**Captura de XML fuera de la BD**

Las empresas pueden mantener los datos en un sistema relacional y sacar dichos datos en formato XML cuando sea preciso.

**FOR XML**

Es la forma más fácil de coger datos en un formato relacional desde SQL Server.

SELECT column list

FROM table list

WHERE filter criteria

FOR XML RAW | AUTO | EXPLICIT [, XMLDATA] [, ELEMENTS]

[, BINARY BASE64]

**Formas de uso**

Se pueden usar los siguientes modos:

RAW

AUTO

EXPLICIT

Los modos RAW y Auto son los más empleados. La razón de no usarse tanto EXPLICIT es que con los otros modos se cubren normalmente las necesidades.

**Modo RAW**

Devuelve las columnas como atributos y las filas como elementos.

Ejemplo:

USE pubs

GO

SELECT \* FROM Authors FOR XML RAW

Las columnas sin nulos tienen un atributo basado en el elemento.

Si recuperamos datos binarios, necesitaremos especificar BINARY BASE64.

SI RECUPERAMOS UN Schema de datos XML con el XML devuelto, especificaremos XMLDATA.

**Modo AUTO**

La consulta es la misma que en el modo RAW excepto que el nombre del elemento representa la fila como el nombre de la tabla. Ejemplo:

USE pubs

GO

SELECT \* FROM authors FOR XML AUTO

El anidado de los elementos depende del orden en la cláusula SELECT.

No se puede usar GROUP BY, pero podemos usar ORDER BY. La alternativa para un GROUP BY es usar un mandato SELECT anidado para lograr los resultados que queremos, aunque esto tiene algunas implicaciones en el rendimiento.

Al usar JOIN’s, veremos que AUTO anidará los resultados, que probablemente es lo que queremos que ocurra. En el caso de que no lo queramos, necesitaremos usar el modo EXPLICIT.

Ejemplo:

USE pubs

GO

SELECT Publishers.Pub\_Name, Titles.Title, Titles.Price

FROM Titles, Publishers WHERE Publishers.Pub\_ID = Titles.Pub\_ID

FOR XML AUTO

Podemos usar la opción ELEMENTS con FOR XML AUTO.

**Modo EXPLICIT**

Permite controlar completamente la forma en que el XML es generado.

Para ello se usa una tabla universal que describe nuestro documento XML.

Esta tabla consta de una columna para cada valor que queremos que se devuelva así como dos etiquetas adicionales:

* Una que identifica las etiquetas en el XML
* Otra que identifica las relaciones madre-hija.

Convertir un código binario (los datos de una base de datos ) en un XML ( no nativo ).

--- El XML puede ser nativo o no nativo

1.-- El XML no nativo es el que se crea / trabaja a nivel de fichero

2.-- El XML nativo es el que está vinculado a una base de datos (es que está almacenado en

una columna o en una tabla)

Union : sirve para "unir" (lo que realmente hace es mostrar las 2 selects ) dos selects , pero ,

en la clausula select tiene que haber la misma cantidad

de columnas ( example : Select 1 ; EmployeeID , FirstName y Select 2 ; ProductID , Productname

y además estas columnas tienen que ser del mismo tipo de datos.( Example : Varchar ,INT y Varchar,INT )

Posible pregunta de examen: Que sabes hacer con XML?

* Crear el XML.
* Darle unas restricciones al XML mediante un XSD.
* Darle una apariencia atractiva mediante el XSLT.
* Convertir el XML en una columna de una tabla de SQL.(UNTYPED)
* Almacenar en un objeto la columna creada y aplicarle las restricciones del XSD.(TYPED)
* Convertir una tabla de SQL a un formato universal mediante el FOR XML (RAW,AUTO,EXPLICIT).