

Capítulo 1

Comunicación en Sistemas Distribuidos

Al finalizar el capítulo, el alumno podrá:

- Conocer los fundamentos de formatos para intercambio de información.
- Implementar aplicaciones que trabajen con los principales formatos de intercambio de información del mercado.

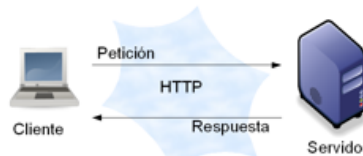
Temas

1. Comunicación basada en protocolo HTTP
2. Normativa XML
3. Normativa JSON
4. XML vs. JSON
5. Herramientas en Java para el procesamiento de XML y JSON

1. Comunicación basada en protocolo HTTP

Comunicación basada en protocolo HTTP

- HTTP (HyperText Transfer Protocol) es un protocolo de nivel de aplicación utilizado para intercambiar la información en la World Wide Web. Sus principales características son:
 - Es un protocolo que no maneja estados, es decir, no almacena información de conexiones previas
 - Esta orientado a las transacciones bajo esquema de petición y respuesta entre un cliente y servidor
 - Utilizado en transferencia de recursos web: páginas estáticas, imágenes, resultados de ejecución de un programa, etc.



1 - 4

Copyright © Todos los Derechos Reservados - Cibertec Perú S.A.C.



1.1. HTTP (HyperText Transfer Protocol)

HTTP (Protocolo de transferencia de hipertexto) es un protocolo de nivel de aplicación utilizado ampliamente para intercambiar información en la [World Wide Web](#). Sus principales características son las siguientes:

- Es un protocolo que no maneja estados, es decir no guarda información de conexiones previas.
- Está orientado a transacciones y sigue el esquema de petición-respuesta entre cliente y servidor.
- Se utiliza para la transferencia de recursos web tales como: páginas estáticas, imágenes, resultados de ejecución de un programa, etc.

HTTP ha pasado por múltiples versiones del protocolo, muchas de las cuales son compatibles con las anteriores. El RFC 2145 describe el uso de los números de versión de HTTP. El cliente le dice al servidor al principio de la petición la versión que usa, y el servidor usa la misma o una anterior en su respuesta.

HTTP/0.9

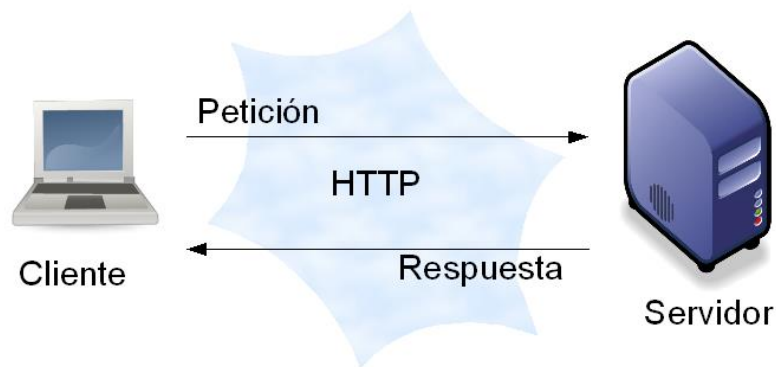
Obsoleta. Soporta sólo un comando, GET, y además no especifica el número de versión HTTP. No soporta cabeceras. Como esta versión no soporta POST, el cliente no puede enviarle mucha información al servidor.

HTTP/1.0 (mayo de 1996)

Esta es la primera revisión del protocolo que especifica su versión en las comunicaciones. Cada petición enviada entre cliente y servidor requiere establecer una nueva conexión entre ambos, lo cual puede sobrecargar al servidor.

HTTP/1.1 (junio de 1999)

Es la versión más utilizada del protocolo. Permite establecer conexiones persistentes entre cliente y servidor disminuyendo la carga del servidor y optimiza la velocidad de transferencia de información. Adicionalmente, permite al cliente enviar múltiples peticiones a la vez por la misma conexión sin esperar respuestas (pipelining) lo que hace posible cargar una página completa con múltiples recursos en una sola petición (e.g. imágenes, contenido estático, etc.).




Intercambio de recursos utilizando el protocolo HTTP


2. Normativas XML

Normativa XML

- Es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium <http://www.w3.org/xml>
- No es realmente un lenguaje en particular, sino, una manera de definir lenguajes para diferentes necesidades
- Se propone como estándar para intercambio de información estructurada entre diferentes plataformas
- Se puede aplicar en bases de datos, editores de texto, hojas de cálculo entre otros



El diagrama muestra un archivo XML centralizado con flechas que indican su conversión a varios formatos: PDF, Word, Excel, PowerPoint, y otros. Debajo del diagrama se encuentra el texto 'XML Conversion'.

1 - 5Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

2.1. XML (Extensible Markup Language)

Es uno de los formatos más utilizados para la representación y transferencia de información. Es un estándar basado en el lenguaje de marcas (<http://www.w3.org/xml/>) que permite que los datos sean descritos independientemente del lenguaje en el que está escrita una aplicación.

Los documentos XML se pueden analizar de dos formas y son las siguientes:

- Por estructura:
 - Un documento XML cuenta con una estructura válida si cumple con todas las reglas básicas del lenguaje de marcas.
 - Una condición básica para trabajar con un documento XML es que esté bien formado.

- Por semántica:
 - Un documento XML cuenta con una semántica válida si cumple con unas reglas de gramática previamente definidas.
 - La “gramática” se puede definir mediante el uso de los siguientes mecanismos:
 - DTD (Document Type Definitions)
 - XSD (XML Schema Definition)

Los DTD y XSD definen el contenido de un documento XML. Pueden indicar el orden de los elementos, cuáles de ellos son obligatorios y cuales son opcionales. Asimismo, definen el rango de valores permitidos y otras características que hacen posible que una aplicación interprete satisfactoriamente el documento XML.

Inicialmente, un DTD era la única forma de describir un documento XML, pero a medida que su utilización evolucionó, comenzaron a surgir algunas restricciones como las que a continuación mencionamos:

- Los DTD no están expresados en XML, son simples documentos de texto, y, en consecuencia, su proceso y manipulación no es tan fácil como en XML.
- Los DTD no permiten que el valor de un elemento al ser escrito sea cualquier otro distinto de un string.
- Los DTD no proporcionan un mecanismo para definir el ámbito de un tag y proteger el nombre de una colisión (permiten ambigüedad de tags).

Los XML Schemas fueron introducidos para evitar las limitaciones de los DTD y, actualmente, son la opción preferida sobre éstos. Dichos esquemas están escritos en XML. Los nombres de los elementos escritos dentro del documento están asociados al concepto de Namespace, el cual proporciona una solución a la colisión o ambigüedad de nombres.

Un esquema XML (XSD) describe la estructura de construcción de un documento XML casi similar a un DTD.

- Define elementos que pueden aparecer en el documento.
- Define atributos que pueden aparecer en el documento.
- Define qué elementos tiene hijos.
- Define el orden de los elementos hijos.
- Define el número de elementos hijos.
- Define si un elemento puede estar vacío o puede incluir texto.
- Define tipos de datos para los elementos y sus atributos.
- Define valores por defecto, valores fijos para los elementos y sus atributos.

Los XML Schemas reemplazan a los DTD's por muchas razones como las siguientes:


- Son extensibles para adiciones futuras.
- Están escritos en XML.
- Soportan tipos de datos.
- Soportan Namespaces.


```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<mensaje>
  <remitente>
    <nombre>Juan Perez</nombre>
    <email>jperez@cibertec.edu.pe</email>
  </remitente>
  <destinatario>
    <nombre>Jorge Cáceres</nombre>
    <email>jcaceres@cibertec.edu.pe</email>
  </destinatario>
  <asunto>Java Architect</asunto>
  <texto>Fundamentos y normativas XML</texto>
</mensaje>
```

3. Normativas JSON

Normativa **JSON**

- JSON es el acrónimo de JavaScript Object Notation
- Es un formato liviano, utilizado para el intercambio de datos entre aplicaciones
- Se propone como una alternativa a XML ya que es capaz de manejar información estructurada entre plataformas.
- La ventaja principal frente a XML es su simplicidad de uso, esto debido a que su sintaxis está basada en JavaScript



1 - 8
Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.


3.1. JSON (JavaScript Object Notation)

Es un formato liviano para intercambio de datos que se propone como alternativa a XML, debido a que es capaz de manejar información estructurada entre diferentes plataformas.

La simplicidad de JSON ha permitido la generalización de su uso, especialmente como alternativa a XML en AJAX. Una de las ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es relativamente sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando la función `eval()`, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la compatibilidad de JavaScript en casi cualquier navegador web.


Representación en XML

```
<customer id="12">
  <first>Juan</first>
  <last>Perez</last>
  <kids>
    <kid>Julio</kid>
    <kid>Jorge</kid>
  </kids>
</customer>
```

Representación en JSON

```
{
  "id" : 12
  "first" : "Juan",
  "last" : "Perez",
  "kids" : ["Julio", "Jorge"]
}
```

4. XML vs JSON



- JSON y XML son dos principales formatos para intercambio de información. Ambos pueden ser usados para representar a información estructurada, objetos y diversos tipos de datos

Descripción	XML	JSON
Tipos de datos	representación de datos no es implícita	representación de datos es implícita
Suporte de arreglos	representación por convenciones	soporte nativo
Suporte de objetos	representación por convenciones	soporte nativo
Soporte de nulos	requiere importación del namespace	soporte nativo
Comentarios	soporte nativo	no soportado

1 - 10 Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.

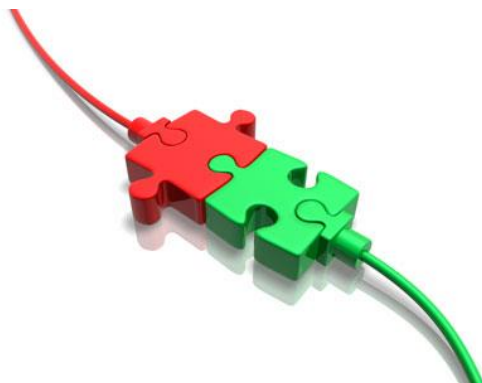
4.1. Comparativa entre XML y JSON

JSON y XML son los principales formatos para intercambio de información entre aplicaciones. Ambos pueden ser usados para representar información estructurada, objetos y diversos tipos de datos. A continuación, se revisarán sus características más resaltantes:

Característica	XML	JSON
Tipos de Datos	La representación de datos no es implícita en el documento y debe realizarse mediante un XML Schema (XSD) el cual soporta una gran variedad de tipos de datos.	La representación de datos es implícita en el documento. Soporta tipos de datos escalares e información estructurada mediante arreglos y objetos.
Suporte de arreglos	Los arreglos deben ser representados por convenciones, algunas veces utilizando un elemento padre y varios elementos hijos anidados.	Soporte nativo de arreglos.
Suporte de objetos	Los objetos deben ser representados por convenciones, algunas veces combinando elementos y atributos.	Soporte nativo de objetos.

Característica	XML	JSON
Soporte de nulos	Requiere importación del namespace xsi:nil.	Soporte nativo a valores nulos.
Comentarios	Soporte nativo y disponible mediante APIs.	No soportados.
Espacios de nombres	Soporte extensible para namespaces, lo cual elimina el riesgo de colisión de nombres de elementos.	No existe el concepto de espacio de nombres.
Validación de formato	Posee un XML Schema estándar encargado de validar el contenido de cada documento XML en forma realmente extensible.	No posee una forma estándar de validación de contenido de documentos JSON. El JSON Schema está aún en proceso de maduración.
Tamaño	Los documentos XML tienden a ser grandes en tamaño, especialmente cuando se utilizan muchos elementos anidados.	Los documentos JSON poseen un tamaño reducido debido a que manejan una sintaxis simple y una representación ligera de los datos.
Procesamiento en JavaScript	Requiere un analizador sintáctico complejo XML DOM y librerías adicionales para convertir la información en objetos JavaScript.	Su procesamiento es tan simple como utilizar la función eval() en JavaScript.

Curva de aprendizaje	Generalmente requiere conocimiento de varios conceptos tales como XSD, XSLT, XPath, XML Namespaces, DOM entre otros.	Aprendizaje rápido debido a que es una tecnología basada en un lenguaje de programación como lo es JavaScript.
Utilización y estándares	Es una forma de transferencia de información muy utilizada y se encuentra asociada con varios estándares del mercado como SOAP, WSDL, IFX (Interactive Financial eXchange) entre otros.	Es mayormente utilizado por aplicaciones web que utilizan AJAX y que se enfocan en maximizar la velocidad de transferencia de información. e.g. Google, Yahoo.



XML para Integración



JSON para velocidad

5. Herramientas para el procesamiento de XML y JSON

Herramientas Java para el procesamiento XML y JSON

- Java Architecture for XML Binding (JAXB): utilizada para las transformaciones de objetos Java a las representaciones en formato XML y viceversa
- Apache XMLBeans: manipulación del contenido XML con los objetos Java y viceversa. Posee utilidades para manejo de esquemas XSD y navegación de un archivo XML
- Jackson Processor: es una librería Java desarrollada como alternativa a los formatos XML. Mejora, velocidad y tamaño de las transformaciones en XML utilizando formato JSON



JSON
Jackson Processor



The Apache POI Project



1 - 14

Copyright © Todos los Derechos Reservados - Cibertec Perú SAC.



JAXB (Java Architecture for XML Binding): Es una especificación desarrollada bajo el JSR 222 en su versión 2.0 y es utilizada para transformar objetos Java a representaciones en formato XML y viceversa.

Apache XMLBeans: Es una herramienta desarrollada por “Apache Software Foundation” que ayuda a manipular contenido XML mediante objetos Java y viceversa. Posee una serie de utilidades para el manejo de esquemas XSD y navegación de un archivo XML a través de cursores.

Jackson Processor: Es una librería Java desarrollada como alternativa a los formatos XML. Su propósito es mejorar la velocidad y tamaño de las transformaciones en XML utilizando otro formato llamado JSON (JavaScript Object Notation).