

Tema 10

Gestió i recuperació d' informació. XPath

Objectius

- Utilitzar els documents XML com a mètode per a l' emmagatzematge de la informació.
- Identificar l' àmbit d' aplicació de l' emmagatzematge d' informació en format XML.
- Comprendre els avantatges i els inconvenients d' emmagatzemar informació en format XML.
- Identificar les característiques dels sistemes d' emmagatzematge d' informació.
- Utilitzar les tècniques de recerca d' informació en documents XML.
- Conèixer i utilitzar el llenguatge XML de consulta d'informació XPath (*XML Path*).
- Utilitzar eines de consulta d' informació en format XML.

Introducció

En els temes anteriors s' han estudiat les característiques generals dels documents XML en relació amb que estiguessin ben formats i fossin vàlids. Una de les característiques principals dels documents XML és la seva capacitat per emmagatzemar informació estructurada en format de text. En aquest tema i en el següent s' estudien algunes de les tecnologies XML que permeten recuperar la informació emmagatzemada en format XML, utilitzant per a això llenguatges de consulta específics. En aquest tema s'aborda l'estudi del llenguatge de consulta XPath (*XML Path*) i en el tema següent s'estudiarà el llenguatge de consulta XQuery (*XML Query*). L'ús del llenguatge de consulta XPath (*XML Path*) és senzill i la seva importància rau en el el que sol emprar-se en combinació amb altres tecnologies XML, com, per exemple, la tecnologia XSLT (*eXtensible Stylesheet Language Transformations*).

Índex

10.1 EMMAGATZEMATGE I CONSULTA D' INFORMACIÓ EN DOCUMENTS XML	2
10.2 ESTRUCTURA JERÀRQUICA EN ARBRE	3
10.3 LLenguatge DE CONSULTA XPATH	4
10.4 EINES PER A LA CONSULTA D' INFORMACIÓ EN FORMAT XML	13
Glossari de termes	14

10.1 EMMAGATZEMATGE I CONSULTA D' INFORMACIÓ EN DOCUMENTS XML

La utilització del llenguatge de marques XML, i les seves tecnologies associades, permeten establir un sistema d' emmagatzematge d' informació fiable i ordenat. Es pot considerar que els documents XML que tinguin una mateixa estructura, tindran també un contingut d' informació homogeni, és a dir, que serà semblant. De manera que l' estructura dels documents XML és la que garanteix la compatibilitat de la informació emmagatzemada. Per aquest motiu, precisament, cal disposar de les tècniques i eines que permetin validar els documents XML creats, mitjançant declaracions DTD o altres tècniques de validació, per poder comprovar l' adaptació del contingut d' informació a l' estructura preestablida. Un cop un document XML s' ha validat, i per tant la seva estructura és l' adequada, aleshores es disposa d' un sistema d' emmagatzematge d' informació en format XML amb la capacitat de poder extreure aquesta informació del document mitjançant la utilització d' un llenguatge de consulta XML.

L' emmagatzematge d' informació és, actualment, una part fonamental de la gestió i administració de les organitzacions. L' ús del llenguatge XML està permetent trencar barreres i crear noves formes d' emmagatzemar i administrar informació. Aquestes noves formes, substancialment, s'aparten de l'ús tradicional de bases de dades relacionals basades en emmagatzematge en format tabular (taules, files o registres i columnes o camps).

Actualment, els sistemes d' emmagatzematge poden ser classificats en dues categories: els centrats en les dades i les centrades en els documents. Els sistemes d' emmagatzematge centrats en dades són altament estructurats, amb tipus de dades de mida limitada, regles poc flexibles per a camps opcionals i contingut basat en registres. Els sistemes d' emmagatzematge centrats en documents tendeixen a ser més impredecibles en mida i en contingut. Les bases de dades relacionals solen ser millors per tractar amb els requeriments d' emmagatzematge centrats en les dades, mentre que els sistemes d' emmagatzematge basats en format XML són típicament millors per emmagatzemar dades centrades en els documents. En l' actualitat, existeixen sistemes gestors de bases de dades específiques per al tractament i la recuperació de la informació en documents XML, com ara BaseX. Aquests sistemes gestors de bases de dades específiques que suporten emmagatzematge en format XML solen rebre el nom de Bases de dades XML natives o NXD (*Native XML Database*). Les fronteres entre les dues categories de sistemes mai han quedat del tot clares. Tant és així que, actualment, les bases de dades relacionals solen també oferir suport i eines per al tractament i consulta d' informació emmagatzemada en format XML.

La utilització de documents XML com a sistema d' emmagatzematge no es queda només en la capacitat d' emmagatzemar informació, sinó també en la capacitat per extreure la informació. Així, si un document està ben format i és vàlid, serà possible localitzar els seus elements i la informació que contenen aquests, de forma senzilla. Efectivament, si un document XML té una estructura ben definida és possible accedir a la informació continguda per, posteriorment, poder tractar-la de la manera més adequada a través dels processaments necessaris. El consorci W3C va ser conscient de tot això i va publicar les especificacions dels dos llenguatges de consulta d'informació sobre documents XML: XPath (*XML Path*) i XQuery (*XML Query*).

- XPath. És un llenguatge bàsic i senzill d' expressions de consulta que permet accedir a parts d' un document XML.
- XQuery. És un altre llenguatge de consultes més complex que permet la manipulació de la informació emmagatzemada en documents XML. El llenguatge XML de consulta XQuery fa ús del llenguatge de consulta XPath.

10.2 ESTRUCTURA JERÀRQUICA EN ARBRE

Per comprendre el sistema d' emmagatzematge d' informació en documents XML i, per tant, el funcionament dels llenguatges de consulta, tant amb XPath com amb Xquery, cal considerar un document XML com un arbre de nodes. Aquesta forma de veure els documents XML dóna lloc al que es coneix com a estructura jeràrquica en arbre dels documents XML.

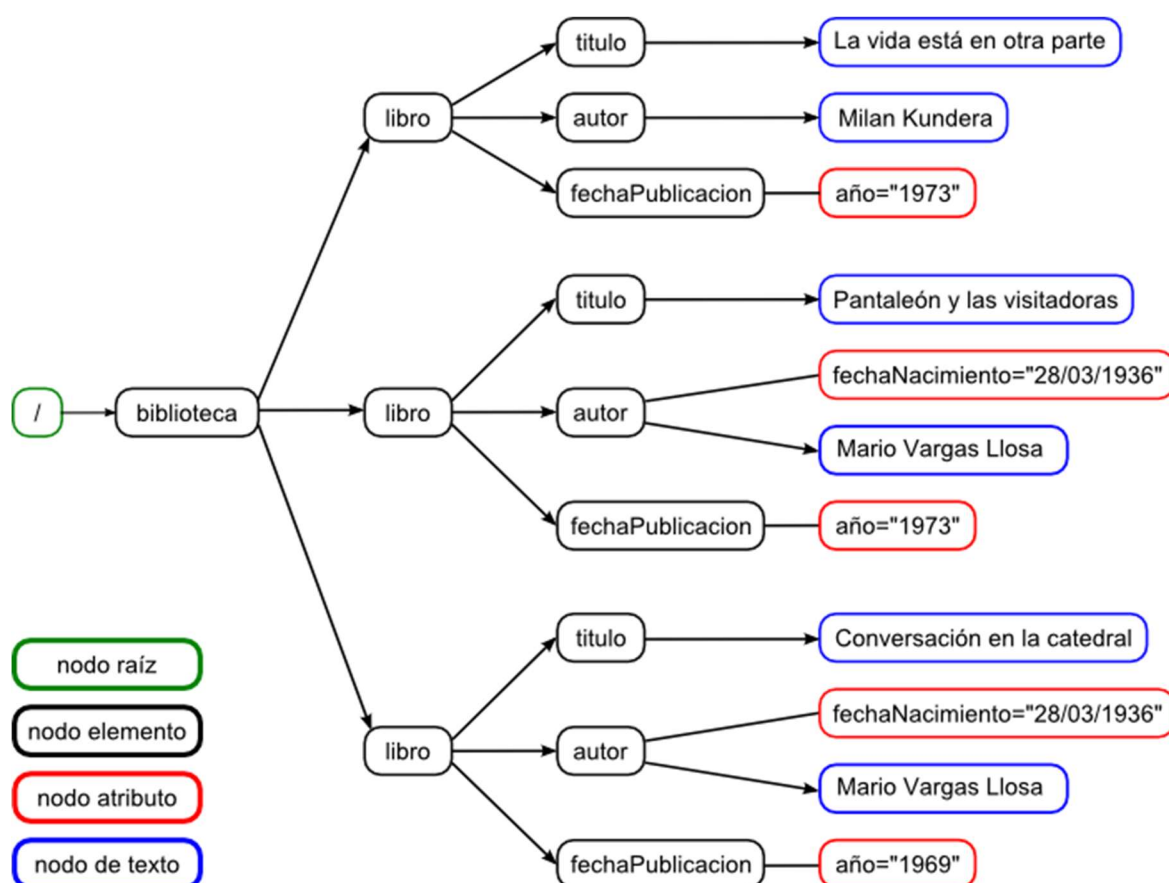
Un document XML pot representar-se com un **arbre dirigit**, per expressar l' estructura jeràrquica existent entre els seus elements o components. Així, es pot considerar, per exemple, que els elements són nodes i que un dels elements que el formen és l' element pare que conté la resta d' elements. Però en els llenguatges de consulta de documents XML no només els elements són nodes, en realitat hi ha set tipus de nodes:

- Arrel.
- Element.
- Atribut.
- Text o informació textual continguda en un element.
- Comentari.
- Instrucció de processament.
- Espai de noms.

Per exemple, el següent document XML:

```
<?xml version="1.0" encoding="utf-8" ?>
<biblioteca>
  <llibre>
    <titulo>La vida està en una altra part</titulo>
    <autor>Milan Kundera</autor>
    <fechaPublicacion any="1973"/>
  </llibre>
  <llibre>
    <titulo>Pantaleón i les visitadores</titulo>
    <autor dataNacimiento="28/03/1936">Mario Vargas Llosa</autor>
    <fechaPublicacion any="1973"/>
  </llibre>
  <llibre>
    <titulo>Conversació a la catedral</titulo>
    <autor dataNacimiento="28/03/1936">Mario Vargas Llosa</autor>
    <fechaPublicacion any="1969"/>
  </llibre>
</biblioteca>
```

es pot representar gràficament mitjançant un arbre dirigit que es pot apreciar en la següent il·lustració. Es pot apreciar que el node arrel (verd), que els nodes element (negre), que els nodes atribut (vermell) i que els nodes de text (blau) estan diferenciats entre si. Cal tenir en compte que els nodes atribut i de text no poden tenir nodes descendents. En realitat, els nodes atribut i de text es consideren com a fills del node que representa l' element corresponent. Les expressions de consulta XPath solen actuar sobre els nodes element, però, per referir-se als atributs o al text s' utilitzen notacions especials.



No existeix una norma sobre la forma de representar el contingut de l' arbre. En aquest cas d' exemple s' ha reflectit el nom dels elements i el seu valor contingut dins d' ell. D' aquesta manera, es pot apreciar, clarament, l' estructura jeràrquica dels elements, així com, els valors dels atributs i del text dels elements que, a més, no tenen nodes descendents.

10.3 LLENGUATGE DE CONSULTA XPATH

En l' apartat anterior, s' ha estudiat l' estructura jeràrquica en forma d' arbre que posseeix un document XML. Així, en un document XML hi ha elements que contenen altres elements i altres que contenen informació textual i que poden contenir, a més, atributs.

Atesa l' estructura jeràrquica en arbre dels documents XML, s' usa XPath per recórrer els elements del document i extreure la informació continguda. XPath és un estàndard de W3C que defineix un conjunt de regles per referenciar parts d' un document XML, de manera que es tracta d' un llenguatge d' expressions de consulta que es fa servir per buscar i trobar informació en documents XML. Cal tenir en compte que XPath s'utilitza sempre en el context d'un llenguatge amfitrió, per exemple, amb el llenguatge XSLT (*eXtensible Stylesheet Language Transformations*) que permet realitzar transformacions sobre documents XML. De manera que XPath no s'utilitza de manera independent, sinó de forma conjunta amb altres tecnologies.

Les característiques més significatives del llenguatge XPath són les següents:

- Proporciona una sintaxi específica per referenciar parts d' un document XML.
- Utilitza expressions que re referencia les rutes, o els camins o *paths* a recórrer fins a assolir un determinat element dins l' estructura jeràrquica en arbre d' un document XML. Les recerques de la informació continguda en els elements XML es defineixen mitjançant expressions que fan referència a aquestes rutes.
- Conté una llibreria de funcions.
- És un estàndard de W3C.

En resum, XPath és un llenguatge d'expressions de consulta que tracta un document XML com un arbre de nodes. De manera que utilitza la definició d' expressions de recerca de tipus ruta per recórrer l' arbre, trobar i extreure la informació corresponent del document XML.

Per abordar l'estudi de XPath s'utilitza el següent document XML d'exemple.

```
<?xml version="1.0" encoding="utf-8" ?>
<llibreria>
  <llibre categoria="Infantil">
    <titulo idioma="anglès">Harry Potter</titulo>
    <autor>J.K. Rowling</autor>
    <anyo>2005</anyo>
    <preu>19.99</preu>
  </llibre>
  <llibre categoria="Informàtica">
    <titulo idioma="espanyol">Aprenent XML</titulo>
    <autor>Erik T. Ray</autor>
    <anyo>2003</anyo>
    <preu>39.95</preu>
  </llibre>
  <llibre categoria="Cuina">
    <titulo idioma="anglès">Everyday Italian</titulo>
    <autor>Giada De Laurentiis</autor>
    <anyo>2005</anyo>
    <preu>30.00</preu>
  </llibre>
</llibreria>
```

En l' exemple anterior, es pot observar que l' element arrel es denomina llibreria. L' element arrel inclou tres elements llibre que, al seu torn, inclouen elements que representen característiques del llibre: títol, autor, any de publicació i preu. Aquests elements contenen la informació textual corresponent. A més, es pot apreciar que l' element llibre té un node atribut, anomenat categoria i l' element títol té un node atribut, anomenat idioma.

També, en l' exemple anterior es poden apreciar les **relacions jeràrquiques** entre els elements:

- **Pares.** Cada node d' element o atribut té un node pare. Per exemple, el node llibre és el node pare dels nodes títol, autor, any i preu. I, el node títol és el node pare de l'atribut idioma.
- **Fills:** Tot element té 0, 1 o més fills. Així, els nodes dels elements títol, autor, any i preu són nodes fill de l'element llibre.
- **Germans.** Són els nodes que tenen el mateix pare. Per exemple, títol, autor, any i preu són nodes germans.
- **Ancestres.** Es denominen nodes ancestres tots els nodes superiors dins l' estructura jeràrquica del document, és a dir, el pare d' un node, el pare del pare, etc. Per exemple, els nodes llibre i llibreria són els nodes ancestres del node títol.
- **Descendents.** Es denominen nodes descendents tots els nodes inferiors dins l' estructura jeràrquica del document, és a dir, el fill d' un node, el fill del fill, etc. Per exemple, els nodes llibre, títol, autor, any i preu són nodes descendents del node llibreria.

Selecció de nodes

Les expressions XPath de selecció de nodes, que també reben el nom d' expressions d' adreçament, poden ser absolutes o relatives. Les **expressions XPath absolutes** són aquelles que comencen pel node arrel, mentre que les **expressions XPath relatives** són aquelles que es construeixen respecte a un node determinat, denominat node de context o node actual. La següent taula mostra les expressions XPath més bàsiques que es poden utilitzar per seleccionar nodes.

Selecció	Descripció
<i>nombre_nodo</i>	Selecciona tots els nodes denominats nombre_nodo.
<i>/</i>	Selecciona des del node arrel.
	Selecciona els nodes del document des del node actual que coincideixen amb la selecció, no importa on es trobin.
<i>.</i>	Selecciona el node actual.
<i>..</i>	Selecciona el pare del node actual.
<i>@</i>	Selecciona atributs.

A la taula següent es poden observar diversos exemples d' expressions XPath sobre el document XML d' exemple.

Expressió XPath	Resultat
llibreria	Selecciona tots els nodes amb el nom llibreria.
/llibreria	Selecciona el node arrel llibreria.
/llibreria/llibre	Selecciona tots els elements llibre que són fills del node llibreria.
llibreria/llibre	Selecciona tots els elements llibre que són fills del node llibreria.
llibre	Selecciona tots els elements llibre no importa on es trobin en el document.
llibreria//títol	Selecciona tots els elements títol que són descendents de l' element llibreria, sense importar on es trobin sota l' element llibreria.
títol/..	Selecciona tots els elements llibre que són pare de l' element títol.
@idioma	Selecciona tots els atributs del document denominats idioma.

Selecció de nodes desconeguts. Comodins

Els comodins de XPath es fan servir per seleccionar elements desconeguts.

Comodí	Descripció
*	Representa qualsevol element node.
@*	Representa qualsevol atribut node.
node()	Representa qualsevol node de qualsevol classe.

La següent taula mostra l' aplicació de diversos exemples sobre el document XML d' exemple.

Expressió XPath	Resultat
/llibreria/*	Selecciona tots els nodes fill de l' element llibreria.
/llibreria/llibre/*	Selecciona tots els elements fill de tots els elements llibre de l' element llibreria.
/llibreria/*/títol	Selecciona tots els elements títol que són néts de l' element llibreria, independentment de l' element pare.
*	Selecciona tots els elements del document.
títol[@*]	Selecciona tots els elements títol que tinguin algun atribut.

Predicats

Els predicats s' utilitzen per trobar un node específic, o bé, un node que conté un valor específic. Els predicats estan sempre dins de corxets. A la taula següent es poden observar diversos exemples d' aplicació sobre el document XML d' exemple.

Expressió XPath	Resultat
/llibreria/llibre[1]	Selecciona el primer element llibre que és fill de l' element llibreria.

Expressió XPath	Resultat
<code>/llibreria/llibre[last()]</code>	Selecciona el darrer element llibre que és fill de l' element llibreria.
<code>/llibreria/llibre[last()-1]</code>	Selecciona el penúltim element llibre que és fill del node llibreria.
<code>/llibreria/llibre[position()<3]</code>	Selecciona els primers dos elements llibre que són fills de l' element llibreria.
<code>/llibreria/llibre[preu]</code>	Selecciona tots els elements llibre que són fills de l' element llibreria i que continguin un element preu.
<code>/llibreria/llibre[anyo=2003]</code>	Selecciona tots els elements llibre que són fills de l' element llibreria i que tenen un element anyo amb un valor igual a 2003.
<code>/llibreria/llibre[preu>35.00]</code>	Selecciona tots els elements llibre que són fills de l' element llibreria i que tenen un element preu amb un valor major de 35.00.
<code>/llibreria/llibre[preu>35.00]/titulo</code>	Selecciona tots els elements títol que són fills de l' element llibre i que tenen un element preu amb un valor major de 35.00.
<code>llibre[autor="J.K. Rowling"]</code>	Selecciona tots els llibres l'autor dels quals és "J.K. Rowling"
<code>titulo[@idioma]</code>	Selecciona tots els elements títol que tenen un atribut anomenat idioma.
<code>titulo[@idioma='anglès']</code>	Selecciona tots els elements títol que tenen un atribut anomenat idioma i el valor del qual és anglès.

Cal destacar que no existeix una funció `first()` en contraposició a `last()`, per la qual cosa per accedir al primer element en el seu lloc s' ha d' utilitzar el marcador 1 com a valor del predicat.

Selecció de diverses rutes

Es pot utilitzar l'operador `|` per obtenir la unió de resultats de recerca. Se sol emprar per seleccionar diverses rutes. A la taula següent es mostren diversos exemples d' ús:

Expressió XPath	Resultat
<code>llibre/titulo llibre/preu</code>	Selecciona tots els elements títol i (<i>AND</i>) tots els elements preu de tots els elements llibre.
<code>títol preu</code>	Selecciona tots els elements títol i (<i>AND</i>) tots els elements preu del document.
<code>/llibreria/llibre/titulo preu</code>	Selecciona tots els elements títol dels elements llibre de l' element llibreria i tots els elements preu del document.

Eixos XPath

Un Eix XPath defineix un conjunt de nodes en relació amb el node actual. Són els següents:

Nom de l' Eix	Descripció
ancestor	Selecciona tots els ancestres (pare, avi, etc.) del node actual.
ancestor-or-self	Selecciona tots els ancestres (pare, avi, etc.) del node actual, incloent-hi el node actual.
attribute	Selecciona tots els atributs del node actual.
child	Selecciona tots els fills del node actual. És l' eix per defecte.
descendant	Selecciona tots els descendents (fill, nét, etc.) del node actual.
descendant-or-self	Selecciona tots els descendents (fill, nét, etc.) del node actual, incloent-hi el node actual.
following	Selecciona tot allò hi ha al document després de l' etiqueta de tancament del node actual.
following-sibling	Selecciona tots els germans que hi ha després del node actual.
namespace	Selecciona tots els nodes namespace del node actual.
parent	Selecciona el pare del node actual.
preceding	Selecciona tots els nodes que hi ha abans del node actual, excepte els ancestres, els atribut i els nodes namespace.
preceding-sibling	Selecciona tots els germans que hi ha abans del node actual.
self	Selecciona el node actual.

En una expressió XPath, una ruta de localització de nodes consisteix en un o més noms de nodes separats per caràcters /. Per realitzar-se una recerca XPath, s' avalua pas a pas cadascun dels nodes de la ruta sobre el conjunt actual de nodes. En cadascun d' aquests passos s' avalua:

- L'eix XPath (*axisname*) que defineix la relació a l'arbre, entre els nodes seleccionats i el node actual. L' eix `child::` està establert per defecte, per la qual cosa se sol ometre.
- El test de node (*nodetest*) que identifica un node dins d'un eix XPath.
- Els predicats (*predicat*) existents en l'expressió per afinar la definició del conjunt de nodes seleccionat.

La sintaxi per a un pas de localització de nodes és la següent: **axisname::nodetest[predicate]**. A XPath hi ha dues formes de sintaxi: la completa i l'abreujada. L' especificació d' eixos dona lloc a la sintaxi completa. Per exemple, les expressions següents XPath són equivalents:

Sintaxi completa: `/child::llibreria/child::llibre/child::titulo/attribute::idioma`
Sintaxi abreujada: `/llibreria/llibre/titulo/@idioma`

Es recomana utilitzar sempre la sintaxi abreujada per formar expressions de consulta XPath, perquè és el tipus de sintaxi que se sol emprar habitualment. Tanmateix, per curiositat, la següent taula mostra diversos exemples de sintaxi completa sobre el document XML d' exemple.

Expressió XPath	Resultat
<code>/llibreria/child::llibre</code>	Selecciona tots els nodes llibre que són fills del node llibreria.
<code>títol/attribute::idioma</code>	Selecciona l' atribut idioma dels nodes títol.
<code>llibre/child::*</code>	Selecciona tots els elements que són fills del node llibre.
<code>títol/attribute::*</code>	Selecciona tots els atributs del node llibre.
<code>llibre/child::node()</code>	Selecciona tots els elements que són fills del node llibre.
<code>/llibreria/descendant::llibre</code>	Selecciona tots els nodes llibre que són descendents del node llibreria.
<code>títol/ancestor::llibre</code>	Selecciona tots els nodes llibre que són ancestres del node títol.
<code>títol/ancestor-or-self::llibre</code>	Selecciona tots els llibres ancestres del node títol i el node actual també si és un node llibre.
<code>/llibreria/child::* /child::preu</code>	Selecciona tots els preus que són nets del node llibreria.

Operadors XPath

Una expressió XPATH retorna un conjunt de nodes, un string, un boolean o un número. Per obtenir el resultat desitjat en cada cas, es poden emprar els següents operadors formant part de les expressions XPath.

Operador	Descripció	Exemple
<code> </code>	AND. Unió de resultats.	<code>llibre cd</code>
<code>+</code>	Suma.	<code>6 + 4</code>
<code>-</code>	Resta.	<code>6 - 4</code>
<code>*</code>	Multiplicació.	<code>6 * 4</code>
<code>div</code>	Divisió.	<code>8 div 4</code>
<code>=</code>	Igualtat.	<code>preu = 9.80</code>
<code>!=</code>	No igual (diferent).	<code>preu != 9.80</code>
<code><</code>	Menor que.	<code>preu < 9.80</code>
<code><=</code>	Menor o igual que.	<code>preu <= 9.80</code>
<code>></code>	Major que.	<code>preu > 9.80</code>
<code>>=</code>	Major o igual que.	<code>preu >= 9.80</code>
<code>or</code>	OR	<code>preu = 9.80 or preu = 9.70</code>
<code>and</code>	AND	<code>preu > 9.00 and preu < 9.90</code>
<code>mod</code>	Mòdul, resta de la divisió sencera.	<code>5 mod 2</code>

Funcions XPath

El llenguatge XPath proporciona una llibreria de funcions que es poden utilitzar formant part dels predicats per afinar o millorar la selecció. Per exemple, anteriorment, ja s'han utilitzat la funció `last()`. La següent taula presenta algunes de les funcions disponibles amb XPath.

Funció	Sintaxi	Descripció
count()	<code>count(nodes)</code>	Retorna el nombre de nodes que pertanyen al conjunt de nodes proporcionat.
id()	<code>id(valor)</code>	Selecciona nodes pel seu ID únic.
last()	<code>last(nodes)</code>	Retorna la posició del darrer node del conjunt de nodes.
name()	<code>name()</code>	Retorna una cadena amb el nom del node actual.
position()	<code>position(n)</code>	Retorna l' element que es troba en la posició n.
substring()	<code>substring(valor,a,b)</code>	Retorna una subcadena del valor entre la posició a i b.
translate()	<code>translate(cad,c1,c2)</code>	A la cadena cad, substitueix la subcadena c1 per c2.
concat()	<code>concat(c1,c2)</code>	Retorna una cadena que representa la concatenació de la cadena c1 amb la cadena c2.
contains()	<code>contains(c1,c2)</code>	Retorna veritable (<i>true</i>) si c1 conté c2.
string()	<code>string()</code>	Converteix un objecte en cadena.
false()	<code>false()</code>	Torna <i>true</i> .
true()	<code>true()</code>	Retorna <i>false</i> .
not()	<code>not(argument)</code>	Torna <i>si</i> l'argument és <i>false</i> , i viceversa.
sum()	<code>sum(nodes)</code>	Retorna la suma de valors del conjunt de nodes.
round()	<code>round(argument)</code>	Retorna un sencer amb el valor més pròxim a l'argument.
number()	<code>number(argument)</code>	Retorna el número corresponent a l' argument.
ceiling	<code>ceiling(decimal)</code>	Avalua un nombre decimal i retorna l' enter més petit major o igual que aquest nombre decimal.
floor	<code>floor(decimal)</code>	Avalua un nombre decimal i retorna el nombre sencer més gran menor o igual que aquest nombre decimal.
max()	<code>max(nodes)</code>	Retorna el valor màxim del conjunt de nodes
min()	<code>min(nodes)</code>	Retorna el valor mínim del conjunt de nodes
node()	<code>node()</code>	Representa qualsevol node de qualsevol classe.
text()	<code>text()</code>	Retorna el contingut textual d' un element.
data()	<code>data()</code>	Retorna el contingut textual d' un atribut

Accés al contingut textual d' elements i atributs

Quan s' avalua una ruta de localització de XPath es retorna un conjunt de nodes. Per exemple, en avaluar-se l'expressió: `/llibreria/llibre/titulo`, s'obtenen tots els elements descendents dels nodes `titol`. És a dir, s' obté el següent resultat:

```
<titulo idioma="angles">Harry Potter</titulo>
```

```
<titulo idioma="espanyol">Aprenent XML</titulo>  
<titulo idioma="angles">Everyday Italian</titulo>
```

Per accedir al **valor del text contingut en un node o llista de nodes** resultant d' una expressió XPath, s' empra la funció **text()** de XPath.

Així, per exemple, si s' avalua l' expressió:

```
/llibreria/libre/titulo/text()
```

s' obtindrà el següent resultat:

```
Harry Potter  
Aprenent XML  
Everyday Italian
```

Per accedir al **valor del text contingut en els atributs** dels elements s' empra la funció **data()** de XPath. Cal tenir en compte que, per seleccionar un atribut d' un element s' ha d' incloure el caràcter **@** davant del nom de l' atribut, evidentment, un cop ha estat definida adequadament la ruta de localització de l' atribut corresponent.

Així, per exemple, si s' avalua l' expressió:

```
/llibreria/libre/titulo/@idioma/data()
```

s' obtindrà el següent resultat:

```
anglesos  
espanyol  
anglesos
```

Quan un analitzador de XPath avalua una expressió de consulta produeix com a resultat un text xml que correspon amb la resposta de l' expressió XPath avaluada. En cas que no existís la ubicació establerta en la ruta de l' expressió XPath, es retornaria el conjunt buit.

Els analitzadors de XPath poden actuar de manera diferent a l' hora de presentar els resultats del contingut textual d' elements i atributs, per aquest motiu s' utilitzen les funcions **text()** i **data()** quan es vol accedir al contingut de text inclòs en els elements i els atributs, respectivament. Habitualment, els analitzadors d' expressions XPath poden retornar directament el contingut textual del node o nodes localitzats, o també, poden retornar el contingut textual dels atributs d' aquests nodes. També cal tenir en compte que existeixen analitzadors de XPath que no incorporen aquesta funcionalitat.

A continuació, s' inclouen alguns exemples d' expressions de consulta resolts que es poden executar sobre el document XML d' exemple. La següent taula mostra els resultats produïts en avaluar les expressions XPath d' exemple.

Expressions XPath de l'exemple	Resultat
<code>/llibreria/libre/titulo</code>	<code><titulo idioma="anglès">Harry Potter</titulo></code>

Expressions XPath de l'exemple	Resultat
	<pre><titulo idioma="espanyol">Aprenent XML</titulo> <titulo idioma="anglès">Everyday Italian</titulo></pre>
<code>/llibreria/llibre[1]/titulo</code>	<pre><titulo idioma="anglès">Harry Potter</titulo></pre>
<code>/llibreria/llibre/preu</code>	<pre><preci>19.99</preu> <preu>39.95</preu> <preci>30.00</preu></pre>
<code>/llibreria/llibre/preu/text()</code>	<pre>19.99 39.95 30.00</pre>
<code>/llibreria/llibre/preu[modifica]</code>	<pre><preci>19.99</preu> <preu>39.95</preu> <preci>30.00</preu></pre>
<code>/llibreria/llibre[preu>35]/preu</code>	<pre><preu>39.95</preu></pre>
<code>/llibreria/llibre[preu>35]/titulo</code>	<pre><titulo idioma="espanyol">Aprenent XML</titulo></pre>
<code>/llibreria/llibre[@categoria="Informàtica"]/titulo</code>	<pre><titulo idioma="espanyol">Aprenent XML</titulo></pre>
<code>titulo[@idioma="anglès"]/text()</code>	<pre>Harry Potter Everyday Italian</pre>
<code>/llibreria/llibre[3]/autor/text()</code>	<pre>Giada De Laurentiis</pre>
<code>llibre[titulo="Aprenent XML"]/autor</code>	<pre><autor>Erik T. Ray</autor></pre>
<code>llibre/titulo[contains(text(),"Har")]</code>	<pre><titulo idioma="anglès">Harry Potter</titulo></pre>
<code>llibre[titulo="Harry Potter"]/@categoria/data()</code>	<pre>Infantil</pre>
<code>llibre[anyo="2005"]/@categoria/data()</code>	<pre>Infantil Cuina</pre>

10.4 EINES PER A LA CONSULTA D' INFORMACIÓ EN FORMAT XML

Com s' ha pogut comprovar al llarg del tema, XPath és un llenguatge de consulta que s' utilitza per seleccionar elements i atributs d' un document XML navegant a través de la seva estructura en forma d' arbre. S' utilitzen expressions de consulta Xpath per seleccionar nodes o conjunts de nodes del document XML en funció de determinats criteris de recerca.

Existeixen diversos tipus d' eines informàtiques que inclouen un analitzador de XPath. En l' actualitat, els editors de codi XML específics solen incloure un analitzador de XPath que permet processar expressions de consulta XPath. A més, les eines de desenvolupament XML inclouen funcions de processament XPath. Alguns dels editors de codi i eines de desenvolupament XML més estesos que inclouen XPath són els següents: XML Copy Editor, Jaxe XML Editor, XMLSpy, Oxygen XML, Stylus Studio, Visual Studio, etc.

D' altra banda, hi ha eines informàtiques específiques que faciliten l' extracció d' informació en documents XML utilitzant expressions de consulta XPath. En general, aquestes eines permeten extreure informació emmagatzemada en documents XML a partir d' expressions de consulta XPath i

XQuery. Algunes d' aquestes eines específiques que solen considerar-se com a bases de dades XML natives són: BaseX, eXist, DBDOM, Berkeley DB XML i dbXML.

Actualment, també es poden utilitzar eines on-line per processar expressions de consulta XPath sobre documents XML. Algunes de les quals són les següents:

- Videlibri: <http://videlibri.sourceforge.net/cgi-bin/xidelcgi>
- XPath Tester: <http://www.xpathtester.com/xpath>
- FreeFormatter: <https://www.freeformatter.com/xpath-tester.html>
- Code Beauty: <https://codebeautify.org/Xpath-Tester>

Glossari de termes

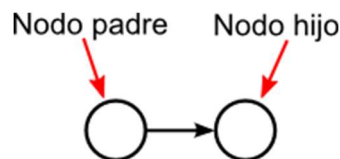
Arbre de nodes. Un arbre és una estructura de dades que equival a un arbre. Al seu torn, un arbre és un cas particular de graf. Els grafs permeten estudiar les interrelacions existents entre els

elements que interactuen entre si unes amb les altres dins d' un conjunt. Aquests conceptes estan definits en la teoria de grafs. El primer article científic relatiu a grafs va ser escrit pel matemàtic suís Leonhard Euler el 1736. Els grafs són àmpliament utilitzats en la disciplina informàtica i la seva aplicació és particularment útil quan s' empen juntament amb els llenguatges de consulta, com XPath, per accedir a la informació estructurada de forma jeràrquica, que està continguda en els documents XML.

Un **graf** és un conjunt d' objectes anomenats nodes o vèrtexs units per enllaços anomenats arcs o arestes. Un **graf dirigit** és un graf en el qual els arcs tenen direcció.

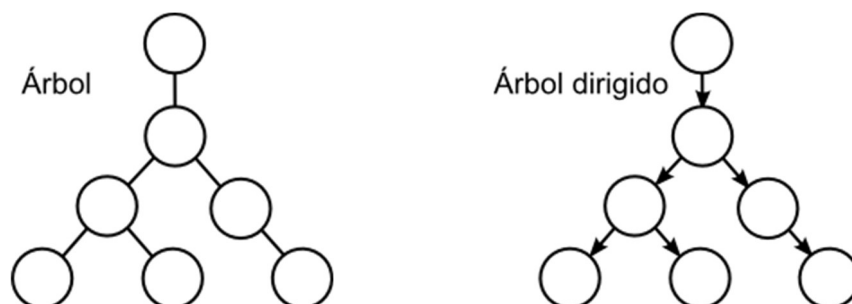


Quan dos nodes estan units per un arc amb direcció, el **node pare** és el node del qual parteix l' arc i el **node fill** és el node al qual arriba l' arc.

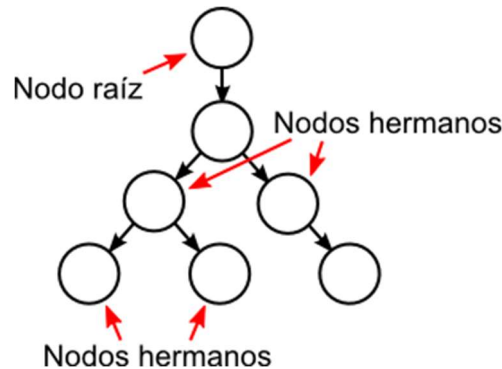


Un **arbre** és un graf en el qual qualsevol parella de vèrtexs estan connectats per un únic camí, és a dir, que no hi ha cicles.

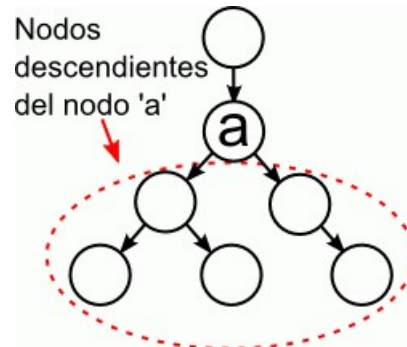
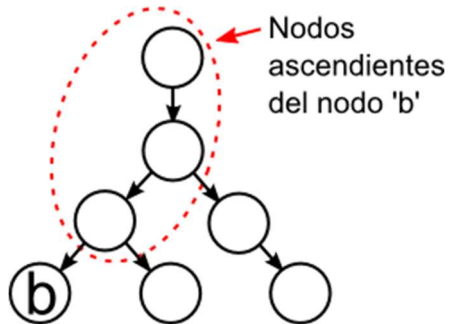
Arbre dirigit. És un arbre en el qual les arestes tenen direcció i tots els nodes menys un, tenen un únic pare.



El **node arrel** d' un arbre dirigit és l'únic node sense pare. Els **nodes germans** són els nodes que tenen el mateix pare.



Els **nodes descendents** d'un node són tots els nodes als quals s' arriba des del node: els fills, els fills dels fills, etc. Els **nodes ascendents** d'un node són tots els nodes dels quals un node és descendent: el pare, el pare del pare o avi, etc.



Graf. És un conjunt d' objectes anomenats nodes o vèrtexs units per enllaços anomenats arcs o arestes.