Analysis of the Efficiency of Data Transmission Format Based on Ajax Applications

Peng Wang, Xiaodong Wu, Huamin Yang
Department of Computer Science and Technology, Changchun University of Science and Technology, 130022,
Changchun ,Jilin, China
e-mail: wpeng@cust.edu.cn

Abstract- **In order to shorten the response time from Ajax real-time applications as far as possible, the client can get data from server timely and users can obtain timelier and more quickly real-time information. Choosing proper data transmission format can effectively shorten the analyzing time, reduce data file redundancy, and bring much convenience to system development and maintenance. Experiments show that when analyzing JSON type data objects, the computer spend less time than XML data type and more time than HTML fragments, but with low file redundancy and more convenience for programming and maintaining. So in some cases, use JSON data type as transmission type will bring faster user experience, and cost on development and maintenance also can be reduced.**

Keywords-Data Transmission Format ; Ajax ; Efficiency ; Comparison

I. INTRODUCTION

In some real-time Web applications, Ajax (Asynchronous JavaScript and XML) does play an important role in these real-time applications due to its support for asynchronous transmission and local refresh. Ajax consists of HTML, JavaScript, DHTML (Dynamical Hypertext Markup Language) and DOM [1] (Document Object Model) components. Since some Web applications require relatively high real-time data, the client side does not refresh the entire page in the case of asynchronous real-time data transmission. Ajax has a variety of data formats to achieve the client and server communication. Common data exchange formats are HTML, XML, JSON, CSV (Comma Separated Values), GZIP. Among them HTML is a relatively simple data transfer format. XML as a generic markup language, has cross-platform, cross-language advantage.While, as for the new generation of plain text data format, JSON has a clear advantage in data exchange of Ajax application. Without considering the case of network transmission, there is an obvious difference in efficiency for computer to analyze objects of different data formats. This article will mainly explore and compare the consuming time on parsing data objects and size of generated data files based on HTML、XML and JSON data interchange format especially.

II. AJAX DATA TRANSMISSION FORMAT

HTML（Hypertext Mark-up Language）is not only currently the most widely used network markup language, but also the main language constituting and describing a web page document. HTML text is a kind of descriptive text composed by HTML tags, in lightweight Ajax applications[2], client and server side exchange data usually in the form of HTML fragments when using HTML as transmission format.

XML[3]（Extensible Markup Language） which like HTML, all belong to SGML(Standard Generalized Markup Language).XML is a cross-platform, dependent on the content technology. XML is designed to transfer and carry data, not for performance or display of data, while HTML language is used to show data. Therefore, the focus of XML is that it explains what data is, and carries data information. Using a series of simple tags to describe the data, these markers can be a convenient tool to create data format used by server-side. The most basic form of a XML document includes statements, handling instructions (optional) and element. XML documents are all starting from a root node, the root node contains a root element, and other elements must be included in the root element. Nested elements are called child elements, child elements can also contain child elements, and data can exist in both their child elements and properties.

And, like XML, JSON [4] is also a kind of data format based on plain text. Since JSON is prepared for JavaScript, JSON data format is very simple. JSON can be transmitted with a simple String, Number or Boolean type variable; you can also use an array, or a complex object. In practice, object data type is usually used for exchanging, while objects are tagged by unordered {} which contains a series of key-value pair Key-Value in JSON.

III. FACTORS THAT AFFECT THE REAL-TIME RESPONSE

Without taking different performances of servers processing the data into account, factors affecting real-time response include:

1. Network transmission environment
2. Transmission and Parsing speed of different data formats.

Because of Ajax's asynchronous transmission and local refresh features, and supposing transmission in the network environment keeps in a level, the difference on the transmission speed of different data formats is not obvious. However, different data formats produce different redundant

978-0-7695-4522-6/11 $26.00 © 2011 IEEE
DOI 10.1109/ICM.2011.199

data objects so there are data files of diverse sizes. At the same time, the time which the computer spends on parsing data object based on different data formats is obvious not the same.

IV. COMPARISON OF THE DATA TRANSMISSION FORMAT

An example about real-time map of geographic information [5] will be taken of illustrating the performance of the transmission between server and client side when using different data formats. Server-side and client interaction information include regional, longitude, latitude information.

A. Transmission information based on HTML data format

When using HTML as a manner of transmission format ,it needs to make the coordinate information into a text string according to certain rules,when information encoding is completed on the server side, then the encoding string will be sent to client-side.In client-side string will be decoding. For example， transmission string is like this{ location：changchun；Longitude：120.37689；Latitude：44.91285 }.So HTML fragment can be like this "%changchun#120.37689#44.91285%". The connection string transfers data using conventional symbols to organize information. On client-side the string will be decoded as location, Longitude, Latitude and geographic information which need to be displayed in real time.

HTML fragments, which is concise, do not waste too much bandwidth resources. But on the server or client side constructing the HTML text string is not conducive to programming, commissioning and maintenance is either not easy. Particularly in the transmission of text data structure complex situations, the debugger will highlight the difficulty of maintaining it.

B. Transmission information based on XML data format

Non-empty XML tags must be composed of the beginning and ending tags, which the marked content is between. Start tag is beginning with "<" and ending with ">"; in the middle is name tag. End tag is beginning with "</" and ending with ">", in the middle is name tag. Between the start tag and end tag is the value of the mark. To rewrite the above HTML fragment into the XML form, it can be like this:

```
<coordinate>
    <Area>changchun</Area>
    <XCoordinate>44.91285</XCoordinate>
    <YCoordinate>120.37689</YCoordinate>
</coordinate>
```

<coordinate> is the root, <XCoordinate> and <YCoordinate> are all child nodes. The coordinates' information is stored in these nodes.

C. Transmission information based on JSON data format

JSON contains simple numerical values, such as string and number type. However, using these simple data types to define complex data format is undoubtedly complex and difficult. In practice, JSON use object type to describe complex data type. When mapping the XML message into JSON format, the converted format is as follows [6]:

```
coordinate:{
 'area':'changchun',
 'xcoordinate': '44.91285',
 'ycoordinate': '120.37689'
  }
```

'coordinate' is a JSON object.In JSON data attributes and values are separated by ":", and different attributes of the same object are separated by ",".

V. EFFICIENCY AND ANALYSIS TEST ON DIFFERENT TYPE DATA FORMATS

Analysis of efficiency of data objects, mainly were focused on the analytical speed and the size of two data files in the following test.

A. Html fragment

In this study, the rules are not used to transfer data in the form of strings, but with concrete instead of div tag. Several objects can be loaded into HTML pages when using Ajax lightweight framework JQuery's load function. Following is the way to achieve:

```
$("#DataReslultDiv").load("CustomerOrders.htm");
```

Load function can get remote html fragments and insert them into DOM elements.

Single HTML object can be described like this:

```
<div class="comment"> area:changchun; xcoordinate: 44.91285; ycoordinate: 120.37689</div>
```

B. XML[7]

Ajax function in JQuery can read and parse Xml file and nodes.

```
$.ajax({
        type: "get",
        url: "CustomerOrders.xml",
        dataType: "xml",
        success: function(xml) {
            var StartTime = new Date();
            $("#DataReslultDiv").empty();
            $(xml).find('data').each(function(i) {
                var Name = $(this).find('area').text();
                var EndTime = new Date();
var txtHtml = "<div class='comment'>" + i + ":" + Name + ":TimeSpan->" + (EndTime - StartTime) + "</div>";

$("#DataReslultDiv").append(txtHtml);
            });
```

```
        }
    });
```

'type' defines the way to get xml, 'url' is the location where the file to visit is, 'dataType' defines the file type(in this case, it is xml type),and 'success' is the function when it successfully returns to call. When function returns, XML nodes will be parsed and the time cost on parsing will be recorded.

The following is a manifestation of the parsing xml object:

```
<data>
        <area>changchun</area>
        <xcoordinate>44.91285</xcoordinate>
        <ycoordinate>120.37689</ycoordinate>
</data>
```

In the object, 'data' is a root node. While 'area'、'xcoordinate'、'ycoordinate' are all child nodes, which separately record location、latitude、and longitude information.

C. JSON

JQuery supplies getJSON() function to access JSON file:

```
$.getJSON('CustomerOrders.js', function(data) {
        var StartTime = new Date();
        $("#DataReslultDiv").empty();
        $.each(data, function(i, item) {
            var Name = item['area'];
            var EndTime = new Date();
            var txtHtml = "<div class='comment'>" + i + ":" + Name + ":TimeSpan->" + (EndTime - StartTime) + "</div>";

$("#DataReslultDiv").append(txtHtml);
        });
    });
```

The first parameter is the function is the file name to be visited, and the second is the function to be executed when visit is successful. The value which the function returns is JSON type data format. In the function JSON array elements will be parsed and the parsing time is also recorded.

Single JSON type data object is:

```
{
"area" :    "2",
"xcoordinate" :    "44.91285",
"ycoordinate" :    "120.37689"
}
```

Environment for tests[8]:

CPU: AMD Athlon(tm) 64 X2 Dual-Core Processor TK-53;

Memory:2G;

OS:Windows XP sp 2.

In large number of parser objects case, test one demonstrates the time which the computer spends on parsing ordinary HTML, XML and JSON object.

Table I: Test one（The time consuming in large number of parsing objects case）

	HTML	XML	JSON
Object Count	10000	10000	10000
Total Time(ms)	308	25675	19484
Average Time(ms)	0.030	2.567	1.948

From test one it is easy to see using Html as transmitting data format can get faster response time than XML and JSON data format [9]. For data quantity is little and data structure is not complex case, Html should be first selection of data transmission format. Meanwhile, the computer parses JSON objects faster than XML objects. Though XML has very good generalization, in certain application environment using JSON as transmission data format will shorten the response time.

Test two records with increasing parsing object number cases (the number parsing object for 100, 500, 1000, 2000, 5,000 respectively) the cost of total and average time when the computer parsing HTML fragment, XML and JSON objects.

Table II: Test two（Time consuming on increasing parsing objects.）

	HTML	XML	JSON
Object Count	100	100	100
Total Time(ms)	14	176	119
Average Time(ms)	0.14	1.76	1.19
Object Count	500	500	500
Total Time (ms)	30	912	648
Average Time(ms)	0.150	1.806	1.296
Object Count	1000	1000	1000
Total Time(ms)	45	1789	1341
Average Time(ms)	0.045	1.789	1.341
Object Count	2000	2000	2000
Total Time(ms)	98	3886	2550
Average Time(ms)	0.049	1.943	1.275
Object Count	5000	5000	5000
Total Time(ms)	212	10922	8080
Average Time(ms)	0.042	2.184	1.616

From test two it can be seen paring single HTML object time doesn't increase with the number of parsing objects. When paring objects number is small (100,500 etc.), the computer will spend more time on HTML objects than other two data format. But when the node number increase (1000, 2000, and 5000) single Html object parsing time consuming keep on the same level. And parsing one JSON object is faster than XML object. And XML object information redundancy is much bigger than JSON object. Below are comparisons on file size in increasing number of parsing objects based on HTML, XML and JSON data format.

Table III: File size comparison on different data format

	HTML	XML	JSON
Object count	100	100	100
File Size（KB）	7	16	8
Object count	500	500	500
File Size（KB）	32	76	31
Object count	1000	1000	1000
File Size（KB）	64	148	57
Object count	2000	2000	2000
File Size（KB）	126	302	114

Object count	5000	5000	5000
File Size（KB）	315	738	294

The file size of HTML fragment and JSON is similar, and the file size is proportional to the number of parsing objects. When the parsing object's structure is linear structure (not nested structure), the size of the file which XML objects generate is twice than HTML fragment and JSON objects. Once the data structure becomes complicated nested structure, this ratio also increases.

EPILOGUE

Through the simulation test we can reach the following conclusion.

The parsing speed of HTML fragments objects is fastest, while with low data redundancy. The parsing speed of JSON objects is almost closed to HTML fragments, but with lowest data redundancy. And the parsing speed of XML objects is slowest. Besides, XML objects generate high data redundancy.

HTML is used for displaying data, not the best format to carry data, and using HTML as transmission data format will bring inconvenience to coding and decoding in practical projects. JSON have an advantage of describing data information, which also has parsing fast and low data redundancy characters. But it is slightly worse in term of generality. Parsing XML objects is slowest, and with high data redundancy. But XML has a popular generality. In practical applications, data format should be chosen depending on different application backgrounds.

REFERENCES

[1] W3C Document Object Model.http://www.w3.org/DOM,2005.

[2] Crane D,Pascarello E,James D. Ajax in action .Manning, 2005.

[3] Extensible markup language (xml) 1.0.http://www.w3.org,2006.

[4] JSON.Introducing JSON.http://www.json.org,2006.

[5] MinHang,HaoFeng.The analysis on geographical information data exchange research methods based on JSON.Surveying and mapping scientific,2010,159~161.

[6] XML and JSON.http://book.51cto.com/art/200803/67482.htm,2008.

[7] Ramon Lawrence. The space efficiency of XML .Information and Software Technology, 2004, (46) :753~759 .

[8] Smith K. Simplifying Ajax-style Web development .Computer, 2006, 395, 39(5) :98-101 .

[9] Smullen,C.and Smullen,S.AJAX Application Server Performance.-Proceedings of the IEEE SoutheastCon 2007,154-158.