

BASES DE DATOS XML

Atendiendo al nivel de estructuración, podemos decir que existen tres tipos de datos:

- Datos estructurados. Son los que tienen un formato estricto. Toda la información recogida se ajusta al mismo formato, como por ejemplo: los datos tabulados en filas y columnas de una tabla.
- Datos desestructurados. No tienen ninguna estructura, como un documento de texto o un archivo de vídeo.
- Datos semi-estructurados. Tienen cierta estructura, pero no toda la información recogida tiene la misma forma, y además puede ir variando de manera dinámica.

XML define un conjunto de reglas semánticas que permiten la organización de información de distintas maneras. Es un estándar definido por el W3C y ofrece muchas ventajas, entre ellas: Lenguaje bien formado, Extensible, Autodescriptivo, Intercambiable.

Los tipos de documentos XML que nos podemos encontrar son:

- Documentos centrados en datos, con las siguientes características: Muchos elementos de pequeño tamaño, Estructura bien definida, Datos muy estructurados o semi-estructurados, Dirigidos a utilización automática.
- Documentos centrados en texto, contenido o documentos con las siguientes características: Pocos elementos, Grandes cantidades de texto, Datos poco estructurados, Orientados a ser interpretados por humanos.

Para almacenar documentos XML tenemos las siguientes opciones:

- Almacenamiento directo del fichero. Es una opción pobre ya que las operaciones que podemos hacer sobre ellos son limitadas, las que proporcione el sistema de archivos.
- Almacenar el documento en una base de datos existente (base de datos relacional, orientada a objetos u objeto-relacional). En este caso existen las siguientes posibilidades:
 - *Directamente como una columna tipo binario grande (BLOB) dentro de una tabla.
 - * Mediante mapeo basado en tablas, o basado en objetos.
- Almacenar el documento en una base de datos nativa XML. El documento, tanto si es centrado en datos como en texto, se almacena y recupera de forma intacta. Como veremos, será la mejor opción, sobre todo si el documento está basado en texto.

Podemos hablar de dos tipos de Sistemas de Bases de Datos que soportan documentos XML:

BD XML-compatibles: desglosan un documento XML en su correspondiente modelo relacional o de objetos.
BD XML Nativas: respetan la estructura del documento, permiten hacer consultas sobre dicha estructura y recuperan el documento tal y como fue insertado originalmente.

Una NXD o BD XML debe cumplir las siguientes propiedades: Define un modelo lógico de datos XML, El documento XML es la unidad lógica de almacenamiento, No tiene ningún modelo de almacenamiento físico subyacente concreto.

Podemos diferenciar o clasificar las BD XML nativas en función del tipo de almacenamiento que utilicen, que puede ser: Almacenamiento basado en texto, Almacenamiento basado en el modelo, Soluciones desarrolladas específicamente para la gestión de documentos XML

- El almacenamiento basado en texto consiste en almacenar el documento XML entero en forma de texto (fichero de texto), y proporcionar alguna funcionalidad de base de datos para acceder a él.
- El almacenamiento basado en modelo consiste en definir un modelo de datos lógico, como DOM, para la estructura jerárquica de los documentos XML y almacenar el modelo binario del documento en un almacén existente o bien específico.

Una BD XML tiene una estructura jerárquica organizada en colecciones y documentos XML. La estructura jerárquica comienza con un nodo raíz ('/'), del que parten colecciones y documentos:

- Una colección: Es un conjunto de documentos agrupados, normalmente, en función de la información que contienen. Puede contener otras colecciones.
- Un documento: Información XML Información de otro tipo y entonces se le denomina non-XML data (Datos no-XML).

Gestores nativos XML: [Comerciales] TaminoXMLServer, TEXTML [Libres] eXist, MongoDB.

XPATH

XPath (XML Path Language) es un lenguaje que permite construir expresiones que recorren y procesan un documento XML. XPath permite buscar y seleccionar teniendo en cuenta la estructura jerárquica del XML.

Características de XPath: Es una sintaxis para definir partes de un documento XML, Usa expresiones de ruta para, navegar en documentos XML, Contiene una biblioteca de funciones estándar, Es un elemento importante en XSLT y en XQuery, XPath 3.0 es una recomendación de W3C desde Abril de 2014.

Tipos de nodos

Las partes de un documento XML se denominan nodos. Existen 7 tipos de nodos diferentes:

- Raíz: El nodo raíz o nodo documento (root node) no debe confundirse con el elemento raíz. Éste es más bien el nodo padre virtual del elemento raíz.
- Elemento (element node)
- Atributo (attribute node)
- Texto (text node)
- Espacio de nombres (namespace node)
- Instrucción de procesamiento (processing instruction node)
- Comentario (comment node)

XQUERY

XQuery es un lenguaje de consulta diseñado para extraer información de colecciones de datos expresadas en XML. Podemos decir que XQuery es a XML lo mismo que SQL es a las bases de datos relacionales. El lenguaje XQuery solo proporciona expresiones para la realización de consultas sobre documentos XML, pero no su actualización (inserción, modificación o eliminación de nodos).

Principales características:

- Está basado en el lenguaje XPath, (XML Path Language), y se fundamenta en él para realizar la selección de información y la iteración a través del conjunto de datos XML.
- Es un lenguaje declarativo, lo que significa que, en vez de ejecutar una lista de comandos como un lenguaje procedimental clásico, cada consulta es una expresión que es evaluada y devuelve un resultado, al igual que en SQL.

Los principales tipos de expresiones de XQuery son:

- Expresiones XPath, para navegar por los documentos.
- Expresiones FLWOR (For, Let, Where, Order, Return) para iterar por los elementos de un conjunto de datos.

El modelo de datos en que se sustenta XQuery es el modelo de datos de XPath.

XPath modela un documento XML como una estructura jerárquica en forma de árbol. El árbol está formado por nodos, y hay siete tipos de nodos: raíz, elemento, texto, atributo, espacio de nombres, instrucción de procesamiento y comentario.

Los principales nodos de la estructura jerárquica o en árbol en un documento XML son: Nodo raíz, Nodo elemento, Nodo texto, Nodo atributo.

Una consulta XQuery es una expresión que lee una secuencia de datos en XML y devuelve como resultado otra secuencia de datos en XML, donde:

- Una secuencia es un conjunto ordenado de cero o más ítems.
- Un ítem es cualquier tipo de nodo del árbol XML o un valor atómico.

FLWOR: Se trata de una expresión que permite la unión de variables sobre conjuntos de nodos y la iteración sobre el resultado. Las diferentes cláusulas de una expresión FLWOR son:

For. Permite seleccionar los nodos que se quieren consultar, guardándose su valor en una variable (identificador que comienza por \$). Al conjunto de valores de la variable se le llama tupla.

Let. (opcional). Asocia valores a variables.

Where (opcional). Permite filtrar los resultados según una condición.

Order (opcional). Permite ordenar la secuencia de valores o resultados.

Return. Genera los valores de salida o devueltos.

API XML: DB

Esta API proporciona una interfaz común para bases de datos nativas o habilitadas para XML y admite el desarrollo de aplicaciones portátiles y reutilizables.

Los componentes básicos empleados por XML: DB API son controladores , colecciones , recursos y servicios .

API XQJ

Estas Bases de Datos son una propuesta estandarizada de la interfaz Java para el acceso a BBDD XML nativas basadas en el modelo de datos XQuery. El objetivo de éstas es conseguir un método fácil y estable de acceso. Es un API bastante similar al JDBC de las bases de datos relacionales.

XQuery API for Java (XQJ) es una interfaz de programación de aplicaciones Java pensada para utilizar el lenguaje XQuery para obtener información de BD-XML nativas, de manera parecida a como JDBC es una API pensada para utilizar el lenguaje SQL para acceder a BDR .

MONGODB

MongoDB es la base de datos NoSQL líder y permite a las empresas ser más ágiles y escalables.

Es una base de datos ágil que permite a los esquemas cambiar rápidamente cuando las aplicaciones evolucionan.