

Formatos para intercambio de información

Como lograr que compartan información el universo de diferentes sistemas disponibles hoy en día? Mediante los formatos para representación de información.

Son formatos robustos, con estructuras jerárquicas, atributos variables, y al mismo tiempo muy fáciles de leer para un humano.

- XML
- JSON
- YAML

XML Extensible Markup Language

Lenguaje Extensible de Marcado

Qué es XML?

Es un lenguaje para organizar contenidos. No está diseñado para ser visualizado o procesado por ningún programa, navegador o aplicación.

Sintaxis

Al igual que HTML es un lenguaje de etiquetas (apertura-cierre y vacías).

El nombre y contenido de cada etiqueta lo define el desarrollador.

Los archivos tienen extension `.xml`

Cómo se interpreta?

Los navegadores muestran el código tal cual.

Mozilla, Chrome y Opera muestran además que les falta información de estilo asociada.

Explorer muestra el código tal cual, pero no dice que le falta algo.

Safari muestra el texto en una sola línea

XML es la base para construir una estructura

Se le debe asociar un CSS para visualizarlo

XML Extensible Markup Language

Lenguaje Extensible de Marcado

Usos

- Transmitir información en servicios web y APIs REST.
- Archivos de configuración.

Ventajas

- Soporta Unicode permitiendo usar cualquier idioma del mundo.
- Amplio soporte en la actualidad.

Desventajas

- Complejo y poco descriptivo para mapear una estructura usando tipos de datos.
- Para documentos grandes requiere uso intensivo de memoria y procesador

JSON Javascript Object Notation

Notación de Objetos de Javascript

Formato ligero de texto para intercambio de datos, alternativa a XML.

En sus inicios fue considerado como parte de Javascript, pero siempre ha sido independiente.

Es independiente del lenguaje de programación y es soportado de modo nativo o por medio de librerías en la mayoría de los lenguajes.

En los últimos 15 años se ha vuelto omnipresente en la web y hoy en día es el formato elegido por casi cualquier web service disponible.

La popularidad de JSON ha dado como resultado que sea soportado también por muchas bases de datos.

Las bases de datos relacionales como PostgreSQL y MySQL incluyen soporte nativo para almacenar y consultar datos en JSON.

Las bases de datos NoSQL como MongoDB y Neo4j también lo soportan, aunque algunas pueden usar versiones ligeramente modificadas como MongoDB.

JSON Java Script Object Notation

Notación de Objetos de Javascript

Usos

Es comúnmente usado (pero no exclusivamente) para el intercambio de información entre clientes y servidores web. Una estructura JSON puede ser pasada por un servidor web a un navegador cliente o a una aplicación móvil que entonces hará alguna acción sobre ella, como desplegar los datos o grabarlos para posteriormente hacer referencia a ellos.

Ventajas

- Simplicidad.
- Fácil de leer
- Fácil implementación.
- Gran desempeño.
- Los navegadores modernos incluyen funciones nativas para codificar y decodificar JSON.

Desventajas

- **Unicode** El Unicode presenta variaciones en las implementaciones.
- **Seguridad** Riesgo por la facilidad de ejecutarlo usando la función `eval` de JavaScript.
- **Sin esquema** : Esto da flexibilidad total, pero puede provocar datos incorrectos.
- **Solo un tipo numérico** : Nos impide aprovechar la diversidad de tipos en los lenguajes de programación.
- **No hay tipo fecha** : Estas deben ser representadas como cadenas, lo que puede provocar discrepancias en el formato
- **No hay comentarios** : Se requiere documentación adicional para ello.
- **Verbal** : Aunque es menos verbal que XML, no es muy conciso. Para alto volumen o servicios de propósito especial es mejor optar por un formato, mas eficiente.

JSON Javascript Object Notation

Notación de Objetos de Javascript

Sintaxis La sintaxis es idéntica a la que Javascript usa para la creación de objetos.

Es una representación en texto plano de datos estructurados.

Esta basado en una lista de pares llave-valor

JSON es un formato de datos genérico con una cantidad minima de tipos:

- Cadenas.
- Números.
- Booleanos.
- Listas.
- Objetos.
- Null.

Aunque su notación es un subconjunto de Javascript, estos tipos son soportados en muchos lenguajes de programación, haciendo de JSON un excelente candidato para transmitir datos entre lenguajes.

Esta constituido por dos estructuras universales que todos los lenguajes de programación soportan:

| Estructura | En los lenguajes se les conoce como: |
|---------------------------------|--|
| Colección de pares nombre/valor | objeto, registro, hash, array asociativo, etc. |
| Lista ordenada de valores | arreglo, vector, lista, etc. |

JSON Javascript Object Notation

Notación de Objetos de Javascript

| Tipo | Descripción |
|----------|---|
| Objeto | Conjunto desordenado de pares <code>nombre:valor</code> separados por coma. Comienza y termina con corchetes <code>{}</code> |
| Arreglo | Es una colección de valores separados por coma. Comienza y termina con corchetes <code>[]</code> |
| Valor | Puede ser <code>string</code> , <code>número</code> , <code>true/false</code> , <code>null</code> , <code>objeto</code> o <code>arreglo</code> . Estas estructuras pueden anidarse |
| String | Colección de cero o más caracteres Unicode. Comienza y termina con comillas dobles <code>" "</code> |
| Caracter | Un carácter es representado por una cadena de un carácter |

XML vs JSON

XML cada día es menos común y es fácil ver porque.

Una misma estructura representada en XML y en JSON:

```
<?xml version="1.0"?>
<alumno>
  <control>12345678</control>
  <nombre>Juan</nombre>
  <apellido>Perez</apellido>
  <materias>
    <materia>
      <nombre>Matematicas I</nombre>
      <calificacion>100</calificacion>
    </materia>
  </materias>
</alumno>
```

```
{
  "control": 12345678,
  "nombre": "Juan",
  "apellido": "Pérez",
  "materias": [
    {
      "nombre": "Matematicas I",
      "calificacion": 100
    }
  ]
}
```


XML vs JSON

Ademas de ser mas extenso y verbal, convertir xml a un objeto Javascript puede llevar muchas líneas de código y requiere adecuaciones basadas en el objeto que va ser interpretado.

Convertir JSON a un objeto Javascript toma una línea de código y no requiere conocimiento previo del objeto que se va recibir.

YAML Yet Another Markup Language

Otro lenguaje de marcado

Documento de texto plano legible para un ser humano.

Facilita el mapeo de estructuras de datos mas complejas.

Como listas y arreglos asociativos.

Es muchisimo mas legible que XML y JSON.

Ventajas

- Mas estricto.
- Mas simple y veloz.
- Elegancia y claridad.
- Ideal para tareas que involucren intervención de un humano.

Usos

- Archivos de configuración, depuración.
- Otros fines en los que se requiera facilidad de lectura.
- Ampliamente usado en Ruby y en el framework Ruby on Rails.

JSON VS YAML

```
{ "Alumnos": [  
  {  
    "Alumno": {  
      "control": 12345678,  
      "nombre": "Fulanito",  
      "apellido": "Perez"  
    }  
  },  
  {  
    "Alumno": {  
      "control": 87654321,  
      "nombre": "Menganito",  
      "apellido": "Lopez"  
    }  
  }  
]  
}
```

```
- Alumnos:  
  - Alumno:  
    control: 12345678  
    nombre: Fulanito  
    apellido: Perez  
  - Alumno:  
    control: 87654321  
    nombre: Menganito  
    apellido: Lopez
```