TEMA 2

REPRESENTACIÓN DE LA INFORMACIÓN

Índice

Bibliografía

- The XML Companion (third edition).
  Neil Bradley. Addison-Wesley

- XML Al descubierto.
  Michel Morrison. Prentice-Hall.

- XML a través de ejemplos.
  A. Gutiérrez, R. Martínez. Ra-Ma

- Beginning XML, 5th Edition. Disponible en O'Reilly. Capítulos 1, 2 y 3. El material de trabajo del que habla se descarga aquí.
- Lenguajes de Marcas y Sistemas de Gestión de Información. Acceso libre en Internet. Capitulo 1.
- Lenguaje de marcas y sistemas de gestión de la información. Disponible en eLibro. Capítulos 1 y 3.

- eLibro y O'Reilly se puede acceder desde la VPN de la UPM o abrirse una cuenta con @alumnos.upm.es y acceder desde cualquier lugar lugar.

Algunos enlaces interesantes

- W3Schools [www.w3schools.com]
- XML.com de O'Reilly [www.xml.com]
- Portal XML para la Industria [www.xml.org]
- ebXML [www.ebxml.org]
- OASIS [www.oasis-open.org]
- TEI [www.tei-c.org]
- Dublin Core [dublincore.org]
- Creando Documentos Electrónicos
  [ota.ahds.ac.uk/documents/creating/]
- Libro blanco Java-XML [java.sun.com/xml/ncfocus.html]

INTRODUCCIÓN

Introducción

- El modelo Internet no posee nivel de presentación.
- La idea fundamental es:

MANEJAR LOS DOCUMENTOS
DE FORMA ORGANIZADA
PARA FACILITAR EL INTERCAMBIO Y
LA MANIPULACIÓN DE SUS DATOS.

Introducción

- La primera tecnología fue el SGML (Standard Generalized Markup Language), desarrollado por IBM. Normalizado en ISO-8879 en 1.986.
- En 1989 se define HTML, lenguaje para navegadores Web.
- En 1998 el W3C lanza XML (eXtensible Markup Language), que es un **metalenguaje** (lenguaje para la creación de lenguajes).
- A partir de este momento, vamos a considerar el objeto "documento" como la unidad completa mínima de intercambio de información entre los procesos de aplicación.

eXtensible Markup Language

(XML)

Introducción

- XML es un subconjunto de SGML.
- Principales características:
  - Extensibilidad.
  - Estructura: bien formado
  - Validación: válido (conforme a un modelo especificado).
- Separa el contenido (los datos) de su presentación (cómo se ven esos datos).

Introducción

- XML no es una solución de nada por sí mismo, sino un marco que permite crear soluciones.
- Permite generar un conjunto personalizado de etiquetas que sirven para codificar tipos específicos de información.
- De un documento XML se puede ver su estructura, pero no el significado de su contenido si no se le ha formateado con **hojas de estilo.**

XML. ¿Dónde está?

- Es un lenguaje para el marcado de documentos.
- Última versión disponible en la página del World Wide Web Consortium:

  http://www.w3.org/TR/xml/

- La página raíz de XML en World Wide Web Consortium es:

  http://www.w3c.org/XML

- El conjunto de recomendaciones se encuentra en:

  https://www.w3.org/standards/xml/

XML. ¿Qué es?

- "XML describe una clase de objetos llamados **DOCUMENTOS XML** y parcialmente describe el comportamiento de programas de computador que pueden procesarlos" (recomendación).

- Es un documento compuesto por una secuencia de caracteres. Estos o bien pertenecerán a las marcas o serán contenido.

Los 10 objetivos de XML

1. XML debe ser utilizable directamente sobre Internet.
2. XML debe soportar una amplia variedad de aplicaciones.
3. XML debe ser compatible con SGML.
4. Debe ser fácil escribir programas que procesen documentos XML.
5. El número de características opcionales en XML debe ser mantenido en un mínimo, idealmente cero.
6. Los documentos XML deben ser legibles por un humano y razonablemente claros.
7. El diseño de XML debe ser preparado rápidamente.
8. El diseño de XML debe ser formal y conciso.
9. Los documentos XML deben ser fáciles de crear.
10. La brevedad en la marcación es de mínima importancia

XML en 10 puntos (según W3C) (I)

http://www.w3.org/XML/1999/XML-in-10-points.html

http://www.w3.org/XML/1999/XML-in-10-points.es.html

1. XML es para estructurar datos. XML es un conjunto de reglas que permiten estructurar datos. No es un lenguaje de programación.

2. XML se parece un poco al HTML. Ambos usan etiquetas (< >) y atributos (nombre = "valor"). En HTML tienen un significado para el navegador, pero en XML sólo delimitan piezas de datos.

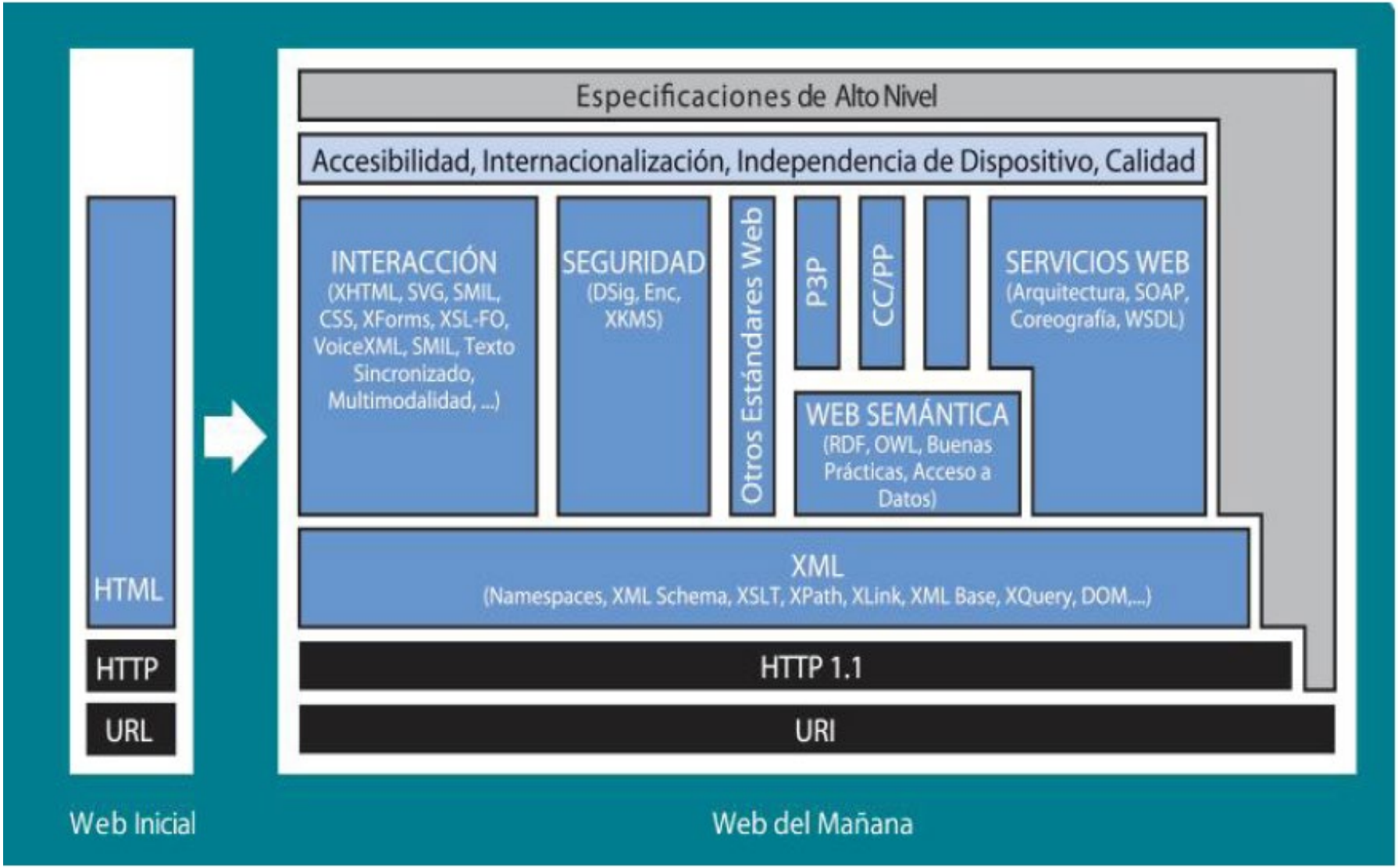
XML en 10 puntos (según W3C) (II)

3. XML es texto, pero no está pensado sólo para ser leído.
4. XML es verboso por diseño. Los archivos XML son casi siempre más grandes que los formatos binarios comparables.
5. XML es una familia de tecnologías.
6. XML es nuevo, pero no tanto. Antes de XML estuvo SGML, desarrollado a principios de los 80, estándar ISO desde 1986.

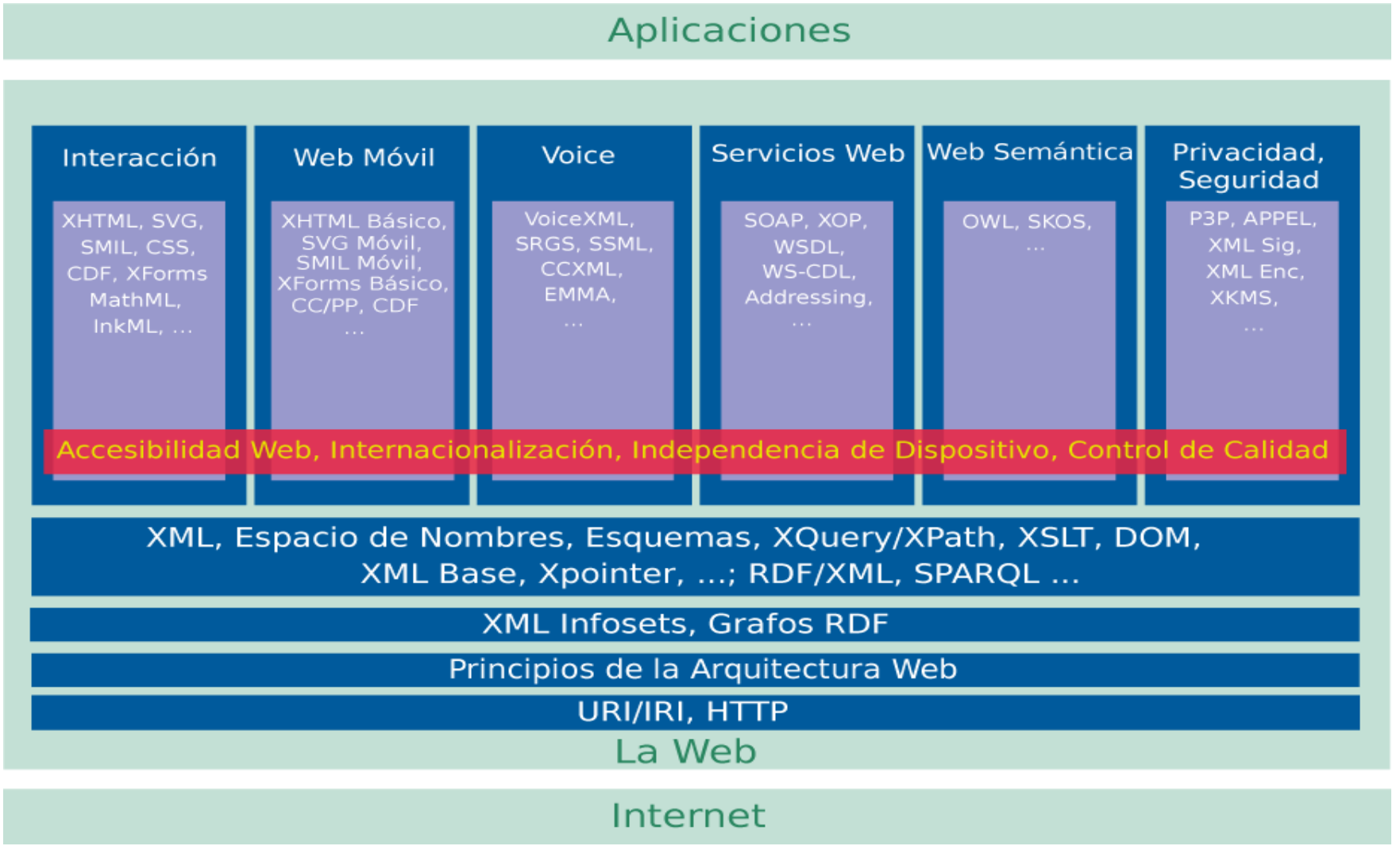
XML en 10 puntos (según W3C) (III)

7. XML lleva HTML a XHTML. La sintaxis de HTML varió ligeramente para adaptarse a las reglas del XML.
8. XML es modular. Un documento XML se puede hacer combinando otros con otros formatos. Para evitar duplicidad en las marcas se definen los espacios de nombres. Para ello se utilizan DTD o XMLSchemas.
9. XML es la base de RDF y Web semántica. Resource Description Framework soporta aplicaciones de descripción de recursos y metadatos. Integra las aplicaciones y los agentes en una Web Semántica.
10. XML es gratuito, independiente de la plataforma y bien soportado.

“La web del mañana”

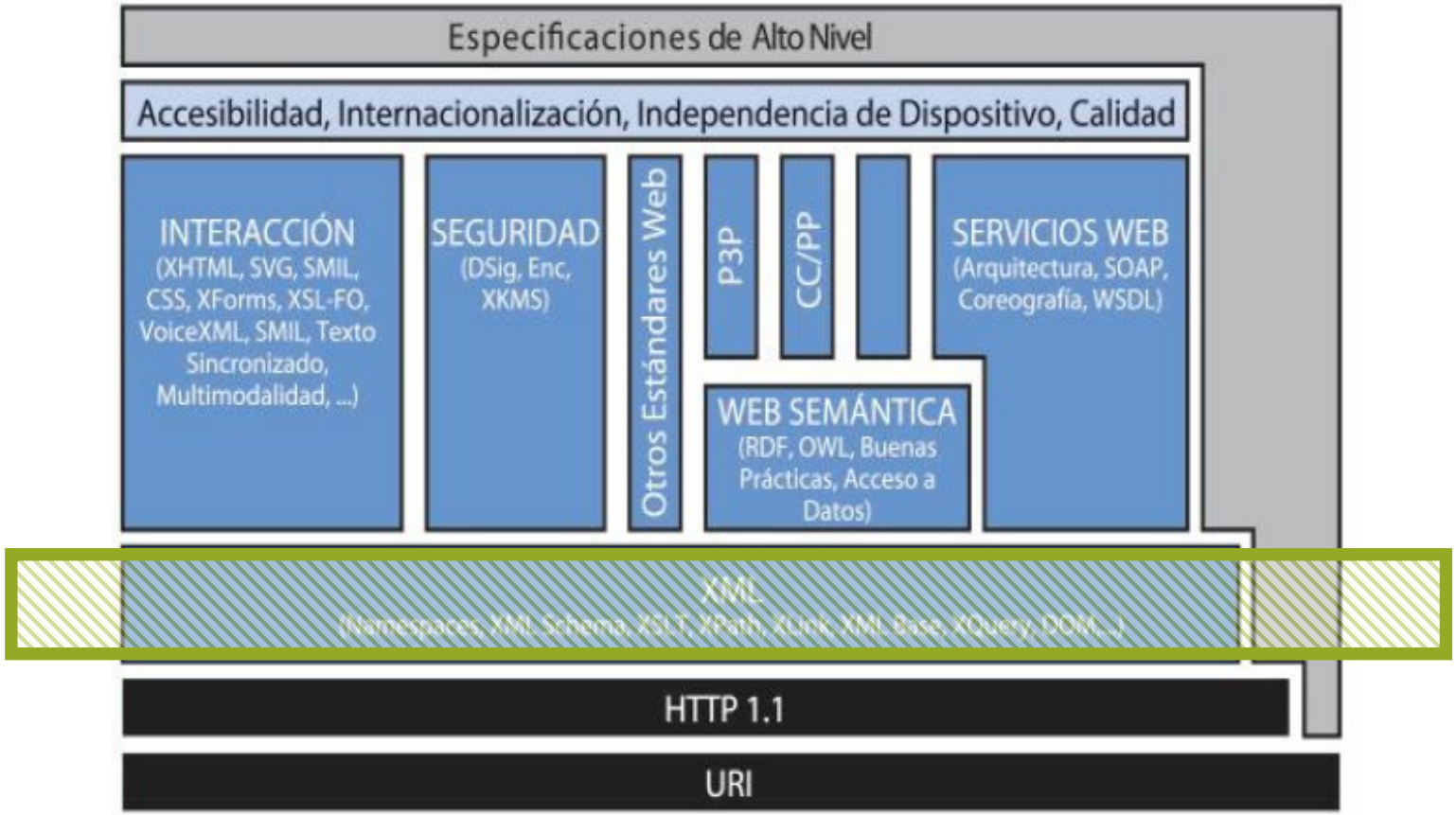


“La web del mañana”



Familia de Tecnologías XML

- Conjunto de recomendaciones que permiten el uso de XML en la web:



Familia de Tecnologías XML

- Namespaces: contexto en el que cada etiqueta tiene significado y no se pueda confundir con otra igual, pero de diferente espacio de nombres.
- XML Schemas: para definir la estructura y la semántica de los documentos XML.
- Recorrido de un documento XML
  - XPath: para especificar la localización de los diferentes componentes que forman un documento XML.
  - XPointer: para localizar los fragmentos que forman cada uno de los componentes.

Familia de Tecnologías XML

- Transformaciones:
  - XSLT: documento XML que establece un conjunto de reglas para transformar un documento XML en otro, de acuerdo a unas hojas de estilo
- Navegación por el documento XML:
  - DOM: conjunto de recomendaciones de W3C que define, entre otras cosas, una interfaz estándar que permite a los diferentes lenguajes de programación el uso y modificación de los documentos XML.

Familia de Tecnologías XML

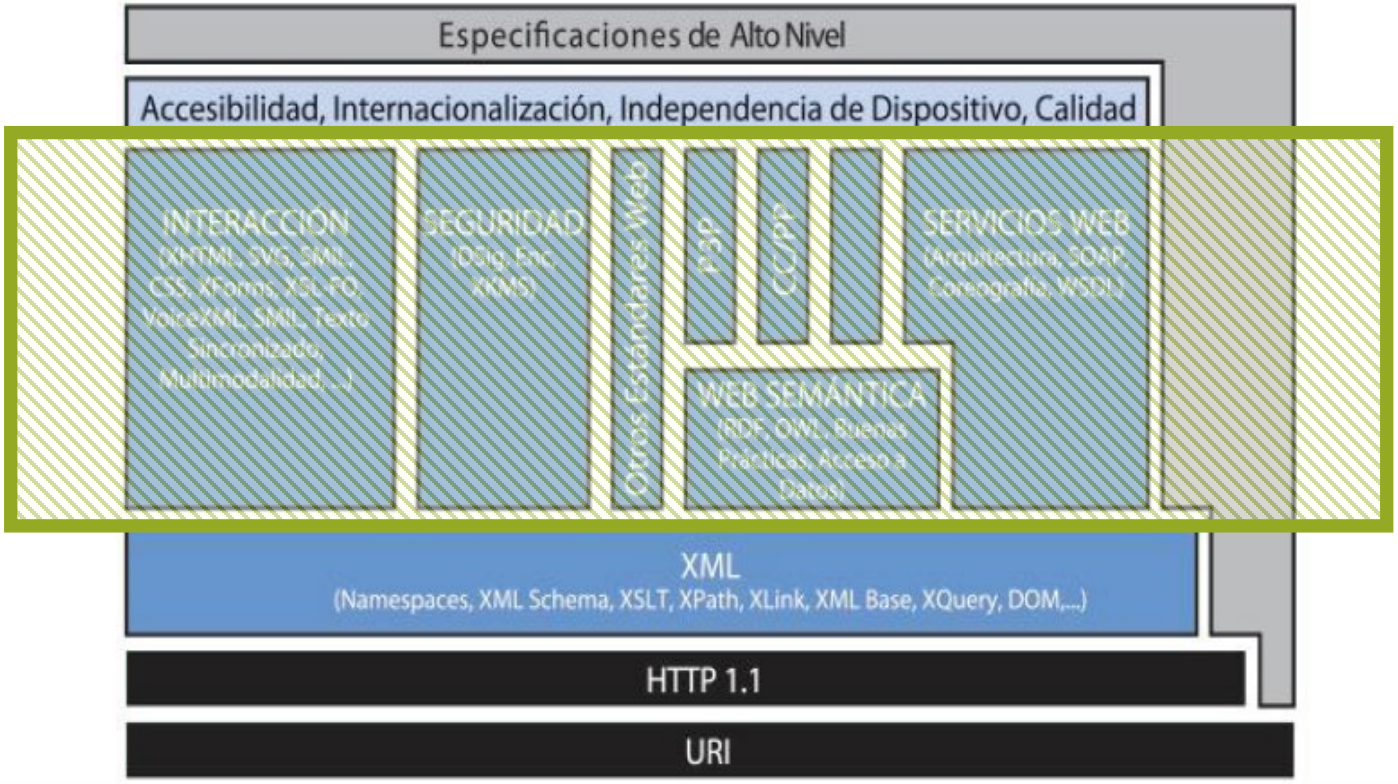
- Vínculos entre documentos:
  - XLink y XBase: para realizar enlaces bidireccionales (entre dos documentos XML) o monodirecionales (de uno a otro o a múltiples documentos XML).
  - XInclude: para incluir un documento en otro. En un documento se pueden incluir tantos otros documentos XML como sea necesario.
- Consultas de datos:
  - XQuery: para localizar datos en un documento XML.

Aplicación XML

- Es una clase de documento XML que define un conjunto de reglas adicionales a la familia XML (reglas de marcación relevantes para una aplicación).
- Es decir, que definen su propio vocabulario XML (o de cualquiera de las tecnologías de la familia XML).
- No hay que confundirlas con las aplicaciones tradicionales (p.e. procesador XML, editor XML…).

Aplicaciones XML

- **Aplicaciones horizontales** (o vocabularios para aplicaciones horizontales): proporcionan mecanismos para resolver tareas genéricas, tales como interacción, Web Services o seguridad. Las establece el W3C.



Aplicaciones XML

- **Aplicaciones de dominio de conocimiento** (o vocabularios para aplicaciones en dominios de conocimiento): cada sector de actividad define sus propios vocabularios XML para su campo de actividad o conocimiento.

- Son definidas por varias organizaciones. La más significativa es OASIS (Organization for Advancement of Structured Information Standars).

Aplicaciones XML horizontales. Interacción.

- **Interacción y presentación de documentos**: los lenguajes (de los muchos existentes) más usados son:
  - XHTML: es HTML reescrito para cumplir las reglas de estructura de un documento XML.
  - XSL-FO (eXtensible Style Language - Formatting Objects): describe como va a ser presentado un documento por pantalla, papel … Necesita una transformación previa a XSLT.
  - InkML (Reconocimiento de tinta digital): formato de datos XML que representa la entrada de datos introducida a través de tinta digital en escritura manual.

Aplicaciones XML horizontales. Interacción.

- SVG (Scalable Vector Graphics): lenguaje XML para la definición de los vectores (coordenadas geométricas) que forman un gráfico. Se usa para gráficos hechos por el usuario, no para fotos, ya que el formato vectorial no es el más apropiado.
- SMIL (Synchronized Multimedia Integration Language): Lenguaje XML que permite realizar presentaciones multimedia con sonidos y gráficos animados.
- VoiceXML: interacción mediante diálogos de voz (por vía telefónica, p.e.) con documentos XML.

Aplicaciones XML horizontales. Interacción.

- CCXML (Call Control XML): complementario con VoiceXML, que permite funciones avanzadas de telefonía, como conferencias multiparte, y de diálogo. Su uso conjunto con VoiceXML permitirá mejorar sustancialmente la interacción voz-XML.
- SSML (Speech Synthesis Markup Language): lenguaje XML para permitir sintetizar la voz en distintas aplicaciones (especialmente en los principales idiomas).
- XV (Xhtml – Voicexml) o interacción multimodal: que permitira hacer una petición mediante voz (VoiceXML, CCXML y SSML) y recibir una respuesta visual (XHTML). Su desarrollo está unido al de los dispositivos móviles.

Aplicaciones XML horizontales. Seguridad.

- **Seguridad en los documentos XML:**
  - Cifrado XML (XML Encription): lenguaje XML que permite la utilización de técnicas criptográficas para el encriptado desencriptado de la información.
  - Firma XML (XML Signature): lenguaje XML para la firma digital, lo que permite que se identifique fácilmente al generador o controlador de esa información.
  - Actualmente se está en el desarrollo de la especificación en XML de políticas de seguridad, básicamente para Web Services (WS-Policy , WS-Trust, WS-Federation, …).

- **Web semántica y RDF:**

  RDF (Resource Description Framework) permite proporcionar información sobre la propia Web, describiendo recursos de manera consistente. El objetivo la clasificación, localización y catalogación automática de los recursos de la Web.

  Se basa en el uso de tripletas:

  Sujeto, predicado, objeto

  identificando todos los recursos mediante una uri.

Aplicaciones XML horizontales. Web semántica.

- Basado en RDF se trata de buscar una semántica que defina los recursos de la web definiendo relaciones (de rango, de restricción ...) entre ellos.
- Se trata que haya un procesado inteligente de los recursos.

Ej.: Vuelos

semántico, ca.(Del gr. σημαντικός, significativo).
1. adj. Perteneciente o relativo a la significación de las palabras.
2. f. Estudio del significado de los signos lingüísticos y de sus combinaciones, desde un punto de vista sincrónico o diacrónico.

Aplicaciones XML horizontales. CC/PP.

- **Capacidades de dispositivo y preferencias de usuario:**

  Debido a la gran cantidad de dispositivos capaces de utilizar el contenido de la web, CC/PP (Composite Capabilities / Preference Profile) es un lenguaje XML que permite comunicar al servidor las características de propio dispositivo, adaptando el resultado de la petición a éstas características.

Aplicaciones XML horizontales. P3P.

- **Platform for Privacy Preferences (P3P):**

  Es lenguaje XML que ofrece a los usuarios una forma sencilla y automatizada de controlar en mayor medida el uso que se hace de su información personal en los sitios Web que visitan.

  Proporciona a los usuarios la posibilidad de decidir si quieren o no, y bajo qué circunstancias, revelar información personal.

Aplicaciones XML horizontales. Web Services.

- **Servicios Web (interacción con aplicaciones distribuidas):**

  Consisten en servicios que a partir de unos parámetros de entrada genere una salida, mediante la colaboración de diferentes aplicaciones.

  El formato de intercambio de documentos es XML (basado en SOAP (Simple Object Access Protocol)), lo que implica que la aplicación se tiene que registrar en el Servicio de Directorio.

Aplicaciones XML horizontales. Web Services.

- Los principales componentes de un servicio Web son:
  - Transferencia de datos en documentos XML usando SOAP.
  - La descripción de los interfaces (WSDL (Web Services Definition Languaje)).
  - La presentación de formularios adaptados a cada aplicación (Xforms).
  - Componentes complementarios: seguridad, control de transacciones, acceso a bases de datos …

Aplicaciones XML de dominio de conocimiento

- Con las tecnologías presentadas, no sólo cambia la forma de gestión de la información en la Web, sino que también cambia la forma de gestión de la información en las empresas, instituciones, sectores profesionales o sectores del conocimiento.

- Utilizan XMLSchemas o DTD para definir estructuras de vocabularios que son utilizados en los sectores indicados (ver ontologías).

Aplicaciones XML de dominio de conocimiento

- Se generan dos tipos de lenguajes:
  - Comunes:
    - Computer Enviroment XML.
    - Customes Information.
    - EDI XML.
    - Human XML
    - Math XML
    - Open Office XML
    - Universal Business Language (UBL).
    - Universal Data Element Framework (UDEF)
    - …

Aplicaciones XML de dominio de conocimiento

- Específicos:
  - Astronomy XML.
  - Chemistry XML.
  - Education XML.
  - Finance XML:
    - eXtensible Bussines Reporting Language (XBRL).
    - Market Data Definition Language (MDDL).
    - Open Financial Exchange (OFX) XMLSchema.
    - …
  - Healhcare XML.
  - Human Resources.
  - Insurance XML.

Aplicaciones XML de dominio de conocimiento

- Legal XML:
  - Legal XML eContracts.
  - Legal XML eNotary.
  - Legal XML Legislative Documents.
  - …
- Manufacturing XML.
- Photo XML.
- Physics XML.
- Publishing XML.
- Telecommunication XML.
- Travel XML.
- …

XML

eXtensible Markup Language

Documentos XML

- Es un tipo de ***objeto de datos*** que están formados por unidades de almacenamiento llamadas ***entidades***, cuyo contenido pueden ser ***datos analizados*** o ***datos no analizados.***
- Datos analizados (PCDATA (Parsed Character DATA) compuestos por:
  - ***Datos de caracteres.***
  - ***Marcas.***
- Datos no analizados (CDATA (Character Data)).

Marcas

- Son símbolos especiales que indican que el texto que les sigue ha de ser procesado.
- Permiten establecer la estructura lógica y de almacenamiento de un documento XML.
- Las marcas van entre "<" y ">":

```
<esto_es_una_marca_XML>
```

Documento XML bien formados

- Su estructura en forma de árbol con elementos distribuidos jerárquicamente.
- Existe siempre un único elemento **raíz**.
- El elemento raíz representa el elemento documento.
- Normas básicas para que un documento XML sea considerado **bien formado:**
  - Un documento XML sólo puede tener un elemento raíz (elemento documento).
  - Todos los elementos con contenido han de tener marcas de apertura y de cierre:

    ```
    <marca> informacion </marca>
    ```

  - Las marcas de los distintos elementos no se pueden solapar.

Documento XML bien formado

- Las marcas anidadas se deben cerrar en orden inverso al de la apertura:

<marca1> <marca2>…….</marca2> </marca1>

- Los atributos de los documentos XML deben ir siempre entre comillas dobles (") o simples (').

- No se pueden usar los caracteres “<”, “>” y “&” en el contenido de un elemento (se sustituyen por < > y &).

Documento XML bien formado

```
<?xml version="1.0"?>
<libro>
     <capitulo>
          <titulo_cap> tema 1 </titulo_cap>
     </capitulo>
     <capitulo>
          <titulo_cap> tema 2 </titulo_cap>
          <seccion>
            <titulo_sec> parte 2.1 </titulo_sec>
          </seccion>
     </capitulo>
</libro>
```

Documento XML bien formado

- Bien formado

```
<seccion>
 <titulo_sec>
     parte 2.1
  </titulo_sec>
</seccion>
```

- Mal formado por solapamiento de marcas.

```
<seccion>
 <titulo_sec>
     parte 2.1
 </seccion>
</titulo_sec>
```

Documentos XML bien formados

- Si los caracteres de marcado se han de utilizar se hace mediante el uso de entidades:

&	“&”
'	“'”
"	“"”
<	“<”
>	“>”

- **XML es sensible a mayúsculas y minúsculas.**

Documento XML bien formado

- Los caracteres que se pueden incluir en un documento XML son:
  - Los del código ASCII (incluidos tabulador, retorno de carro).
  - Los caracteres gráficos de ISO/IEC 10646.

- Se incluyen mediante referencias a los caracteres UTF-8 y UTF-16 usando codificación UNICODE

```
&#codigoCaracter;
```

Documentos XML bien formados

```
<TextoConAcentos>
 <item> A con acento: &#193;</item>
 <item> A con acento: &#xc1;</item>
 <item> E con acento: &#201;</item>
 <item> I con acento: &#205;</item>
 <item> O con acento: &#211;</item>
 <item> U con acento: &#218;</item>
</TextoConAcentos>
```

La codificación UNICODE se puede encontrar en

https://en.wikipedia.org/wiki/List_of_Unicode_characters

Estructura de un documento XML

- Consta de dos partes:
  - Prólogo:
    - La declaración XML.
    - Instrucciones de procesamiento.
    - La declaración de Tipo de Documento (DOCTYPE).
    - Comentarios
  - Contenido: el elemento raíz (entidad documento) y el resto de elementos y su contenido:
    - Elementos
    - Referencias a entidades.
    - Secciones CDATA
    - Comentarios.

Estructura. Prólogo. Declaración XML

```
<?xml version="1.0" [encoding = "ISO-8859-1"]
  [standalone="yes"]?>
```

- Es la primera instrucción de un documento XML:
  - Marca el documento como texto XML.
  - Declara cual es la versión de XML usada en el documento.
  - Indica la codificación empleada para los caracteres.
  - Indica si el documento es autónomo o no.

- Sirven para pasar información que no ha de ser analizada por el procesador XML.
- Es opcional.
- Comienzan identificando la aplicación destino que ha de procesar dicha instrucción:

```
<?xml-stylesheet type="text/xsl"
   href="transfor.xsl"?>
```

  que serviría para el procesamiento con hojas de estilo xsl.

- Pueden estar también situadas en cualquier punto del documento.

Estructura. Prólogo. Declaración DOCTYPE

- Permite definir una serie de restricciones que ha de cumplir un documento XML:
  - Indica la entidad documento (elemento raíz).
  - Indica donde están las restricciones .

```
<!DOCTYPE libro SYSTEM "libro.dtd">
```

Se emplea SYSTEM "archivo" si el archivo está en la misma máquina o PUBLIC "ID" "URI" si es una referencia a un documento externo.

```
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
    "http://www.wapforum.org/DTD/wml13.dtd">
```

Estructura. Prólogo. Comentarios

- Son datos no analizados por el procesador XML.
- Sirven para documentar el contenido del documento XML.
- Se sitúan entre <!-- y -->:

```
<!-- Esto es un comentario -->
```

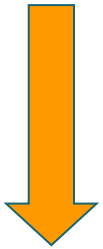
- Los comentarios pueden aparecer también en cualquier parte de un documento XML, es decir, en el prólogo o en el contenido.
- La secuencia `--` no puede formar parte de un comentario

Estructura. Contenido.

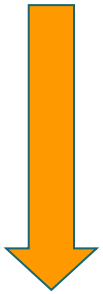
- Contiene información sobre la información.
- Se estructura la información contenida en un documento, utilizando XML.
- Componentes clave:
  - Elementos.
  - Atributos.

Elementos

Unidad lógica básica con capacidad de representar la estructura lógica y semántica de un documento XML.



Unidad básica de representación del contenido de un documento XML.



Cada una de las piezas en las que se puede dividir la información de un documento XML.

Elementos

- Sólo hay un elemento raíz, que no puede formar parte de los demás.
- El nombre de las etiquetas delimitadoras de un elemento debe ser igual. (XML es sensible a mayúsculas y minúsculas).
- Si la etiqueta inicial de un elemento está dentro de otro elemento, la etiqueta final también debe estarlo. El anidamiento debe ser correcto.

Elementos

```
<?xml version="1.0"?>
<libro>
        <capitulo numero="1">
            <titulo_cap>
                tema 1
            </titulo_cap>
        </capitulo>
        <capitulo numero="2">
            <titulo_cap>
                tema 2
            </titulo_cap>
            <seccion>
                <titulo_sec>
                    parte 2.1
                </titulo_sec>
            </seccion>
        </capitulo>
</libro>
```

Elementos

- El elemento *titulo_sec* está dentro del elemento `seccion`:

```
<seccion>
      <titulo_sec> parte 2.1 </titulo_sec>
</seccion>
```

- Mal anidamiento de las marcas delimitadoras de los elementos

```
<seccion> <titulo_sec>
          parte 2.1
</seccion> </titulo_sec>
```

Elementos

- Un elemento sin contenido es:

```
<elementoVacio></elementoVacio>
```

que se puede expresar también:

```
<elementoVacio/>
```

- Los nombres de los elementos pueden estar formados por:
  - Letras (mayúsculas y minúsculas).
  - Números.
  - Guión y guión bajo.
  - Punto y dos puntos

Elementos

- Ningún nombre puede empezar por xml, XML o combinaciones de ellas. Están reservados.
- El carácter inicial debe ser una letra, un guión bajo o dos puntos.
- Un nombre debe estar formado al menos por un carácter.

Elementos

```
<nombresPermitidos>
        <nOmBrE/>
        <Que_nombre_tan_largo/>
        <Que.nombre.tan.largo/>
        <A123-456-789.987_654_321...../>
        <_:nombre/>
        <nombre_xml/>
</nombresPermitidos>
```

Elementos

```
<nombresNoPermitidos>
        <un;nOmBrE/>
        <@#$%(){}+*/>
        <Que nombre tan largo/>
        <123-456-789.987_654_321...../>
        <-_.nombre/>
        <xml_nombre/>
        <Xml-nombre/>
</nombresNoPermitidos>
```

Elementos. Uso.

- Delimitan el contenido de información del documento.
- El anidamiento permite conocer cual es la estructura lógica del documento y la relación entre sus componentes.
- Las relaciones entre los elementos informan sobre el significado del contenido de un elemento en concreto.

Atributos. Reglas de formación.

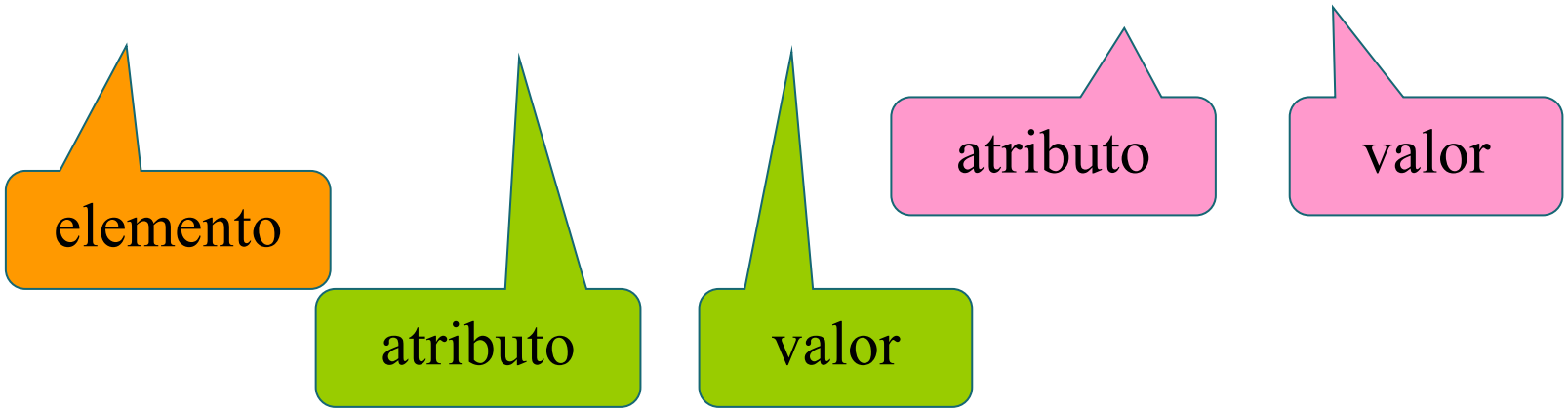
- Actúan como modificadores de los elementos.
- Incluyen información adicional aplicable a un elemento.
- Pueden ir situados en:
  - Una marca de inicio de elemento.
  - En un elemento vacío.
- Esta compuesto por el par:

```
nombreAtributo = "valorAtributo"
```

Atributos

- Un elemento puede contener varios atributos.

```
<articulo codigo="123" precio="25"/>
```



Referencias a entidades

- Las referencias a entidades sirven para utilizar un contenido que ha debido ser definido previamente (en el documento que modela los datos) y que tiene que tener un nombre (nombre de la entidad).

```
<!ENTITY asig "Aplicaciones telematicas">

<pie> Nombre de la asignatura: &asig; </pie>
```

- Existen cinco entidades predefinidas:

"&" "'" """ "<" ">"

Secciones CDATA (character data)

- Estas secciones contienen información que no debe ser procesada.
- Se tratan una secuencia de caracteres, sin estructura determinada.

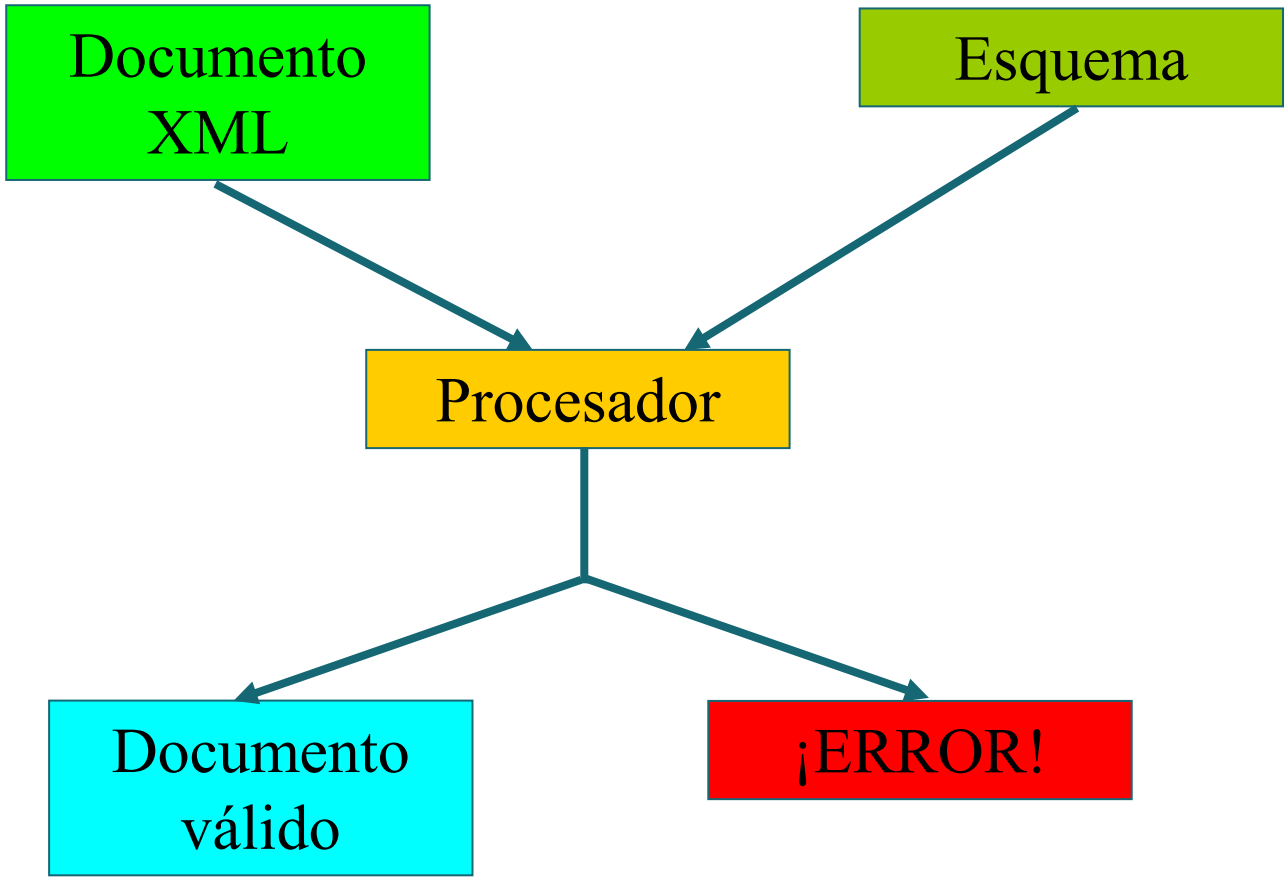
```
<![CDATA[<saludo>Hola, mundo!</saludo>]]>
```

Documentos XML válidos

Introducción

- En un **esquema** se establecen las normas estructurales y sintácticas que ha de cumplir un elemento XML para que sea válido.
- Declara los diversos elementos que pueden ser utilizados en un documento XML y el tipo contenido y atributos de dichos elementos.
- Establece como se han de usar los elementos para crear la estructura de información del documento XML.
- Si un documento XML cumple las reglas establecidas en el esquema se dice que es un **documento válido.**
- Un documento válido siempre es bien formado.
- Un documento bien formado no tiene porque ser válido.

- Esquema de procesamiento:



Introducción

- Para definir un esquema se utiliza un lenguaje de esquema:
  - Definidos por W3C:
    - DTD (Document Type Definition)
    - XMLSchema.
  - No definidos por el W3C:
    - RELAX-NG (REgular LAnguage for XML Next Generation) definido por OASIS.
    - Schematron
    - DSD (Document Structure Definition)

Comparación entre lenguajes de esquema disponible en:

https://en.wikipedia.org/wiki/XML_schema#Languages

TEMA 2

REPRESENTACIÓN DE LA INFORMACIÓN