



**<?xml?>**



# INTRODUCCIÓN XML

- XML es un lenguaje de marcas extensibles y tiene su origen en la gestión de documentos, proviene del lenguaje SQML creado por IBM.



# PARTES

- **Estructura**
  - División del XML
- **Schema**
  - Lenguaje XML Schema
- **Consulta y transformación**
  - Herramientas para consultar y transformar los datos XML
- **APIs**
  - Interfaz de aplicación
- **Almacenamiento de datos XML**
  - Bases de datos no relacionales y relacionales
- **Aplicaciones**
  - Almacenamiento y comunicación de datos para el acceso a servicios Web



# ESTRUCTURA

## XML SE DIVIDE EN PARTES: ELEMENTOS.

- **ELEMENTOS:** son señalados mediante etiquetas (pueden tener un atributo).

Ejemplo: `<nombre>Alfredo Reino</nombre>`

- Estructura del XML

XML → Prólogo

→ Etiquetas

→ (opcional) CDATA (secciones caracteres especiales)

**Diferencia HTML:** XML contiene datos que se autodefinen.

**Dos tipos de documentos:** los válidos y los bien formados.

- **Bien formados:** cumple todas las definiciones básicas de formato.
- **Válidos:** documentos correctos sintácticamente y además la relación entre elementos se especifica.



# SCHEMA

**XML Schema:** especifica la estructura de los documentos XML.

**DTD** (Document Type Definition), tiene una función similar al XML Schema:

- **Funcionalidad:** Restringir el tipo de información en el documento.
  - **Problemas:** datos muy generales, propia sintaxis especial, elementos con un orden especificado.

Tras esto, se desarrolló **XML Schema**.

- **XML Schema:** Permite flexibilidad en los tipos, extensión de tipos complejos mediante el uso de una forma de herencia, restricciones de unicidad y de clave externa.
- **Desventaja:** más complicado que las DTDs.
- **Formas de codificarlo:** XML Schema Simple, XML Schema Dividido, XML Schema usando nombrado de Tipos.



# CONSULTA Y TRANSFORMACIÓN

Sirven para extraer información de los documentos XML.

**Varios lenguajes de consulta y transformación:** *XPath*, *XSLT*, *Xquery*.

- **XPath:** Trata partes de un documento XML mediante expresiones de rutas de acceso.
- **XSLT:** mecanismo de transformación del lenguaje XSL, transformar un documento XML en otro documento XML o a otros formatos como HTML.
- **Xquery:** diseñado para consulta de colecciones de datos XML. Es semánticamente similar a SQL e incluye varias capacidades de programación.



# APIs

**APIs:** conocidas como interfaces, permiten comunicar dos sistemas que no están en el mismo lenguaje.

**En XML hay 2:** DOM (Document Object Model), SAX (Simple API for XML).

- **DOM:** trata el contenido del XML como un árbol, los programas que acceden al documento a través de la API DOM pueden insertar arbitrariamente, borrar y reordenar nodos, es para el mundo web.
- **SAX:** es un modelo de eventos diseñados para proporcionar una interface común entre analizadores y aplicaciones. Es fácil y intuitiva, muy empleada por los programadores de Java y Python.



# ALMACENAMIENTO DE DATOS XML

## Diferentes alternativas:

- **Bases de datos relacionales:** varias formas; **Almacenamiento como cadenas** (se almacena cada elemento hijo del elemento de mayor nivel como una cadena en una tupla separada de la BD), **Representación arbórea** (los datos se almacenan mediante el uso de un par de relaciones nodos, hijo), **Asignación a relaciones** (los elementos XML conocidos se asignan a relaciones y atributos, los desconocidos son un enfoque híbrido).
- **Almacenamiento de datos no relacionales:** varias formas; **Almacenamiento en archivos planos** (ya que XML es en su mayoría un formato de archivo, un mecanismo natural de almacenamiento es simplemente un archivo plano), **Almacenamiento en una base de datos XML** ( ventaja de que al añadir el lenguaje XML en una BD proporciona consultas declarativas).





# APLICACIONES

**Aplicaciones de XML:** se utilizan para el almacenamiento y comunicación de datos para el acceso a servicios Web.

- Muchas aplicaciones necesitan almacenar datos estructurados complejos que no se modelan fácilmente. **Solución:** usar XML.

## Intercambio de datos

- **Normas basadas en XML:** sirven para representar datos de aplicaciones especializadas. Van desde aplicaciones de negocio como banca (RosettaNet), transportes (necesitan información sobre los bienes transportados), a aplicaciones científicas; química (ChemML norma para representar la industria química) y biología molecular.



## Servicios Web

- **Conectarse a una base de datos externa:** muchas aplicaciones y webs no nos permiten conectarnos a una base de datos externa, usando SQL. **Solución;** sacar los resultados en XML.



## Mediación de datos

- **Gestión Centralizada:** soluciona el problema extrayendo una representación XML desde los diferentes sitios.