

UF3

Persistència en SGBD-XML

Annex 07

Instruccions Update XML

Isidre Guixà i Miranda
Institut Milà i Fontanals d'Igualada

Accés a dades

30 de octubre de 2024
Isidre Guixà i Miranda
Institut Milà i Fontanals
C/. Emili Vallès, 4
08700 - Igualada

Aquest material està protegit sota llicència de [Creative Commons BY-NC-SA 3.0](#)

En cas de suggeriment i/o detecció d'error, podeu informar a iguixa@xtec.cat

Castellano

Este material está protegido bajo licencia [Creative Commons BY-NC-SA 3.0](#)
En caso de sugerencia y/o detección de error, dirigir-se a iguixa@xtec.cat

Euskara

Material hau [Creative Commons BY-NC-SA 3.0](#)
Iradokizuna bada eta / edo errore detekzioa, iguixa@xtec.cat zinen joan

Galego

Este material está protexido baixo [Creative Commons BY-NC-SA 3.0](#)
Se suxestión e / ou detección de erros, indica iguixa@xtec.cat

Els continguts de la unitat formativa “Persistència en BD-XML natives” es posen a la pràctica en els SGBD:

- eXist-db, que facilita dues implementacions per a poder efectuar canvis en els documents XML emmagatzemats a la base de dades. Per una banda l'extensió [XUpdate](#) promoguda pel grup XML:DB inactiu des del 2003. Per altra, una extensió [Update](#) basada a la proposada per [Patrick Lehti](#). La documentació específica de la implementació Update facilitada per exist-db es pot trobar [aquí](#).
- Sedna, que facilita l'extensió [Update](#) proposada per [Patrick Lehti](#) per les operacions d'actualització dels documents XML emmagatzemats a la base de dades. La documentació específica de la implementació [Update](#) facilitada per Sedna es pot trobar [aquí](#).
- BaseX, que incorpora una completa implementació de [XQuery Update Facility \(XQUF\)](#) recomanat per W3C. La documentació específica de la implementació XQUF facilitada per BaseX es pot trobar [aquí](#).

El present document és un recull de la sintaxi de les operacions d'actualització XML més habituals en els anteriors SGBD. Per informació més detallada, reviseu la documentació específica de cada SGBD.

Instruccions comprovades en BaseX 10.7 – eXist-db 6.2.0 – Sedna 3.5

INSERT - Inserció de un o varis nodes dins un o varis nodes	
eXist-db	<code>update insert (nodeAInserir seqüènciaNodes) (into following preceding) onInserir</code>
Sedna	<i>En cas de seqüència de nodes, han d'anar entre parèntesi, separats per coma.</i>
BaseX	<i>Si la inserció (sigui d'un o varis nodes) es produeix dins varis nodes, cal utilitzar instrucció FLWOR:</i> <pre>for \$n in onInserir return insert (node nodes) (nodeAInserir seqüènciaNodes) ((as (first last))? into) after before \$n</pre> <i>Però si la inserció és només dins un node, podem no utilitzar el «for»:</i> <pre>insert (node nodes) (nodeAInserir seqüènciaNodes) ((as (first last))? into) after before onInserir</pre> <i>En cas de seqüència de nodes, han d'anar entre parèntesi, separats per coma.</i> <i>into: Afegeix al final dins la ubicació indicada</i>

INSERT - Inserció d'un o varis atributs dins un o varis nodes	
eXist-db	<code>update insert (attribute nomAtribut {'valor'},...) (into following preceding) onInserir</code>
Sedna	<i>Les clàusules following i preceding serveixen per ubicar els atributs després o abans d'un altre atribut</i>
BaseX	<i>Si la inserció (sigui d'un o varis nodes) es produeix dins varis nodes, cal utilitzar instrucció FLWOR:</i> <pre>for \$n in onInserir return insert (node nodes) (attribute nomAtribut {'valor'},...) ((as (first last))? into) after before \$n</pre> <i>Però si la inserció és només dins un node, podem no utilitzar el «for»:</i> <pre>insert (node nodes) (attribute nomAtribut {'valor'},...) ((as (first last))? into) after before onInserir</pre> <i>En cas de seqüència de nodes, han d'anar entre parèntesi, separats per coma.</i>

	<i>També admet la sintaxi</i> <code>attribute {'nomAtribut'}{'valor'}</code> <i>into: Afegeix a l'inici dels atributs</i>
--	--

INSERT - Inserció d'un o varis nodes i atributs dins un o varis node	
eXist-db	<code>update insert (seqüènciaAtributs, seqüènciaNodes) (into following preceding) onInserir</code>
Sedna	<i>Primer cal indicar els atributs i després els nodes. Sedna ho controla. eXist-db no ho controla però si no es posen en "ordre", té efectes indesitjats. Sembla que following i preceding tenen problemes si conviuen atributs i nodes i, en tal situació, millor usar into.</i>
BaseX	<i>Si la inserció (sigui d'un o varis nodes) es produeix dins varis nodes, cal utilitzar instrucció FLWOR:</i> <code>for \$n in onInserir return insert (node nodes) (seqüènciaAtributs, seqüènciaNodes) ((as (first last))? into) after before \$n</code> <i>Però si la inserció és només dins un node, podem no utilitzar el «for»:</i> <code>insert (node nodes) (seqüènciaAtributs, seqüènciaNodes) ((as (first last))? into) after before onInserir</code> <i>Primer cal indicar els atributs i després els nodes. Els atributs queden afegits a l'inici.</i>
Aconseguir ubicar en el lloc adequat atributs i nodes en una única instrucció és complicat. Millor fer-ho amb instruccions separades (atributs per una banda i nodes per una altra).	

RENAME - Canviar el nom d'atribut(s) o node(s)	
eXist-db	<code>update rename nodeOAtributAReanomenar as 'nouNom'</code> <i>o també</i> <code>for \$n in nodeOAtributAReanomenar return update rename \$n as 'nouNom'</code>
Sedna	<code>update rename nodeOAtributAReanomenar on nouNom</code>
BaseX	<code>for \$n in nodeOAtributAReanomenar return rename node \$n as 'nouNom'</code> <i>En cas de ser un únic atribut o node, es podria no utilitzar el «for»:</i> <code>rename node nodeOAtributAReanomenar as 'nouNom'</code>

REPLACE VALUE – Substituir el valor de node(s) o atribut(s)	
eXist-db	<code>update value nodeOAtribut with 'nouValor'</code> <i>o també</i> <code>for \$n in nodeOAtributAReanomenar return update value \$n with 'nouValor'</code>
Sedna	<i>No existeix aquesta instrucció. Cal substituir el node o l'atribut sencer</i>
BaseX	<code>for \$n in nodeOAtributAReanomenar return replace value of node \$n with 'nouValor'</code> <i>En cas de ser un únic atribut o node, es podria no utilitzar el «for»:</i> <code>replace value of node nodeOAtribut with 'nouValor'</code>

REPLACE – Substituir de node(s) o un atribut(s)	
eXist-db	<pre>update replace nodeOAtribut with nouNodeONouAtribut o també for \$n in nodeOAtributAReanomenar return update replace \$n with nouNodeONouAtribut</pre> <p><i>Per indicar el nou atribut cal seguir la norma indicada a la instrucció INSERT. En eXist-db 2.0-5.2., la substitució d'un atribut només substitueix el contingut i no el nom de l'atribut. Error potser solucionat en posteriors versions</i></p>
Sedna	<pre>update replace \$n in nodeOAtribut with nouNodeONouAtribut</pre> <p><i>Per indicar el nou atribut cal seguir la norma indicada a la instrucció INSERT.</i></p>
BaseX	<pre>for \$n in nodeOAtributAReanomenar return replace node \$n with nouNodeONouAtribut</pre> <p><i>En cas de ser un únic atribut o node, es podria no utilitzar el «for»:</i> <pre>replace node nodeOAtribut with nouNodeONouAtribut</pre> <i>Per indicar el nou atribut cal seguir la norma indicada a la instrucció INSERT</i></p>

DELETE – Eliminar node(s) o atribut(s)	
eXist-db	<pre>update delete nodeOAtributAEliminar o també for \$n in nodeOAtributAEliminar return update delete \$n</pre>
Sedna	<pre>update delete nodeOAtributAEliminar</pre>
BaseX	<pre>delete node nodeOAtributAEliminar</pre> <p><i>A diferència de les anteriors instruccions en BaseX, aquesta instrucció serveix per eliminar un o varis nodes/atributs. També es podria utilitzar el «for»:</i> <pre>for \$n in nodeOAtributAEliminar return delete node \$n</pre></p>

WARNING important en eXist-db per les instruccions d'actualització dins FLWOR

When using update within the return clause of a FLWOR expression, be cautious when deleting or replacing nodes that are still being used by the enclosing code. This is because a delete or replace will be processed immediately, and so the deleted or replaced node will no longer be available to the query. *Such actions can corrupt the database!*

For example, the following expression will throw the database into an inconsistent state if //address returns more than one node:

```
for $address in //address
return
  update delete //address
```

However, an expression like the following is safe as it only modifies the current iteration variable. The following example only deletes \$address (the current iteration variable) instead of //address (all addresses in the database):

```
for $address in //address
return
  update delete $address
```

Si la BD es queda en estat inconsistent, es pot intentar solucionar reindexant la col·lecció afectada (opció dins menú File del client Java).