient la quantitat de 0's i 1's volem emmagatzemar. Suposant que es repeteixi molt obtindrem una cadena de bits com la següent:

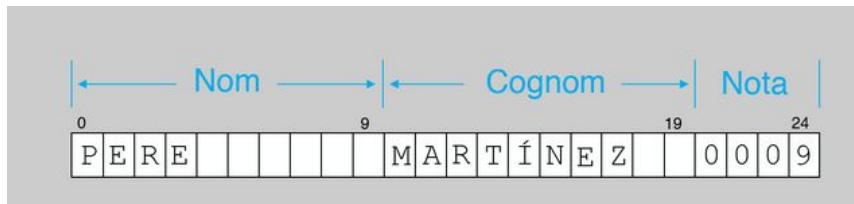
```
202120102011202011202041102011202011202011
```

# Dades binàries (III)

- Dades estructurades

- Les dades en la manera com les generem els humans no estan en un format que en faciliti el tractament automàtic per part d'un ordinador. Per aquest motiu sovint les dades que han de ser processades pels ordinadors es converteixen a algun format que sigui més idoni per al tractament. El més corrent és tractar les dades per tal que tinguin algun tipus d'estructura.
- La manera més corrent d'estructurar dades binàries sol ser tenir-les agrupades en registres que contenen la informació repetitiva d'una dada en concret. És habitual que els llenguatges de programació tinguin alguna manera de definir dades estructurades. Per exemple, per a aquestes tasques el llenguatge C fa servir els **struct**.

```
struct alumne {  
    char nom[10];  
    char cognom[10];  
    int nota;  
}
```



Generalment aquestes dades estructurades s'emmagatzemen en forma de llistes o conjunts de registres, de manera que el desenvolupador del programa podrà accedir a les dades de tots els **alumnes** simplement recorrent els diferents registres un per un.

Les dades estructurades facilitaran que les aplicacions les puguin tractar de manera automàtica.

# Dades binàries (IV)

- Lectura de dades

- Forma de lectura del processador: Tots els processadors no emmagatzemen la informació de la mateixa manera (tècnicament es fa referència a l'ordre de lectura en les adreces de memòria). Hi ha dos grans sistemes per emmagatzemar la informació en ordinadors:

- **Big endian:** les dades s'escriuen en l'ordre en què es creen. Així, per escriure **hola** en l'ordinador s'emmagatzemaria **h, o, l, a**. Aquest sistema és el que fan servir els processadors de Motorola.
- **Little endian:** les dades es desen de menys rellevant a més rellevant: **a, l, o, h**. Aquest sistema és el que fan servir els processadors d'Intel.

- Forma de lectura dels humans: Un problema diferent és que les dades en format binari estan pensades per ser llegides per màquines, però no per humans, de manera que són ideals per ser emmagatzemades en màquines, van bé per a la comunicació d'informació entre màquines, però en canvi perquè un humà les pugui fer servir **caldrà tenir un programa específic** per llegir-les. Això és el que passa per exemple, amb els fitxers d'imatges JPG, PNG, GIF..., que poden ser llegits per diferents programes perquè la seva especificació és pública.

- JPG - Estàndard ISO/IEC 10918
- GIF - Especificació del W3C gif89a
- PNG - Estàndard ISO/IEC 15948



# Dades de text (I)

Per a un ordinador no hi ha gaire diferència a l'hora d'emmagatzemar els fitxers de text o els fitxers binaris, ja que els fitxers de text també són tires de bits. La diferència és que aquest cop els bits estan agrupats d'una manera estàndard i coneguda: un **codi de caràcters**. Aquesta codificació sol consistir a determinar una quantitat de bits predefinida per marcar un caràcter i posteriorment s'associa un valor numèric a cada un dels caràcters.

L'equivalència entre els caràcters i els seus valors numèrics no es pot fer de manera aleatòria, ja que s'estaria creant el mateix problema que hi ha amb les dades binàries. Si es vol aconseguir que les dades es puguin llegir en diferents sistemes cal seguir algun tipus de norma coneguda per tothom. Per aquest motiu van aparèixer els **estàndards de codificació de caràcters**.

Caràcter	Valor decimal	Valor binari
A	0	000
E	1	001
I	2	010
O	3	011
U	4	100
Espai	5	101

Caràcter	Valor decimal	Valor binari
Espai	0	000
A	1	001
E	2	010
I	3	011
O	4	100
U	5	101



# Dades de text (II)

## ASCII

Un dels primers estàndards que va ser adoptat majoritàriament va ser ASCII (*American standard code for information interchange*), codifica cada un dels caràcters amb set bits i defineix a quin valor numèric es correspon cada un dels caràcters de la llengua anglesa.

Caràcter	Valor decimal	Caràcter	Valor decimal	Caràcter	Valor decimal	Caràcter	Valor decimal	Caràcter	Valor decimal
	32	3	51	F	70	Y	89	l	108
!	33	4	52	G	71	Z	90	m	109
"	34	5	53	H	72	[	91	n	110
#	35	6	54	I	73	\	92	o	111
\$	36	7	55	J	74	]	93	p	112
%	37	8	56	K	75	^	94	q	113
&	38	9	57	L	76	_	95	r	114
'	39	:	58	M	77	`	96	s	115
(	40	;	59	N	78	a	97	t	116
)	41	<	60	O	79	b	98	u	117
*	42	=	61	P	80	c	99	v	118
+	43	>	62	Q	81	d	100	w	119
,	44	?	63	R	82	e	101	x	120
-	45	@	64	S	83	f	102	y	121
.	46	A	65	T	84	g	103	z	122
/	47	B	66	U	85	h	104	{	123
0	48	C	67	V	86	i	105		124
1	49	D	68	W	87	j	106	}	125
2	50	E	69	X	88	k	107	~	126

# Dades de text

Lectura de dades automatitzada:

Els programes d'ordinador encara no són gaire bons interpretant les dades si són en text narratiu i, per tant, generalment convé que les dades que hauran de ser tractades per programes d'ordinador estiguin definides amb algun tipus d'**estructura** per tal que els siguin més fàcils de tractar.

S'han inventat sistemes per fer que les dades dels fitxers de text puguin ser estructurades. Un dels formats que s'ha fet servir durant molt de temps per exportar dades estructurades contingudes en bases de dades o fulls de càlcul a text ha estat el CSV (*comma separated values*).

```
"Manel", "Puig", "Garcia", 8  
"Pere", "González", "Puigdevall", 5  
"Maria", "Pous", "Canadell", 7
```

Dada	Resultat
Manel	Dada de text perquè està entre cometes
Puig	Dada de text perquè està entre cometes
Garcia	Dada de text perquè està entre cometes
8	Dada numèrica

CSV: és una manera senzilla de desar dades estructurades en format de text que permet a un programa identificar les diferents dades que conté cada registre i a més interpretar de quin tipus són. Simplement es limita a separar cada un dels registres de l'estructura en línies i els camps se separen amb comes. A més, per poder definir els tipus de dades, envolta de cometes les dades de text, mentre que no es posen cometes en les numèriques.

# Fitxers de marques (I)

Els fitxers de marques són una manera diferent d'emmagatzemar informació en ordinadors que s'afegeix a les maneres d'emmagatzemar la informació per mitjà de fitxers binaris o fitxers de text. L'objectiu principal dels fitxers de marques és intentar **recollir les millors característiques dels fitxers de text i binaris**. Els fitxers de marques prenen com a base els fitxers de text per aprofitar-se de les característiques més interessants d'aquest tipus de fitxers:

- La facilitat de creació i lectura.
- El compliment d'estàndards d'emmagatzematge definits i públics.

Però els fitxers de marques no solament s'intenten aprofitar de les característiques dels fitxers de text sinó que també intenten **aconseguir les característiques més interessants dels fitxers binaris**, com:

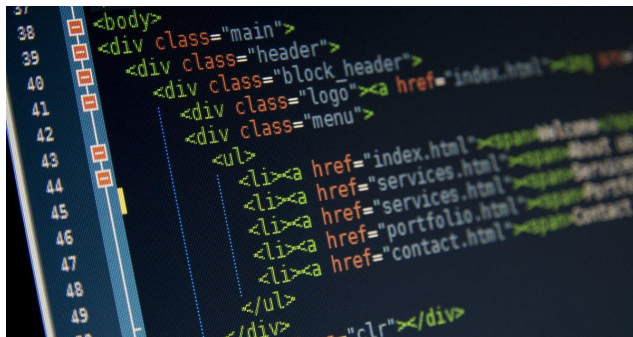
- La incorporació de metadades.
- La definició de l'estructura de les dades.

Això fa que els llenguatges de marques adquireixin una de les característiques més interessants dels fitxers binaris, que és la possibilitat d'incorporar informació sobre les dades –metadades– però intentant que afecti el mínim possible la llegibilitat del document. També permeten definir les dades i la seva estructura de manera que sigui senzill per un programa poder-les interpretar.

# Fitxers de marques (II)

Gràcies als avantatges que ofereixen els llenguatges de marques, aquests s'han convertit ràpidament en una de les maneres habituals de representar dades i es poden trobar contínuament en les tasques habituals amb ordinadors:

- L'exponent més popular és Internet –el Web–, que està basat totalment en els llenguatges de marques.
- Molts dels programes d'ordinador que feu servir habitualment fan servir en algun moment alguna o altra forma d'algun llenguatge de marques per a emmagatzemar les seves dades de configuració o de resultats:
  - Internament els formats de documents de Microsoft Office o d'OpenOffice.org o LibreOffice estan basats en llenguatges de marques.
  - Microsoft Visual Studio desa la seva configuració fent servir llenguatges de marques.
  - etc.



```
37 <body>
38 <div class="main">
39 <div class="header">
40 <div class="block_header">
41 <div class="logo"><a href="index.html">
42 <div class="menu">
43 <ul>
44 <li><a href="index.html">
45 <li><a href="services.html">
46 <li><a href="services.html">
47 <li><a href="portfolio.html">
48 <li><a href="contact.html">
49 </ul>
</div>
</div>
```

# Les marques

Les marques són una sèrie de codis que s'incorporen als documents electrònics per determinar-ne el format, la manera com s'han d'imprimir, l'estructura de les dades, etc. Per tant, són **anotacions que s'incorporen a les dades però que no en formen part**.

Les marques, per tant, han de ser fàcilment distingibles del text normal (per la seva posició, perquè segueixen algun tipus de sintaxi, etc.). Les marques més usades són les que estan formades per textos descriptius i estan envoltades dels símbols de “més petit” (<) i “més gran” (>) i normalment n'hi sol haver una al principi i una al final:

```
<nom>Manel Puig Garcia</nom>
```

Aquestes marques poden ser imbricades per **indicar estructures de dades**:

```
<persona>  
  <nom>Manel Puig Garcia</nom>  
  <nom>Pere González Puigdevall</nom>  
  <nom>Maria Pous Canadell</nom>  
</persona>
```

# Altres formes de marques

Hi han altres formes de marques:

- Una altra idea consisteix a trobar alguna combinació de caràcters que surti rarament en el llenguatge habitual. El TeX fa servir les barres invertides per a indicar l'inici de les marques.

```
\section{Persones}  
\begin{itemize}  
\item Manel Puig Garcia  
\item Pere González Puigdevall  
\item Maria Pous Canadell  
\end{itemize}
```

- Altres llenguatges de marques fan servir caràcters no habituals en determinades posicions per indicar que són marques. Per exemple amb Wiki Markup els caràcters “=” a la primera posició d'una línia es fan servir per indicar que el text és un títol d'apartat i el \* per les llistes de punts:

```
= Persones =  
* Manel Puig Garcia  
* Pere González Puigdevall  
* Maria Pous Canadell
```

# Característiques dels llenguatges de marques

Els llenguatges de marques són una manera de codificar un document de text de manera que per mitjà de les marques (l'equivalent de les metadades dels fitxers binaris) s'hi incorpora informació relativa a com s'ha de representar el text, sobre quina estructura tenen les dades que conté, etc...

Principals característiques:

- Es basen en text pla: els llenguatges de marques es poden interpretar directament i això només és possible si fem servir el format de text, ja que els binaris requereixen un programa per interpretar-los. Però a més tenen l'avantatge que són independents de la plataforma, del sistema operatiu o del programa. **Només requereixen un simple editor de textos.**
- Ús de metadades: Les marques són la manera com s'afegeixen les metadades als documents de text i com s'aconsegueixen superar les limitacions del format de text i aconseguir alguns dels avantatges dels fitxers binaris.
- Facilitat de procés: El fet d'incloure l'estructura permetrà que un programa pugui interpretar cada una de les dades d'un fitxer de marques per representar-lo o tractar-lo convenientment, ja que mostren l'estructura de les dades que contenen.
- Facilitat de creació i representació de dades diverses: Actualment s'estan fent servir fitxers de marques per representar imatges vectorials, fórmules matemàtiques, crear pàgines web, executar funcions remotes per mitjà de serveis web, representar música o sons, etc. I sense importar quin tipus de dades s'hi representin sempre hi haurà la possibilitat de crear aquests fitxers des d'un editor de text bàsic.

# Classificació dels llenguatges de marques

Hi han diferents tipus de classificacions dels llenguatges de marques, però la més habitual es basa en 2:

- **Llenguatges procedimentals i de presentació**, orientats a especificar com s'ha de representar la informació. En aquests llenguatges el que es fa és indicar de quina manera s'ha de fer la presentació de les dades. Ja sigui per mitjà d'informació per al disseny (marcar negretes, títols, etc.) o de procediments que ha de fer el programari de representació. L'exemple més popular d'aquests llenguatges és l'HTML però n'hi ha molts més: TeX, Wikitext...

## Wiki markup (Wikipedia)

```
= Classe =  
  
== Assignatura: XML ==  
[[Fitxer:xml.png]]  
  
'''Professor'''  
:* 'Manel Puig'  
  
'''Alumnes'''  
  
:* Frederic Puig  
:* Filomeno Garcia  
:* Manel Puigdevall
```

→ Resultat →

## Classe

Assignatura: XML



**Professor:**

- Manel Puig

**Alumnes**

- Frederic Puig
- Filomeno Garcia
- Manel Puigdevall



# Classificació dels llenguatges de marques

- **Llenguatges descriptius o semàntics:** orientats a descriure l'estructura de les dades que conté. En aquests llenguatges es descriu quina estructura lògica té el document ignorant de quina manera serà representada en els programes. Només es posen les marques amb l'objectiu de definir les parts que donen estructura al document.

```
<alumnes>
  <persona>
    <nom>Pere</nom>
    <cognom>Puig</cognom>
  </persona>
  <persona>
    <nom>Manel</nom>
    <cognom>Garcia</cognom>
  </persona>
</alumnes>
```

Aquest document mostra **quina és l'estructura de les dades** que conté i a més aquesta també es pot descobrir tot interpretant les etiquetes el seu contingut semàntic. A partir dels coneixements que es tinguin es dedueix que **Pere** és el nom d'una persona que és un alumne.