

Analysing Impact of Delimiters on the Size of JSON Data Interchange Format

Harpreet Singh Padda¹ and Gulabchand K. Gupta²

¹Research Scholar, Department of Computer Science, JJT University, Jhunjhunu, Rajasthan, India

² Department of Computer Science, JJT University, Jhunjhunu, Rajasthan, India

Abstract: *JSON (JavaScript Object Notation) is the most used data interchange format for communication between applications and devices over the Internet. The other major formats include XML, CSV, Protocol Buffers, Atom, YAML and many more formats which are currently in existence but the major share remains between JSON and XML. JSON is lighter in size and faster to process as compared to XML which has been discussed and proven in various comparison studies. JSON is a text based format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language. In this paper we discuss about the amount of extra data that is included in a JSON string in the form of delimiters and escape characters using examples and conclude by deriving formulas to calculate the amount of delimiters used based on the composition of JSON data string.*

Key Words: JSON, Delimiters, Data Interchange Format, XML

1. INTRODUCTION

JSON (JavaScript Object Notation) is a prominent data interchange format used by various applications to transfer data over the network. JSON (JavaScript Object Notation) [1-2] is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. JSON has characteristics of being small in size and faster transmission speed as compared to the other popular data interchange format i.e. XML [3]. In one of the papers author did a quantitative and qualitative comparison of around twelve libraries including JSON wherein it was shown that JSON performs better than XML [4]. There have been studies wherein translation of XML to JSON has been discussed to improve transmission speed [5]. JSON is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December

1999. The JSON format is such that it requires use of delimiters to separate out the keys from the values as well as multiple key-value pairs from each other. Additionally the JSON format requires use of escape characters to be used along with special characters which are part of the data string. In this research paper we analyze the amount of data that gets inserted in the form of delimiters and escape characters into the final JSON string adding to the data size of the actual data that needed to be transferred. We look at sample JSON data strings having different number of key-value pairs as well as special characters in it and then draw a relationship among these variables.

2. JSON FORMAT

The data in JSON consists of the following two components:

- a. Key-Value pairs
- b. Ordered list of items / Array

To represent a Key-Value pair in a JSON string, the following format is used:

```
{
  "<key1>": "<value1>",
  "<key2>": "<value2>",
  "<key3>": "<value3>",
  .....
};
```

To represent an ordered list of items / Array, the following format is used:

```
[
  {
    "<item1- key1>": "<value>",
    "<item1- key2>": "<value>",
    "<item1- key3>": "<value>",
    ....
  },
  {
    "<item2- key1>": "<value>",
    "<item2- key2>": "<value>",
    "<item2- key3>": "<value>",
    ....
  }
  .....
];
```

A key can also have an array of values in the following format:

```
{
  "myArray": [ "a", "b", "c", "d",... ]
}
```

A key can also have key-value pairs as its value in the following format:

```
{
  "myObject": {
    "key1": "value1",
    "key2": "value2",
    ....
  }
};
```

Additionally, any special characters that appear in either key or value need to be escaped. The list of special characters used in JSON is:

- \b Backspace (ascii code 08)
- \f Form feed (ascii code 0C)
- \n New line
- \r Carriage return
- \t Tab
- \" Double quote
- \\ Backslash character

Fig. 1 shows a pictorial representation of the characters that need to be escaped in JSON:

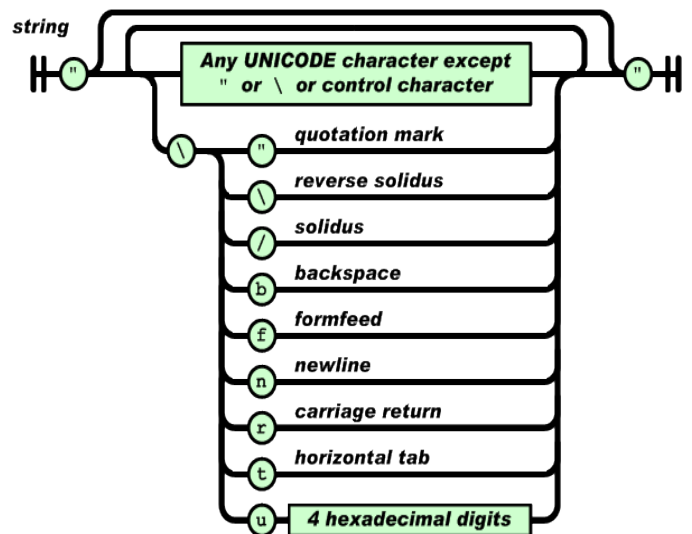


Fig - 1:JSON data format [1]

Any string or characters in JSON [1] have to be encoded in Unicode format. The default encoding used in JSON is UTF-8.

Example JSON String:

```
{
  "array": [
    1,
    2,
    3
  ],
  "boolean": true,
  "null": null,
  "number": 123,
  "object": {
    "a": "b",
    "c": "d",
    "e": "f"
  },
  "string": "\"Hello World\""
}
```

3. ANALYSIS OF SPACE TAKEN BY DELIMITERS & SPECIAL CHARACTERS

To analyse the volume of delimiters included in a JSON string we reviewed the JSON format and defined the following variables for which delimiters are needed:

- KV** - Number of key/value pairs
- Arr** - Number of Arrays / Objects
- ArrIt** - Number of items in an Array / Object
- Del** - Number of delimiters in the string

For a single KV we need 5 Del's as shown in the example below:

Del-1	“
	key1
Del-2	”
Del-3	:
Del-4	“
	value1
Del-5	”

In case of an array, for each Arr we need 5 Del's as shown in the example below:

Del-1	"
	Array
Del-2	"
Del-3	:
Del-4	[
Del-5]

Additionally for each array item, ArrIt, we need 1 Del for ArrIt>1 as shown in the example below:

	ArrIt-1
Del-1,	
	ArrIt-2

Based on the above observations we propose the following formulas to calculate the number of delimiters in a JSON string:

- $1KV=5Del$
- $1Arr=5Del$
- $1ArrIt=1Del-1$

Consider the below string having 2 KV's 3 Arr's with each having 2 ArrIt's, the number of Del's would be calculated as follows:

$$(2 \times 5Del) + (2 \times 5Del) + ((2 \times 1Del) - 1) + ((2 \times 1Del) - 1) = 22Del$$

4. CONCLUSIONS

JSON is extensively used for communication between applications which thrive to have the data transferred at the maximum speed possible. Thus any decrease in the JSON data packet size would have a direct impact on the responsiveness of these applications.

From the above illustrations, it's clear that a lot of space is used in a JSON data packet by delimiters and escape characters. Although delimiters are required to separate the content into logical blocks we see a huge potential in optimizing the use of delimiters and thus bringing down the size of the JSON packet. The amount of delimiters in a small JSON string might be negligible but for applications have a large JSON data string removal or optimal use of these delimiters / escape characters can lead to huge space saving and thus improved speed.

REFERENCES

- [1]. Introducing JSON. <http://www.json.org>
- [2]. T. Bray, Ed., Google Inc. IETF, RFC 7159. March 2014
- [3]. Boci Lin, Yan Chen, Xu Chen, Y ingying Yu, "Comparison between JSON and XML in applications on AJAX", International Conference on Computer Science and Service System, 2012.
- [4]. Kazuaki Maeda, "Performance Evaluation of Object Serialization Libraries in XML, JSON and Binary Formats", IEEE (2012) pages 177-182
- [5]. G. Wang, "Improving Data Transmission in Web Applications via the Translation between XML and JSON", Proceedings of third International Conference on Communications and Mobile Computing (CMC), (2011) pages 182-185.

BIOGRAPHIES



Harpreet Singh Padda did his MSc. in Computer Science from University of Mumbai in the year 2004. He has more than 12 years of experience working on .Net and web technologies like HTML, jQuery, Javascript and other. Currently he is pursuing Ph.D. in Computer Science from JIT University



Dr. Gulabchand K. Gupta has worked in Science, Engineering & Polytechnic colleges as well as in Management Institutes. He has more than 28 years of teaching experience. He has done M.Sc. (Electronics), M.Tech. (Computer Science & Engineering) and completed his Ph.D. degree from University of Mumbai in the year 1999. He has taught diverse set of subjects. He has published more than 35 research papers in national and international journals & conferences and has guided M.Sc, M.E., M.Phil and Ph.D students. He is a life member of "Computer Society of India" and "Indian Science Congress Association"