UF1. NF1. INTRODUCCIÓ ALS LLENGUATGES DE MARQUES. LLENGUATGE XML

Índex

1. LLENGUATGES DE MARQUES

Els llenguatges de marques estan per tots els costats, en el nostre ordinador existeixen nombrosos arxius que mai imaginaríem que estan escrits amb un llenguatge de marques.

1.1. Què és un llenguatge de marques?

Un llenguatge de marques és un llenguatge en el qual les seves parts es diferencien entre si mitjançant senyals. És una forma de codificar un document que, juntament amb el text, incorpora etiquetes o marques que contenen informació addicional sobre l'estructura del text o la seva presentació.

Les marques també estan formades de text, però que és interpretat quan es mostra el document, i solen anomenar-se també etiquetes.

El llenguatge de marques més popular és HTML.

1.2. Classificació

Existeixen tres classes de llenguatges de marques:

- **Marques de presentació**: aquestes marques indiquen el format-marc del text. El seu ús comença a reduir-se atès que és poc flexible, especialment en grans projectes.
- **Marques de procediments**: aquestes marques s'utilitzen per a la presentació del text, interpretant-se cadascuna en l'ordre que en apareixen. Per exemple, la marca que s'agrega immediatament abans d'un text perquè es vegi en negreta. Després ha d'existir la marca corresponent que acabi o tancament la negreta. Altres marques de procediments poden ser centrar text, canvi de grandària de font, canvis d'estils, etc. Alguns llenguatges de marques de procediment són nroff, troff, TeX, PostScript HTML, etc.
- **Marques descriptives**: També cridades marcat descriptiu, o semàntic. Aquí s'utilitzen les marques per a descriure fragments de text sense especificar com han de representar-se. Alguns llenguatges dissenyats per a això són el SGML i el XML. En els llenguatges de marques descriptives el format està separat del contingut, permetent flexibilitat a l'hora de reformatejar un text.

1.3. Exemples de llenguatges de marques

- Darwin Information Typing Architecture (DITA)
- DocBook
- Extensible HyperText Markup Language (XHTML)
- Extensible Markup Language (XML)
- Standard Generalized Markup Language (SGML)
- HyperText Markup Language (HTML)
- Lilypond (sistema per a notació musical)
- Maker Interchange Format (MIF)
- Mathematics Markup Language (MathML)
- Microsoft Assistance Markup Language (MAML)
- Music Extensible Markup Language (MusicXML)

- Rich Text Format (RTF)
- S1000D (Especificació internaciol per a documentació tècnicarelacionada amb l'àrea comercial i militar).
- TeX, LaTeX (utilitzat generalment en matemàtiques i publicacions acadèmiques).
- Text Encoding Initiative (TEI). (format XML per publicacions digitals)
- Wireless Markup Language (WML), Wireless TV Markup Language (WTVML)
- XHTML Basic (subconjunt de XHTML per a dispositius portàtils, per reemplaçar a WML, XHTML MP i C-HTML).

1.3.1. SGML (Standard Generalized Markup Language)

És un estàndard internacional publicat per la ISO (Organització Internacional d'Estàndards). SGML va establir dues regles principals:

- La sintaxi que havia d'utilitzar-se per a dissenyar un conjunt de marques aplicables a cada tipus de document.
- La forma en la qual s'han d'intercalar marques en el text d'un document per a identificar les seves parts estructurals.

El conjunt de marques que es poden utilitzar amb cada tipus de document constituïx una DTD o definició de tipus de document. El concepte de DTD s'ha reutilitzat també en li llenguatge XML.

1.3.2. HTML (Hyper Text Markup Language)

És una aplicació del llenguatge SGML que especifica com s'han de codificar els documents per a distribuir-los en la Web. El seu origen es remunta a començament dels anys noranta, quan Tim Berners Llig, del CERN, van desenvolupar el World Wide Web.

HTML era independent de plataformes maquinari o programari específiques, el que li convertia en la solució idònia per als problemes d'intercanvi de documentació en format electrònic.

HTML presenta algunes limitacions:

- Incapacitat per a presentar les característiques tipogràfiques i presentacions complexes dels documents.
- La falta de capacitat expressiva del llenguatge, degut al fet que només es pot utilitzar un nombre limitat de marques predefinides en l'especificació.

1.3.3. XML (eXtensible Markup Language)

Va començar a desenvolupar-se en 1996 pel W3C (el comitè encarregat de normalitzar i controlar el desenvolupament dels estàndards per a la Web) amb un clar propòsit: dissenyar un llenguatge de marques optimitzat per a poder ser utilitzat en Internet. Havia de combinar la simplicitat d'HTML, amb la capacitat expressiva de SGML. XML és un llenguatge per a representar dades i informació. XML es diu extensible perquè podem crear les nostres pròpies etiquetes, en lloc d'estar subjectes a un conjunt d'elles com ocorre en HTML.

1.3.4. XHTML (eXtensible Hyper Text Markup Language)

Es presenta com una redefinició del llenguatge HTML utilitzant la sintaxi de XML. El W3C ho va publicar en l'any 1999 i és l'última versió del llenguatge HTML, després de la versió 4.0.

S'han inclòs totes les etiquetes HTML però seguint les directrius de XML. És a dir que , entre altres coses, cada etiqueta que s'obri ha de tancar-se amb un ordre.

2. XML

2.1. Un petit exemple

Abans de res , veurem un petit exemple d'un document XML, anomenat casablanca.xml

```
<?xml version="1.0"?>
<movies>
      <movie>
            <title>Casablanca</title>
            <director>Michael Curtiz</director>
            <actors>
                  <actor>Humphrey Bogart</actor>
            </actors>
      </movie>
      <movie> ... </movie>
</movies>
```

2.2. Característiques

XML ha passat des considerat una alternativa a HTML, a convertir-se en el llenguatge amb major impacte en el desenvolupament d'aplicacions informàtiques per a Internet i Intranet.

Les principals característiques que oferix XML són:

- **Conjunt de marques obertes i ampliables**: podem definir noves marques per a codificar l'estructura i contingut de diferents tipus de documents.
- **Distinció entre l'estructura i la presentació dels documents**: les marques d'un document XML no indiquen gens sobre com ha de presentar-se el document. Per a indicar com s'ha de presentar un document en pantalla o en paper,serà necessari crear una fulla s'estilo a part i associar-la al document. Per a crear fulles d'estil per a documents XML disposem de dues alternatives: les fulles d'estil CSS, ja utilitzades amb pàgines HTML, i les XSL, dissenyades específicament per a XML.
- **Gestió d'hipervincles avançada**: els hipervincles XML poden crear una relació entre més de dos documents.
- **Modularitat**: diem que un document és modular quan està format per diferents arxius XML que es presenten com si es tractés d'un únic document.

2.3. Estructura

2.3.1. Parts d'un document XML

Un document XML pot presentar tres parts diferents: el pròleg, el cos i l'epíleg.

PRÒLEG

És una part opcional.

La primera línia s'encarrega de presentar el tipus de document, la versió de la norma a la qual

s'adhereix, la codificació de caràcter (US-ASCII, UTF-8, UTF-7, UCS-2, EUCJP, Big5, ISO-8859-1, ISO-8859-7, etc. i altres característiques.

```
<?xml version=”1.0” encoding=”ISO-8859-1”?>
```

L'etiqueta anterior està composta de forma diferent a la resta de les etiquetes que apareixen en el document. Als símbols “<” i “>” se'ls afegix un símbol de tancament d'interrogació. Aquestes instruccions es coneixen com **instruccions de procés**.

El conjunt de caràcters ISO-8859-1 referenciat en aquesta declaració inclou tots els caràcters usats en la majoria dels llenguatges d'Europa Occidental. Si no s'especifica encoding, el parser XML assumeix que els caràcters pertanyen al conjunt UTF-8, un estàndard Unicode que suporta virtualment cada caràcter i ideograma de qualsevol llenguatge del món.

La segona línia defineix el tipus de document, especificant que DTD valida i defineix les dades que conté.

Exemples de pròlegs:

```
<?xml version="1.0" encoding="UTF-7"?>
<!DOCTYPE mensaje SYSTEM “mensaje.dtd”>

<?xml version="1.0" encoding=“ISO-8859-1"?>
<!DOCTYPE HTML PUBLIC “-/ /W3C/ /DTD HTML 3.2 Final/ /EN”>

<?xml version="1.0" encoding=“Big5"?>
```

Altra instrucció de procés seria:

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

COS

Les dades que un document XML ens oferix estan en el que es coneix com cos. És un arbre únic d'elements marcats, amb imbricament estricte.

En el nostre exemple, és tot el que va entre <movie> i </movie>, aquest element es coneix com l'arrel de l'element.

EPÍLEG

Situat a continuació del cos, pot estar compost d'instruccions de procés com les del pròleg, llevat de declaracions xml o de tipus de document.

2.3.2. Elements

Els elements XML poden tenir contingut (més elements, caràcters, o ambdós alhora), o bé ser elements buits.

Un element amb contingut és, per exemple:

<nom>Joan Reverter</nom>

Per diferenciar entre els diferents elements que composan un document XML hem d'utilitzar etiquetes. Les etiquetes estan compostes per un nom i uns atributs, ha d'existir una d'obertura i altra de tancament i deuen estar correctament situades.

Totes les etiquetes que apareixen en els nostres documents XML han de seguir unes normes molt senzilles:

- Deuen estar delimitades pels símbols < i >.
- El nom de l'etiqueta ha de començar per qualsevol lletra, un guió baix (_) o dos punts (:). A partir d'aquest moment, podem utilitzar també el guió (-)
- Les majúscules i les minúscules són diferents, una mica que no és cert en HTML, on <BR> i <br> són la mateixa etiqueta. En un document XML i color="2A6A6E"són etiquetes diferents.
- Cada etiqueta d'obertura ha de tenir una de tancament.

Les etiquetes també poden estar buides. Un element buit, és el qual no té contingut. Es poden escriure de dues maneres diferents:

```
<actor></actor>
```

o de forma abreujada:

```
<actor/>
```

La sintaxi d'HTML permet etiquetes buides tipus `<hr>`. En HTML reformulat perquè sigui un document XML bé format, s'hauria d'usar `<hr/>`.

2.4. Comentaris

Un document XML pot contenir anotacions en forma de comentari. Els comentaris no són part del contingut d'informació del document, i poden ser ignorats pels processadors XML. Els comentaris s'escriuen com:

```
<!-- ...texte del comentari... -->
```

El text d'un comentari no pot contenir la seqüència --.

2.5. Atributs

Les etiquetes poden aprofitar-se per a incloure altres dades, utilitzant atributs. Si has vist HTML ja coneixeràs una mica dels atributs. Per exemple, l'etiqueta HTML per a crear un enllaç a altra pàgina podria ser el següent:

```
<a href=http://www.google.es>Enlace a yahoo</a>
```

En aquest cas, l'etiqueta a se refereix a un enllaç i href és un atribut.

En XML no és diferent a HTML, l'etiqueta denota el nom de l'element, i l'atribut les seves propietats. La diferència és que XML és més estricte que HTML.

```
<gat raça="Persa">Micifú</gat>
```

Igual que en altres cadenes literals de XML, els atributs poden estar marcats entre cometes verticals (') o dobles ("). Quan s'usa un per a delimitar el valor de l'atribut, l'altre tipus es pot usar dintre.

```
<verdura clase="pastanaga" longitud='15" i mig'>
<cita texte="'Hola! Bon dia', va dir ell">
```

Els atributs s'usen per a diferenciar entre elements del mateix tipus. Per exemple:

```
<gat raça="Persa">Micifú</gat>
<gat raça="Siames">Milu</gat>
```

2.6. Imbricament d'elements

El contingut dels elements no està limitat a només text; els elements poden contenir altres elements, que al seu torn poden contenir text o altres elements, i així successivament.

Un document XML és un **arbre d'elements**. No hi ha límit per a la profunditat de l'arbre, a més de que els elements poden repetir-se.

A l'element que està dintre d'un altre se li coneix com **fill**. L'element en el qual aquest està contingut es coneix com **pare**.

```
<name>
    <first-name>Joan</first-name>
    <surname>Reverter</surname>
</name>
```

El document ha de tenir un sol element arrel. Dit d'una altra manera, tots els elements del document han de ser fills d'un sol element.

2.7. Entitats predefinides

En XML 1.0, es defineixen cinc entitats per a representar caràcters especials i que no s'interpretin com marcat pel processador XML. És a dir, que així podem usar el caràcter "<" sense que s'interpreti com el començament d'una etiqueta XML, per exemple.

Les entitats són:

Entitat	Caràcter
&amp;	&
&lt;	<
&gt;	>
&apos;	'
&quot;	"

2.8. Secccions CDATA

Existeix altra construcció en XML que permet especificar dades, utilitzant qualsevol caràcter, especial o no, sense que s'interpreti com marcat XML. Les seccions CDATA s'utilitzen per a indicar-li a l'analitzador que un determinat fragment del document XML no ha de ser analitzat. Existeixen caràcters que no poden ser inclosos directament com contingut, com són (<) (>) o (&), per als quals hem d'usar expressions equivalents.

En ocasions, utilitzar aquests equivalents no resulta pràctic ni desitjable. El cas més freqüent que se sol presentar és aquell en el qual volem incloure codi font en algun llenguatge de programació, com per exemple Javascript, o fins i tot utilitzar HTML.

Un element CDATA ha de començar amb `<![CDATA[` i acabar amb `]]>`

```
<example>
<![CDATA[
<HTML>
<HEAD><TITLE>Rock & Roll</TITLE></HEAD>
]]>
</example>
```

2.9. Elaboració de documents ben formats

Es diu que un document XML està bé format quan compleix les regles sintàctiques indicades. Els processadors XML poden rebutjar qualsevol document que no estigui bé format.

2.10. Utilització d'espais de noms en XML

El poder de XML prové de la seva flexibilitat, del fet que puguem definir les nostres pròpies etiquetes per a descriure les nostres dades. Vegem un exemple:

```
<address>
      <name>
            <title>Sra.</title>
            <first-name>Maria</first-name>
            <surname>Martin López</surname>
      </name>
      <street>Mayor 14</street>
      <city>Madrid</city>
      <postcode>34829</postcode>
<address>
```

El document inclou l'element <title> per al títol o tractament de cortesia, una elecció perfectament raonable com nom d'un element. Si creiem una llibreria online, podria triar el crear un element <title> per a emmagatzemar el títol d'un llibre. Totes són eleccions raonables, però totes elles creen elements amb el mateix nom. Com saber si, donat un element <title> es refereix a una persona o a un llibre ? Amb els **namespaces**.

Per a usar un namespace, definim un prefix namespace i el mapegem a una cadena particular.

Així és com s'haurien de definir prefixos namespace per als nostres dos elements :

```
<?xml version="1.0"?>
<user_summary
  xmlns:address="http://www.xyz.com/address/"
  xmlns:books="http://www.zyx.com/books/"
>
... <address:name><title>Mrs.</title> ... </address:name> ...
... <books:title>El Senyor dels Anells</books:title> ...
```

En aquest exemple, els dos prefixos del namespace són: address i books.

Cal adonar-se que la definició d'un namespace per a un element particular significa que tots els elements fills pertanyen al mateix namespace. El primer element <title> pertany al namespace address degut al fet que el seu element pare <address:name>, ja pertanyia a aquest namespace.

El punt final: La cadena d'una definició de namespace és solament una cadena. Sí, aquesta cadena sembla una URL, però no ho és. Podríem definir xmlns:address=”joan" i funcionaria exactament igual. L'única cosa que importa sobre la cadena del namespace és que sigui única; per això la major part de les definicions de namespace semblen URLs. Els parser XML no van a l'adreça http://www.zyx.com/books/ per a buscar una DTD o un Esquema, simplement usen aquests textos com cadenes.

2.11. Eines de programari

Disposem de les següents eines per a comprovar que un document aquest bé format:

- Un client Web
- Un editor de XML
- Un validador online: http://validator.w3c.org