

Serialización/deserialización de objetos y transmisión de datos con JSON: una revisión de la literatura

Object serialization/deserialization and data transmission with JSON: paper review

Juan Antonio Mora-Castillo¹

Fecha de recepción: 22 de abril del 2015

Fecha de aprobación: 27 de agosto del 2015

Mora-Castillo, J. Serialización/deserialización de objetos y transmisión de datos con JSON: una revisión de la literatura. *Tecnología en Marcha*. Vol. 29, N° 1, Enero-Marzo. Pág 118-125.

¹ Gerente de Desarrollo de Sistemas en la empresa Desert Development Company. Estudiante de la Maestría en Administración de la Tecnología de Información, Universidad Nacional, Costa Rica. Correo electrónico: jmora_castillo@ice.co.cr Teléfono: (506)8842-7994.

Palabras clave

Desarrollo de software; JSON; transmisión de datos; serialización/deserialización de datos; revisión de literatura.

Resumen

En este artículo se presentan los resultados de la revisión de literatura sobre uno de los métodos de serialización/deserialización de objetos que usa la notación JSON (siglas en inglés de JavaScript Object Notation, o Notación de Objetos de JavaScript).

Uno de los resultados más importantes de dicha revisión es que JSON es el método con mayor desempeño a la hora de serializar/deserializar objetos. También se concluye que no existen muchas revisiones de literatura sobre el uso de JSON como herramienta de serialización/deserialización sino casos prácticos. Si se toma en cuenta que la serialización/deserialización de objetos es un mecanismo por el cual un objeto pasa de ser objeto a texto, es transportado a cualquier destino y finalmente se vuelve a transformar a objeto, dicha información teórica resulta importante, ya que enriquecería más el tema para los usuarios de JSON en el desarrollo de sus aplicaciones.

Keywords

Software Development; JSON; Data Transmission; Data Serialization/Deserialization; Literature Review.

Abstract

The present systematic literature review is to review the literature of one method of serialization / deserialization of objects which uses JSON notification. Among the most important results of this review is that JSON is the method with higher performance when serializing / deserializing; also, a conclusion is that there are not many reviews literature on the use of JSON as serialization tool / deserialization but practical cases. Considering that the serialization / deserialization of objects is a mechanism by which an object changes from a text object is transported to any destination and finally becomes transform object. This theoretical information would be important because the issue further enrich users JSON in developing their applications.

Resumen

Introducción

Hoy en día, los desarrolladores de aplicaciones *web* enfrentan el reto de incluir en sus programas algoritmos de serialización/deserialización y transmisión de datos que permitan convertir objetos de diferentes tipos a texto, transportarlos y finalmente volver a ser el objeto que eran antes.

Según la definición que proporciona Microsoft en su sitio web oficial para desarrolladores MSDN, JavaScript Object Notation (JSON) es un formato de codificación eficaz que permite intercambios rápidos de cantidades pequeñas de datos entre los exploradores de cliente, tales como Internet Explorer, Google Chrome y servicios web, por lo cual se le considera un gran método de desarrollo web para la serialización/deserialización en la creación de aplicaciones.

La literatura seleccionada explica el proceso de serialización/deserialización y transporte de objetos por medio de JSON, con un enfoque de comparación del rendimiento sobre otras tecnologías similares.

Esta revisión de literatura busca ofrecer un panorama del uso de JSON como librería de programación para realizar la serialización/deserialización y transmisión de datos en el campo del desarrollo de aplicaciones web, con el fin de ofrecer información sobre cómo otros investigadores profundizaron en el tema.

Método de revisión de la literatura

Para llevar a cabo la revisión de la literatura se realizó un proceso iterativo de lectura a través de la investigación de otras revisiones de literatura, artículos y páginas web oficiales que tratan el tema del desarrollo de aplicaciones y de diferentes tecnologías relacionadas con la serialización/deserialización de objetos mediante la notación JSON.

Esta revisión de literatura se apoyó en la taxonomía de Cooper para definir cada uno de los aspectos sugeridos por dicho autor, dando un enfoque de investigación de resultados, cuyo objetivo es la exploración del tema de serialización/deserialización de objetos y transmisión de datos por medio de JSON.

La perspectiva aplicada, según la taxonomía de Cooper, es una representación neutral de la información brindada por los artículos encontrados e incluidos en dicha revisión de literatura, proporcionando una cobertura representativa de la información.

Entre otros aspectos, está la cobertura, que es de tipo representativo; el tipo de organización, que es conceptual; y la audiencia a la que va dirigida la presente revisión de literatura, que está constituida por personas relacionadas con el área de desarrollo de aplicaciones web, por lo que se considera una audiencia especializada.

Una vez clasificada la información, se definieron los principales temas y conceptos relacionados con JSON, serialización/deserialización y transmisión de datos, con el fin de argumentar la búsqueda y evaluación de la teoría.

Las búsquedas realizadas para encontrar la información fueron ejecutadas en Google Académico, que fue la base de datos en la que se pudo encontrar la información con los criterios de búsqueda primarios utilizados en este buscador.

El criterio número uno fue “JSON” + “Serialization” + “Literature Review”, con un resultado de 127 hipervínculos, de los cuales se utilizaron los artículos llamados “Issues of Modeling Web Information Systems: A Literature Review” y “Evaluation of Protocol Buffers as Data Serialization Format for Microblogging Communication. A Literature Review”.

Seguidamente, se ejecutó otro criterio de búsqueda para obtener más documentación referente al tema. El criterio ejecutado fue “JSON” + “Serialization”, con un resultado de 3680 hipervínculos de los cuales se tomaron dos artículos llamados “Comparison of JSON and XML Data Interchange Formats: A Case Study” y “Comparison between JSON and YAML for data serialization: A Case Study”.

También se consideró como fuente secundaria la página MSDN de Microsoft y se realizaron los mismos filtros en el buscador de la página, con un resultado de 153 hipervínculos, de los cuales se tomó el artículo “Serialización JSON”.

Durante la selección de temas, se hizo un diagrama de afinidad en el cual se organizaron los temas que estarían cubiertos por la revisión de literatura, como son: 1) la serialización/deserialización de objetos, 2) transmisión de datos y 3) la notificación notación JSON. El primer tema se subdividió en los subtemas 1.1) desempeño con respecto a XML, 1.2) desempeño

con respecto a YAML y 1.3) transformación de objetos a textos y viceversa. El segundo gran tema elegido para esta revisión de literatura se dividió en los siguientes subtemas: 2.1) envío y recepción de mensajes y 2.2) buen desempeño en redes computacionales. Y por último, el tercer gran tema tiene como subtemas 3.1) JSON y la serialización de objetos y 3.2) JSON y la transmisión de datos.

Introducción a los conceptos y formulación de la revisión de literatura

JSON (JavaScript Object Notation)

Según MSDN, el sitio *web* oficial para desarrolladores de Microsoft, JSON es un formato de codificación de datos eficaz que permite intercambios rápidos de cantidades pequeñas de datos entre los exploradores de cliente como Internet Explorer, Google Chrome y servicios web, por lo cual se le considera una gran librería de desarrollo web para la serialización/deserialización y la transmisión de datos.

El artículo “Comparison between JSON and YAML for data serialization” indica que JSON fue especificado e introducido por Douglas Crockford en 2001, utilizándolo en su compañía. Crockford no fue el creador pero sí fue el primero en darle una especificación completa.

Ahora bien, si se revisa el artículo “Comparison of JSON and XML Data Interchange Formats”, JSON fue diseñado para ser un lenguaje de intercambio de datos comprensible para los seres humanos, fácil de interpretar y usar por las computadoras. JSON es directamente soportado por el lenguaje de programación JavaScript y es la mejor opción para la serialización/deserialización e intercambio de datos.

Además, se estima que JSON puede serializar/deserializar 100 veces más rápido que otras tecnologías, tales como las librerías de XML en navegadores modernos, pero a pesar de esto, existen muchas quejas de que JSON no puede manejar NameSpaces, validaciones de datos ni tiene extensibilidad, lo que es visto como una debilidad de dicha tecnología, pero este tema no será abordado en el presente artículo.

La sintaxis de JSON debe poder ser interpretada por los humanos de una forma fácil. La figura 1 describe cómo se utiliza JSON para codificar el nombre y apellido de un objeto persona.

```
{
    "firstname" : "John",
    "lastname"  : "Smith"
}
```

Figura 1. Codificación de nombre y apellido de un objeto persona.

Imagen tomada del artículo “Comparison of JSON and XML Data Interchange Formats”, de los autores Nurzhan Nurseitov, Michael Paulson, Randall Reynolds y Clemente Izurrieta, del departamento de Ciencias de la Computación de la Universidad Montana State Bozeman, Estados Unidos.

Serialización/deserialización de objetos con JSON

El artículo “Evaluación of Protocol Buffers as Data Serialization Format for Microblogging Communication” menciona que la serialización es un proceso para convertir datos en grandes

cantidades de bits llamados en inglés Stream, que luego pueden ser enviados por la red o almacenados en bases de datos. Lo opuesto a esto se conoce como deserialización, que sería volver un objeto a su estado natural. Este proceso lo utilizan la mayoría de aplicaciones en internet.

Existen dos tipos de serialización/deserialización: 1) el que tiene formato de texto para poder ser interpretado por el ser humano y las computadoras, y 2) la binaria, que solo es interpretada por las computadoras. El presente artículo se enfoca en el formato de texto que puede ser interpretado por el ser humano, como se indica en el ejemplo (figura 2) de un objeto persona serializado por medio de JSON.

```
message Person {
  required string name = 1;
  required int32 id = 2;
  optional string email = 3;
  enum PhoneType {
    MOBILE = 0;
    HOME = 1;
    WORK = 2;
  }
  message PhoneNumber {
    required string number = 1;
    optional PhoneType type = 2;
  }
  repeated PhoneNumber phone = 4;
}
```

Figura 2. Ejemplo de un objeto persona serializado por medio de JSON. Fuente: Imagen tomada del artículo “Evaluation of Protocol Buffers as Data Serialization Format for Microblogging Communication”, de Canggih Puspo Widowo.

La interpretación de la imagen anterior, según el autor, es un objeto persona que tiene las siguientes propiedades 1) Nombre, 2) Identificación, 3) Email, 4) Tipo de teléfono, 5) Número de teléfono.

El sitio web oficial de desarrolladores de Microsoft presenta el siguiente ejemplo de cómo se puede llevar a cabo la serialización/deserialización de un objeto a través del lenguaje JavaScript utilizando la notación de JSON.

El primer paso es definir la clase que contendrá el objeto que se va a serializar, como se muestra en la figura 3.

```
[DataContract]
class Person
{
    [DataMember]
    internal string name;

    [DataMember]
    internal int age;
}
```

Figura 3. Clase de objeto que se va a serializar. Fuente: Imagen tomada del sitio web MSDN de Microsoft (2015).

Seguidamente, se realiza el proceso de serialización, como se ilustra en la figura 4.

```

Person p = new Person();
//Set up Person object...
MemoryStream stream1 = new MemoryStream();
DataContractJsonSerializer ser = new DataContractJsonSerializer(typeof(Person));
ser.WriteObject(stream1, p);
    
```

Figura 4. Proceso de serialización. Fuente: Imagen tomada del sitio web MSDN de Microsoft (2015).

Con lo cual se genera el resultado que se describe en la figura 5.

```
{“age”:42,“name”:“John”}
```

Figura 5. Resultado al ejecutar código. Fuente: Imagen tomada del sitio web MSDN de Microsoft (2015).

Por último, el receptor debe realizar el algoritmo mostrado en la figura 6 para deserializar el objeto.

```
Person p2 = (Person)ser.ReadObjectct(stream1);
```

Figura 6. Proceso de deserialización del objeto. Fuente: Imagen tomada del sitio web MSDN de Microsoft (2015).

Transmisión de datos con JSON

Según el artículo “Comparison of JSON and XML Data Interchange Formats”, se realizó un caso práctico en el que se utilizó JSON y otra tecnología para comparar la serialización/deserialización y transmisión de datos en un ambiente igual para ambas tecnologías, con un servidor receptor y un cliente emisor de iguales características para la prueba. Se seleccionaron ciertas medidas para ejecutarlas en la prueba, básicamente con el fin de medir el rendimiento de las dos tecnologías.

En esta prueba se pudo constatar que JSON corre más rápido en la serialización/deserialización de datos, pero requiere de más recursos de CPU, lo que no es una limitante hoy en día. Ambas tecnologías hicieron el mismo uso de la memoria y en la transmisión de datos las dos hicieron el mismo uso de CPU y Memoria.

El artículo “Comparison between JSON and YAML for data serialization” hizo la misma investigación, pero en vez de utilizar XML como contraparte, usó YAML. De manera similar, JSON resultó ser mucho más rápido a la hora de serializar/deserializar datos pero utilizando mayores recursos. La transmisión fue similar entre ambas tecnologías.

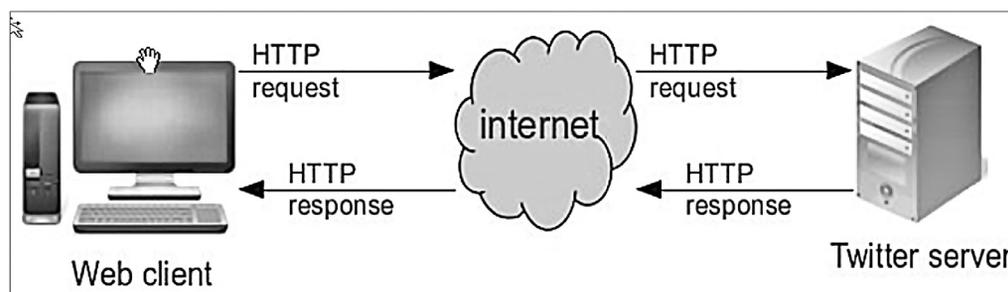


Figura 7. Arquitectura de las pruebas aplicadas por los investigadores. Fuente: Imagen tomada del artículo “Evaluation of Protocol Buffers as Data Serialization Format for Microblogging Communication”, de Canggih Puspo Widowo.

El artículo “Esquema de Servicios para Televisión Digital Interactiva, Basados en el Protocolo REST-JSON” indica que una buena práctica para el transporte de mensajes a través de internet es el protocolo REST. El protocolo REST y JSON conforman la plataforma completa requerida para un rápido envío de datos serializados que es consumido por cualquier aplicación por medio de una dirección URL, proceso llamado servicios web, según el autor. En el artículo se menciona que este esquema es el utilizado hoy en día por importantes páginas web como Facebook y Twitter.

Resultados y discusión

Basado en la información encontrada en las fuentes primarias y secundarias se generaron las siguientes notas, que constituyen un resultado esperado debido al uso que hoy en día tiene la tecnología JSON en sitios web importantes a nivel mundial.

1. JSON debe ser fácil de interpretar por el ser humano y el resultado final de su serialización siempre es texto plano.
2. JSON es una librería utilizada para la serialización y transmisión de datos en las plataformas web.
3. JSON es considerada la librería más rápida para la serialización y deserialización de datos con respecto a XML y YAML; sin embargo, hace mayor uso de recursos de CPU.
4. JSON es considerada similar a otras tecnologías con respecto a la transmisión de datos, pero se puede utilizar el mecanismo de servicios web, como lo es el protocolo REST para efectuar una buena transmisión de datos.
5. Después de la comprensión del artículo, los investigadores serán capaces de seleccionar JSON como herramienta de serialización/deserialización, ya que se demostró, por medio de pruebas encontradas en las revisiones de literatura y casos de estudio, que es la forma más rápida de serializar y deserializar objetos.
6. Actualmente, la serialización/deserialización con JSON es un tema muy técnico, por lo que no existen muchas revisiones de literatura en las bases de datos consultadas.

Teniendo en cuenta los resultados de las pruebas ejecutadas en los artículos y siendo JSON la notación más utilizada en la industria del software web en las principales páginas, se podría recomendar su uso a la comunidad desarrolladora de software.

Conclusión

En este artículo se presentaron los resultados de una revisión de literatura sistemática sobre la serialización/deserialización de objetos y transmisión de datos con JSON. Se definió un criterio de búsqueda y se seleccionaron elementos de la literatura representativos, que fueron evaluados y seleccionados para tomarlos en cuenta en este trabajo a partir de fuentes primarias como revisiones de literatura y casos de estudio. Además, se tomó la información referente al problema planteado y se utilizaron algunas otras fuentes secundarias para ilustrar el tema.

La revisión de literatura se enfocó en la información sobre el uso de JSON para el proceso de serialización/deserialización. También se revisaron los casos de estudio sobre la comparación de JSON con otras tecnologías con relación a su desempeño. Se revisó la literatura sobre transmisión de datos, enfocando el tema del transporte de datos serializados y concluyendo con los resultados recomendados por las revisiones de literatura acerca del uso de JSON como herramienta para el desarrollo web.

En conclusión, se considera que la literatura es mala, ya que existe mucha información técnica y ejemplos de código de programación pero muy poca información teórica o revisiones de literatura sobre la serialización/deserialización y la transmisión de datos utilizando la notación JSON.

La principal limitación del estudio de la revisión de literatura es la cantidad excesiva de información técnica que se encontró, por lo que se tuvieron que especificar más puntualmente las necesidades que tenía esta revisión a partir de criterios en la búsqueda y del conocimiento del autor como especialista en tecnología. Otra de las limitaciones fue el poco tiempo que se tuvo para llevar a cabo la revisión, que fue de un mes.

Como trabajo futuro, se recomienda hacer más revisiones de literatura sobre el tema, ya que es mucha la información sobre como implementar JSON en programas computacionales pero poca la información teórica. También es aconsejable realizar pruebas funcionales del uso de JSON con más tecnologías; profundizar en su rendimiento con respecto a la serialización y deserialización de objetos y, por último, agregar más información acerca del porcentaje de uso a nivel mundial; es decir, conocer un aproximado de la cantidad de sitios web que usan JSON y están registrados en dominios oficiales.

Bibliografía

Chanchí, G.E. (2011). *Esquema de servicios para Televisión Digital Interactiva, basados en el Protocolo REST-JSON*.
Obtenido de <http://scholar.google.com/>

Ericksson, M. Hallberg, V. (2011). *Comparison between JSON and YAML for data serialization: A Case Study 2011*.
Obtenido desde la página: <http://scholar.google.com/>

Molnar, B. & Benczura, A. (2013). *Issues of Modeling Web Information Systems*.

Nurseitov, N., Paulson, M. & Reynolds, R. (2013). *Comparison of JSON and XML Data Interchange Formats: A Case Study*. Obtenido de <http://scholar.google.com/>

Página web:

Serialización JSON, Microsoft MSDN Web Page. (01/04/2015). Obtenido de [https://msdn.microsoft.com/es-es/library/bb410770\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/bb410770(v=vs.110).aspx)