

XML

M04 UF1 A1.1

XML eXtensible Markup Language

Format universal per a documents i dades estructurades a Internet

És a dir:

Un format DE TEXT que permet crear arxius, on les dades estan estructurades i són fàcils de compartir i transformar a altres formats

XML aplicacions habituals

Arxius d'ofimàtica (docx, xlsx ...)

Paràmetres de configuració (.plist)

Dibuixos tècnics (.svg)

Pàgines web (.html)

...

XML vs HTML

XML és un format més estricte que HTML:

- Si l'arxiu XML no està ben format, no funciona
- XML no pre-defineix elements/tags ni els seus atributs, aquests són a lliure convenció de cada arxiu

XML, avantatges

- És portable (és a dir fàcil d'interpretar per diferents aplicacions)
- Depuració (és fàcil veure si el format és correcte o incorrecte)
- Independència de la plataforma (no depèn ni del processador ni del SO)
- Editable (arxius de text plà)

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
<!DOCTYPE note SYSTEM "Note.dtd">
<comanda>
  <producte codi="a0" seccio="b">
    <nom>Xinxeta</nom>
    <quantitat>20</quantitat>
  </producte>
  <producte codi="a1" seccio="b">
    <nom>Boli</nom>
    <quantitat>10</quantitat>
  </producte>
  <enviament origen="Sevilla" desti="Barcelona" />
</comanda>
```

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
```

Defineix la versió de XML i format de l'arxiu

```
<!DOCTYPE note SYSTEM "Note.dtd">
```

```
<comanda>
```

```
  <producte codi="a0" seccio="b">
```

```
    <nom>Xinxeta</nom>
```

```
    <quantitat>20</quantitat>
```

```
  </producte>
```

```
  <producte codi="a1" seccio="b">
```

```
    <nom>Boli</nom>
```

```
    <quantitat>10</quantitat>
```

```
  </producte>
```

```
  <enviament origen="Sevilla" desti="Barcelona" />
```

```
</comanda>
```

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
<!DOCTYPE note SYSTEM "Note.dtd">
<comanda>
  <producte codi="a0" seccio="b">
    <nom>Xinxeta</nom>
    <quantitat>20</quantitat>
  </producte>
  <producte codi="a1" seccio="b">
    <nom>Boli</nom>
    <quantitat>10</quantitat>
  </producte>
  <enviament origen="Sevilla" desti="Barcelona" />
</comanda>
```

← És opcional, és per validar que els elements i els atributs segueixen una definició específica

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
```

```
<!DOCTYPE note SYSTEM "Note.dtd">
```

<comanda>

```
<producte codi="a0" seccio="b">
```

```
<nom>Xinxeta</nom>
```

```
<quantitat>20</quantitat>
```

```
</producte>
```

```
<producte codi="a1" seccio="b">
```

```
<nom>Boli</nom>
```

```
<quantitat>10</quantitat>
```

```
</producte>
```

```
<enviament origen="Sevilla" desti="Barcelona" />
```

```
</comanda>
```

Elements o etiquetes

- Es defineixen entre '<' i '>'
- Diferència entre majúscules i minúscules
- Poden tenir altres continguts a dins

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
```

```
<!DOCTYPE note SYSTEM "Note.dtd">
```

```
<comanda>
```

```
  <producte codi="a0" seccio="b">
```

```
    <nom>Xinxeta</nom>
```

```
    <quantitat>20</quantitat>
```

```
  </producte>
```

```
  <producte codi="a1" seccio="b">
```

```
    <nom>Boli</nom>
```

```
    <quantitat>10</quantitat>
```

```
  </producte>
```

```
  <enviament origen="Sevilla" desti="Barcelona" />
```

```
</comanda>
```

Defineix la versió de XML i format de l'arxiu

Elements o etiquetes

- Es defineixen entre '<' i '>'
- Diferència entre majúscules i minúscules
- Poden tenir altres continguts a dins

S'han de 'tancar' amb '</', '>' si tenen contingut o amb '</>' si no

Exemple d'XML

```
<?xml version="1.1" encoding="UTF-8" ?>
<!DOCTYPE note SYSTEM "Note.dtd">
<comanda>
  <producte codi="a0" seccio="b">
    <nom>Xinxeta</nom>
    <quantitat>20</quantitat>
  </producte>
  <producte codi="a1" seccio="b">
    <nom>Boli</nom>
    <quantitat>10</quantitat>
  </producte>
  <enviament origen="Sevilla" desti="Barcelona" />
</comanda>
```

Atributs

- Es defineixen dins de l'obertura de l'element
- Un element pot tenir, 0 o més atributs
- Format: etiqueta="valor"

XML anidament

Els elements es poden anidar

```
<comanda>
```

```
  <producte>Xinxeta</producte> ← Correcte !
```

```
</comanda>
```

Però no poden trencar la relació de quin element és el que està a contingut en un altre

```
<comanda>
```

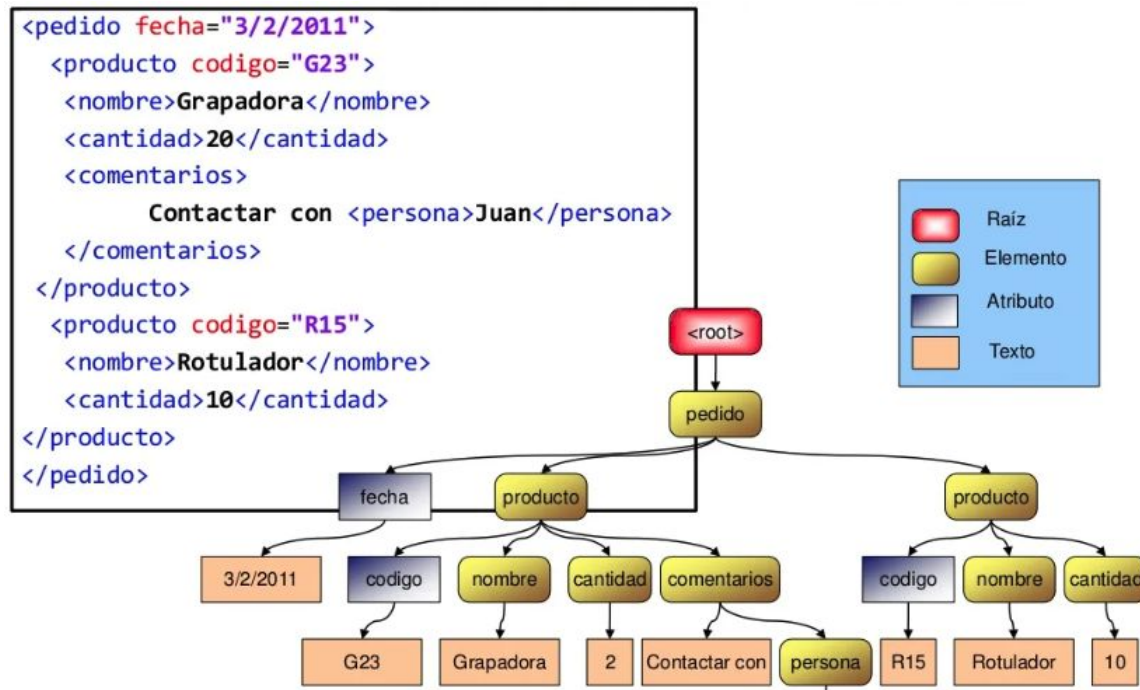
```
  <producte>Xinxeta</comanda>
```

```
</producte>
```

← ERROR, mal anidats!

XML arbre d'elements

Al final, podem veure els documents XML com un conjunt d'informació relacionada en una estructura de arbre



XML, arrel del document

Els documents només poden tenir un element principal (anomenat **arrel**)

```
<?xml version="1.1" encoding="UTF-8" ?>
<!DOCTYPE note SYSTEM "Note.dtd">
<comanda>
  <producte codi="a0" seccio="b">
    <nom>Xinxeta</nom>
    <quantitat>20</quantitat>
  </producte>
  <enviament origen="Sevilla" desti="Barcelona" />
</comanda>
```

Correcte !

```
<?xml version="1.1" encoding="UTF-8" ?>
<!DOCTYPE note SYSTEM "Note.dtd">
<comanda>
  <producte codi="a0" seccio="b">
    <nom>Xinxeta</nom>
    <quantitat>20</quantitat>
  </producte>
</comanda>
<comanda>
  <enviament origen="Sevilla" desti="Barcelona" />
</comanda>
```

ERROR, dues arrels!

XML elements buits autotancats

Normalment els elements tenen altres elements a dins i es tanquen amb `</nom>`

```
<producte codi="a0" seccio="b">  
  <nom>Xinxeta</nom>  
  <quantitat>20</quantitat>  
</producte>
```

Però quan només tenen atributs i no tenen res a dins es poden tancar amb `/>`

```
<nom text="Albert"></nom>  
<nom text="Albert" />
```

XML comentaris

Els comentaris es posen entre `<!-- i -->`

```
<producte codi="a0" seccio="b">  
    <!-- Els productes no són radioactius -->  
    <nom>Xinxeta</nom>  
    <quantitat>20</quantitat>  
</producte>
```

Els comentaris no han de tenir informació important del document, perquè l'interpret XML els ignora

Han de tenir explicacions entenedores pels humans que treballin amb el document

XML caràcters especials

Com que amb XML els caràcters <, >, “, ‘ i & estan reservats pel codi, quan els hem de fer servir entre el text hem de posar les següents equivalències:

< s'escriu amb <

> s'escriu amb >

“ s'escriu amb "

‘ s'escriu amb '

& s'escriu amb &

```
<producte codi="a0" seccio="b">
```

```
  <nom>Pinyol i &quot;arxinacs&quot;</nom>
```

```
  <marca>Ben&Joe</marca>
```

```
</producte>
```

Representa els textos:

Nom: Pinyol i “arxinacs”

Marca: Ben&Joe

XML seccions CDATA

Les seccions CDATA permeten escriure codi que no serà processat

<element>

<material>fusta</fusta>

<![CDATA[

Normalment les seccions CDATA contenen instruccions específiques d'una aplicació, però que els intèrprets de XML ignoren completament

]]>

</element>

XML, instruccions de processament

Les instruccions de processament van entre `<? i ?>` l'interpret valida que segueixin el format establert. Exemples:

```
<?sort ascending ?>
```

```
<?php 2+3 ?>
```

```
<?xml-stylesheet type="text/xsl" href="estils.css" ?>
```

```
<?xml version="1.1" ?>
```

XML, validacions

Hi ha dos maneres de validar els documents XML

Document ben format: que compleix els estàndards de XML (té només una arrel, els atributs estan al declarar l'element, ...)

Document vàlid: si està ben format i els elements encaixen amb la declaració que es fa al DTD