

Análisis de esquemas XML [1]

Dirigido a: Administradores de Bases de Datos | Área: Bases de Datos

Autor: Pablo F. Dueñas | Servicios Profesionales Danysoft

Resumen

Un esquema XML describe la estructura de un documento XML. El lenguaje que usa se llama Definición de Esquema XML (XSD por sus siglas en inglés). Mediante este lenguaje se define qué es legal en la construcción de un documento XML. Como sabemos, una de las ventajas de un documento XML es su flexibilidad. Pero esta flexibilidad tiene un precio: el sistema que trata de leer mediante métodos automatizados un documento, va a tener problemas para entenderlo si no hay un acuerdo sobre la estructura, elementos, atributos, orden, etc. que tiene dicho documento. Por lo tanto, al crear un XSD estamos definiendo un contrato mediante el cual es posible el entendimiento entre las partes. El ejemplo más patente son las fechas, que según el país tienen un formato distinto, pero que en el estándar XSD se define uno único. En el caso que vamos a estudiar, la Administración nos explica qué quiere, cómo lo quiere y qué devuelve.

Introducción

Cada vez se está extendiendo más la implantación de los documentos XML. Al ser un sistema abierto y flexible, están sustituyendo en la transferencia de datos a los ficheros planos. Un ejemplo actual lo tenemos en los Certificados de Empresa pedidos por el Servicio Público de Empleo Estatal, en concreto de su Administración Electrónica, conocida como redtrabaj@.

Vamos a estudiar los retos técnicos que supone el cambio en el formato de los ficheros. Ya anticipamos que estos artículos no van a entrar en la normativa legal –por lo que pido disculpas si hay algún error en ellos–, sino en la parte técnica. Veremos los formatos de los ficheros y cómo resolver su creación en SQL Server. Al ser un caso general, servirá de base para cualquier otro donde se usen ficheros XML.

Comenzamos esta serie de artículos por la definición del fichero de intercambio: Los esquemas XML.

Partes de un esquema

Una de las cosas más importantes de un XSD es que es un estándar –recomendación– del W3C. Por lo tanto, es independiente de cualquier vendedor o desarrollador de software. Las razones de su implantación cada vez mayor, además de lo expuesto, son:

- Su extensibilidad, un esquema puede extender otro, como veremos más adelante, donde se reusa el esquema de envío en el de respuesta. También veremos cómo se crean tipos específicos de la comunicación.
- Están escritos en un estándar como es XML. Por lo tanto no usan un nuevo lenguaje, pueden verse en cualquier editor o analizador («parser») XML, se pueden manipular mediante el estándar XML DOM que soportan la mayor parte de los navegadores y compiladores y se puede formatear mediante hojas de estilo XSLT.
- Su sintaxis es rica y poderosa. Van más allá de un documento «bien formado» al definir –concretar– los elementos y atributos, lo que evita malos entendidos.
- Usan tipos de datos. Esto es muy importante porque se define el contenido, facilita el trabajo con bases de datos, restringe los datos, permite patrones de datos y facilita la conversión entre tipos de datos.
- Usan espacios de nombres.

Pero ¿qué es lo que definen?, así podremos ir viendo sus partes basándonos en el ejemplo que proponemos:

- Qué elementos pueden aparecer en un documento. Lo que es «legal», como hemos dicho antes.
- Lo mismo para los atributos.
- Qué elementos son hijos, el orden de los hijos y el número de veces que puede aparecer un hijo.
- Si un elemento es vacío o puede incluir texto.
- Los tipos de datos para elementos y atributos.
- Valores por defecto y fijos para elementos y atributos.

Pero pasemos a la acción, comencemos con el análisis.

XMLSpy

Aunque XML fue diseñado para que cualquier procesador de texto lo editara, existen editores especializados que son superiores y más poderosos. El que puede ser el mejor editor de XML del mercado es Altova XMLSpy, el estándar industrial en desarrollo y modelado XML. XMLSpy hace muy fácil crear cualquier tipo de documentos XML incluidos esquemas, hojas de estilo (XSL), consultas XQuery, etc. De hecho, los esquemas, la documentación y los ejemplos que estamos analizando, usan ese software. XMLSpy tiene muchas características dirigidas a los desarrolladores como el formateo de impresión, el autocompletado y elección de etiquetas, uso y aprovechamiento de esquemas, que hacen muy fácil la visualización, edición y creación de documentos XML. Es un Entorno de Desarrollo Integrado (IDE) para el diseño, prueba y manipulación de todo tipo de XML. No sólo los desarrolladores se benefician de su uso, sino cualquiera que tenga que manejar documentos XML. Lo vamos a comprobar a continuación.

El elemento «schema»

El elemento esquema es la raíz de cualquier esquema XML. Puede contener atributos:

```
1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <xsd:annotation>
```

En este caso el atributo `xmlns:xsd="http://www.w3.org/2001/XMLSchema"` nos indica que los elementos y tipos de datos usados en el esquema vienen del espacio de nombres `http://www.w3.org/2001/XMLSchema`, que es el estándar para esquemas. También nos dice que los elementos y tipos de datos que vengan de dicho espacio de nombres, deben usar el prefijo «xsd». Este prefijo no es estándar, se puede usar el que se quiera y, de hecho, en otra literatura se suele usar «xs». Puede tener más atributos, pero en este caso no se usan.

El esquema puede tener varios tipos de elementos. Como veremos, uno de ellos se llama `element`, aunque el primero que nos encontramos es `<xsd:annotation>`: es un elemento de primer nivel que contiene comentarios y que sirve como una documentación en línea. En el presente caso sólo tiene un elemento hijo, `<xsd:documentation>`, que contiene un atributo indicando el idioma del texto que tiene. Nos dice qué define el esquema.

Pero lo más importante es el único elemento `<xsd:element name="Certificado_empresa">` porque va a ser el elemento raíz del documento XML que use este esquema.

Esquemas

En los esquemas hay elementos simples y complejos. Un elemento es simple cuando sólo tiene texto. Si tiene atributos u otros elementos, ya es complejo. Por lo tanto, por ahora no hemos visto ninguno simple, pues incluso el documentación tiene un atributo. Veamos, pues el primer elemento, que es complejo porque tiene otros elementos dentro. Para ello usaremos las vistas de XMLSpy que nos van a ayudar mucho. Dentro de la vista esquema, podemos ver:

comment	NIVEL 0	
element	Certificado_empresa	ann:
comment	NIVEL 1	
complexType	CERTIFICADO_EMPRESA_TYPE	ann:
comment	NIVEL 1.1	
complexType	CUENTA_COTIZACION_TYPE	ann:
comment	NIVEL 2	

Debajo del comentario XML que nos indica quién es el elemento raíz, vemos que está **Certificado_empresa** que tiene un botón a su izquierda. Mediante este botón saltamos a la vista gráfica que es uno de los aportes más importantes de XMLSpy:

Inmediatamente vemos que el elemento **Certificado_empresa** es del tipo **CERTIFICADO_EMPRESA_TYPE**, es un tipo complejo, que tiene una o varias **Cuenta_cotizacion**. Debajo de **Cuenta_cotizacion** vemos «1..∞». Si miramos el texto que define este elemento:

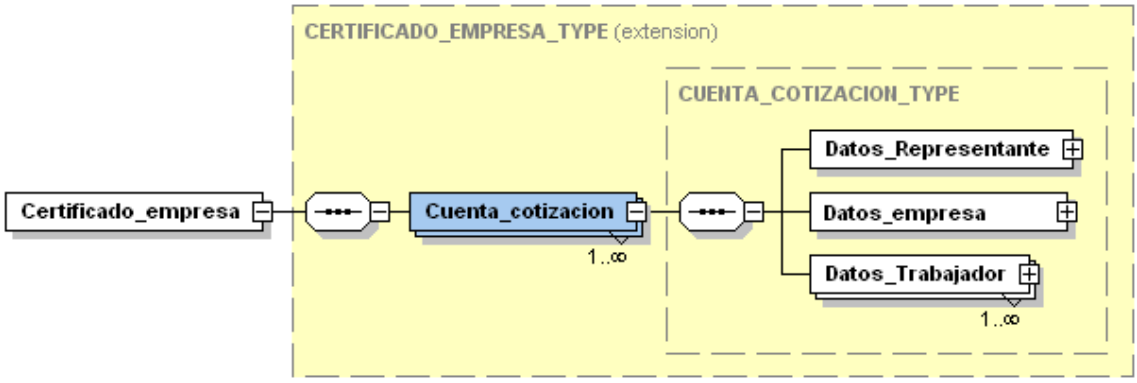
```

9      <!-- NIVEL 0 -->
10     <xsd:element name="Certificado_empresa">
11       <xsd:complexType>
12         <xsd:complexContent>
13           <xsd:extension base="CERTIFICADO_EMPRESA_TYPE"/>
14         </xsd:complexContent>
15       </xsd:complexType>
16     </xsd:element>

```

Vemos que es mucho más complejo que la forma gráfica más los detalles que nos ofrece XMLSpy. Sobre todo si el texto del esquema no está ordenado –que no es el caso, este esquema está creado con la que se considera mejor herramienta profesional–, lo que hace bastante difícil seguir la jerarquía de elementos del esquema. Podemos comparar la vista gráfica con muchas otras y rápido veremos la cantidad de tiempo que nos ahorra y lo fácil que nos hace entender el esquema.

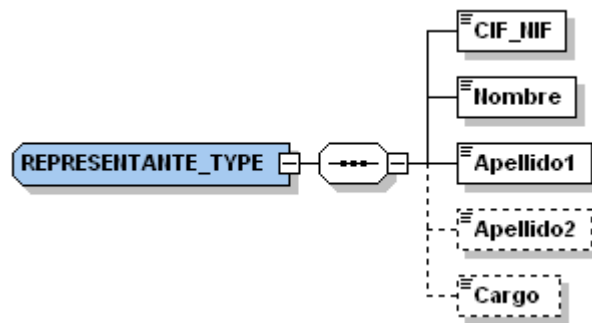
Además es muy fácil ver los componentes de **Cuenta_cotizacion**: Basta hacer clic en el signo más que está a su derecha:



Vemos que un [Certificado_empresa](#) es único, que tiene una o más [Cuenta_cotizacion](#), del tipo [CUENTA_COTIZACION_TYPE](#), que está compuesto cada tipo por tres elementos complejos: [Datos_Representante](#), [Datos_Empresa](#) y [Datos_Trabajador](#). Los dos primeros sólo pueden aparecer una vez y el último una o más. Es decir, inmediatamente vemos que se necesitan los datos de un representante, de una empresa y que se pueden enviar uno o más trabajadores por cuenta cotización.

Elementos complejos

Cada uno de estos elementos se puede analizar por separado. Si volvemos a la vista de esquema y hacemos clic en el botón a la izquierda de [Datos_Representante](#), nos muestra:



Inmediatamente vemos los datos que necesitamos del representante, pero, si nos fijamos, hay dos elementos cuyos bordes están entrecortados: son opcionales. Por lo tanto, también podemos apreciar de forma gráfica que, si no tenemos el segundo apellido o el cargo, no es necesario ponerlo.

Si queremos ver los tipos complejos que forman este esquema, claro que tenemos una ventana para ello:

Captura de pantalla de una herramienta de desarrollo de esquemas XML. El panel principal muestra el diagrama de **REPRESEITANTE_TYPE** con sus subelementos: **CIF_HIF**, **Hombre**, **Apellido1**, **Apellido2** (con borde entrecortado) y **Cargo** (con borde entrecortado). A la derecha, el panel **Components** muestra una lista de elementos y tipos complejos, incluyendo **Certificado_empresa**, **CERTIFICADO_EMPRESA_TYPE**, **COTIZACION_REA_TYPE**, **COTIZACION_TYPE**, **CUENTA_COTIZACION_TYPE**, **DISTRIBUCION_JORNADAS_TYP**, **EMPRESA_TYPE** y **PERIODO DISTRIBUCION JORNADA**. El panel **Details** muestra los detalles del elemento seleccionado, **REPRESEITANTE_TYPE**, con campos para **name**, **base**, **mixed** y **abstract**. El panel **Facets** está vacío. En la parte inferior, se ven las pestañas **Text**, **Grid**, **Schema**, **WSDL** y **XBRL**.

En la ventana componentes, pestaña globales, vemos que hay un elemento, varios tipos complejos y, aunque no están visibles, más abajo los tipos simples. Si estamos analizando un esquema, esta ventana nos conduce rápidamente por los distintos componentes, pero la idea de los tipos es que se puedan usar en varios sitios, como se ve también en este esquema, por lo que una vez definidos, esta ventana ayuda mucho en la creación de elementos al no tener más que usar su definición.

Conclusión

En esta primera parte del artículo hemos comenzado el análisis de un caso real. Hemos visto la importancia que tienen los esquemas dentro de los documentos XML, qué es un esquema y comenzado el estudio mediante una de las herramientas líderes para trabajar con todo el universo XML, que nos aísla completamente de la complejidad del esquema en estudio. Nos referimos a XMLSpy.

Para más información.

Danysoft, es el representante oficial de XMLSpy en la península ibérica, ofreciéndole tanto las licencias de XMLSpy 2011 en las mejores condiciones, como los servicios de formación y consultoría necesarios para su correcto uso. Puede contactar con Danysoft en el 902 123146, o ver más información en www.danysoft.com