International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 1– No.1, November 2012 – www.ijais.org

Energy Efficient Data Formatting Scheme: A Review and Analysis on XML Alternatives

P Shadiya
Department of Computer Science and Engineering, MES College of Engineering
Kuttippuram, Kerala, India

P P Abdul Haleem
Department of Computer Science
Farook College, Calicut
Kerala, India

ABSTRACT

A format for data representation and associated facilities can be referred as a Data Formatting Scheme (DFS). The de-facto standard format is eXtended Markup Language (XML). XML is advantageous because of its unique features such as structured code, secure, self-descriptiveness etc,. But XML has an inherent limitation due to the abundant use of tags - verbosity. Several alternative proposals are being discussed in the literature to solve this problem. In this paper a review on different alternative methods for XML are done. Three categories of methods are explored in this paper - XML compression, alternative binary formats for XML and alternative non-binary formats for XML. It is observed that there is a scope for data formatting schemes based on Non-binary formats due to their readability and less verbosity.

Keywords- Data Formatting Schemes, XML, Alternative XML Methods

1. INTRODUCTION

Data Formatting Schemes refers to the format for data representation and associated facilities such as schema awareness, trust management and security specifications. XML [1] is the de-facto standard. The unique features [2][3][4][5] of XML such as platform independence, schema awareness, structured code, security and self-descriptiveness lead to the universal use of XML as a format for data representation. In additional to the intended use as a messaging format, XML is also being used in applications such as message syntax format and in web services [6][7]. In spite of all the advantages, XML has an inherent limitation due to the abundant use of tags - verbosity. XML document can be 3 to 20 times large as compared to its binary or text file representation [8].

1.1 MOTIVATION

Verbosity invariably affects bandwidth, processing power, and energy consumption [2][9]. There is an abundant penetration of constrained wireless mobile devices into the traditional networking arena. Wireless mobile devices impose a bunch of limitations (with limited resources such as battery power, memory, processing power and input and screen) [2]. Several limitations (such as unreliable channels, low bandwidth, increased latency, increased rate of retransmission of lost packets and weak security features) of wireless networks are also reported in the literature) [2]. One of the main concern with wireless mobile devices is energy consumption [10]. The largest chunk of power consumed in a wireless mobile device is during the transmission and reception of the messages [11]. Packet losses and retransmissions through the relatively unreliable channels result in further resource crunch - the packet loss in Wireless Local Area Network (WLAN) is estimated to be 30 percent, which results in heavy drain of battery power [3].

It can be concluded that measures for size reduction of messages to achieve energy conservation is very important, especially in constrained wireless mobile devices. Reduction of message size results in reduction in the number of packet transmissions, which in turn reduces the data transmission cost and storage space [11]. In the wireless world, reducing the message size is of primary importance than the processing efficiency gains through an alternative format [2]. Hence there is a scope for alternative mechanisms.

The rest of the paper is organized as follows: Section 2 discusses about the state of the art. In section 3, a brief discussion about the analysis and observations is presented. Section 4 concludes the paper.

2. STATE OF ART

Several solutions are proposed in the literature to overcome limitations of XML. Some of the topics explored under XML alternative mechanisms are XML compression techniques, alternative binary data formatting schemes for XML, and alternative non binary data formatting schemes for XML.

2.1 XML Compression Techniques

These techniques are developed to reduce the usage of storage space and network bandwidth [12]. A brief discussion about the merits and demerits of the prominent methods of this category are given below.

2.1.1 GZip [13]: It is the most widely used commercial compressor available. gZip uses a combination of the LZ77 algorithm and Huffman coding. Since XML is represented as a text file, an obvious choice for a compression tool would be a general-purpose compression tool such as gZip.

Advantages of gZip [13] include: (i) this tool is widely available in both open-source and commercial implementations, (ii) gZip provides better compression rate (40-50 percent) and freedom from patented algorithms, (iii) gZip requires no knowledge of the document-structure for its usage, and (iv) gZip is built into http and web-servers as a standard feature.

Compression of elements/attributes may be limited in gZip due to the long-range dependencies between elements and between attributes – this is considered as a disadvantage of this method [13].

2.1.2 XMill [14]: It is based on a regrouping strategy that takes advantage of the XML elements. XMill groups XML text strings with respect to their meaning and exploits similarities between those text strings for compression [14].

XMill achieves better compression rate [14] compared to gZip. But it does not support searching through the compressed data [14].

2.1.3 XGrind [14]: It is a compression tool for XML documents that supports querying the compressed document. At the same time, this tool retains the structure of the original XML document too. This facilitates reuse of standard XML techniques for processing the compressed document [14]. It is advantageous due to considerable improvements in query response time. But the problem is only XML data that has DTD structure can be compressed using this method [14]. .

2.1.4 Xpress [14]: It is a compressor that supports direct and efficient evaluations of queries on compressed XML data. XPress adopts a novel encoding method, called reverse arithmetic encoding.

Advantages of Xpress [14] include: (i) it achieves significant improvements on query performance for compressed XML data and reasonable compression ratios and (ii) it does not make use Document Type Definition (DTD).

Advantages of XML compression techniques [12][13] can be summarized as (i) reduced storage size and network bandwidth, (ii) reduces main memory requirements of processing and querying XML documents, and (iii) reduced cost of transmitting XML data over the network.

Disadvantages of XML Compression Techniques [12][13][15] include as (i) additional overhead due to compression and decompression, (ii) not human readable, and (iii) platform dependent.

2.2 Alternative Binary Data Formatting Schemes for XML

It is a compact representation of XML document in binary formats [18]. It is used when ordinary XML document is not an option due to performance limitation. The intent is to decrease the size and reduce the required processing at remote nodes [16]. There are many different alternatives for XML in binary formats such as Wireless Binary XML (WBXML), XML Binary Infoset Encoding (XBIS), and Abstract Syntax Notation One (ASN.1) [17][18].

2.2.I. WBXML [16]: It is developed to support wireless device as it is transmitted in a compact manner over mobile networks. By reducing file size, WBXML addresses a key power management issue in mobile devices. It provides many of XMLs benefits and incur minimal overhead. But the main disadvantage [16] of using WBXML is that it does not compress the values that are not specified in the Document Type Definition (DTD).

2.2.2. XBIS [16]: XBIS is an encoding format that eliminates the padding of XML text documents. It retains the XML schema format of the native data. It is faster to generate and interpret, but it is unsuitable due to its performance degradation [24].

2.2.3. ASN.1 / Fast Infoset [16]: It is an International Telecommunications Union (ITU) standard (X.891) predating XML. It provides smaller encoding sizes and faster processing than a World Wide Web Consortium (W3C) XML representation. It is modified to provide more data security features. It is widely used in telecommunication industry [24].

Advantages of Binary Data Formatting Schemes [18] can be summarizes as follows: (i) it reduces the verbosity of XML document, and (ii) it enables random access and indexing of XML documents.

Disadvantages of Binary Data Formatting Schemes [18] can be summarized as follows: (i) there is additional burden due to encoding or decoding of text document to binary format, (ii) it is platform dependent, and (iii) not human readable.

2.3 Alternative Non Binary Data Formatting Schemes for XML

Non-Binary data formatting schemes discussed under this section includes Open Node Syntax (ONX) [20], Java Syntax Number One (JSON) [21], Simple Outline XML (SOX) [22], YAML Ain't Markup Language (YAML) [23], and Thinned YAML (TYAML) [23].

2.3.1. ONX [20]: It is designed to be data-oriented mark-up language. Here the data is represented in a generic and compact way. It retains the end tag concept in XML, but care is taken to see that the overall verbosity is less than that of XML.

2.3.2. JSON [21]: It is a compact, text based, programming language model data interchange format. It includes a subset of the JavaScript language. It is human readable and supports Unicode. But it is not suitable for large data and multimedia formats.

2.3.3. SOX [22]: SOX supports XML features like elements, attributes, and text. Other parts such as processing instructions, comments, and entities of XML are not supported by SOX.

2.3.4. YAML [23]: YAML is highly human readable and lightweight data serialization language. It is portable between programming languages. It is expressive and extensive i.e., matching with the native data structures of agile languages. It supports the essential features of XML such as human readability, schema awareness and namespaces. But it is very sensitive to whitespace and special characters.

Thus the advantages of Non- Binary XML [20][21][22][23] are as follows: (i) overall verbosity less than XML, (ii) human readable, (iii) no additional overhead due to encoding or decoding, and (iv) built in data structures are present especially in case of YAML.

3. ANALYSIS AND OBSERVATIONS

Performance comparison of three different XML alternatives is described in the Table 3.1 Different performance parameters considered are (i) human readability (whether it is easily readable), (ii) verbosity (number of data structure used), (iii) additional overhead due to encoding and decoding, (iv) platform independence, and (v) support for random access and queries.

Table 1: Performance Comparison of XML Alternatives

	Compression Techniques	Binary Format	Non Binary Format
Human readable	NO	NO	YES
Verbosity	LESS	LESS	LESS
Additional overhead due to encoding/ decoding	MORE	MORE	NO
Platform independence	YES	NO	YES
Support random access and queries	NO	YES	YES

From table 3.1 it can be concluded that non binary alternative formats for XML retains the main features of XML such as human readability and platform independence. At the same time, these formats have reduced verbosity.

4. CONCLUSION

Despite the advantage of XML as a legacy standard for data representation and exchange, XML is crippled with an inherent limitation – verbosity. Increased verbosity results in difficulties in utilizing XML in wireless mobile networks and devices. This gave way to many alternative mechanisms for XML. These mechanisms include XML compression techniques, XML alternative binary formats and XML alternative non binary formats. Even though XML compression techniques and XML alternative binary formats help in verbosity reduction, it can be concluded that XML alternative non binary formats retains almost all features of XML sans verbosity.

5. REFERENCES

[1] Maletic. J. I, Collard. M. and Kagdi. H, "Leveraging XML technologies in developing program analysis tools", In Proceedings of 4th International Workshop on Adoption-Centric Software Engineering (ACSE'04), Edinburgh, Scotland, vol. 25, pp. 80-85, May 2004.

[2] J. Kangasharju, T. Lindholm, and S. Tarkoma, "XML security with binary XML for mobile web services," In International Journal of Web Services Research (IJWSR), vol. 5, no. 3, pp. 1-19, 2008.

[3] Paradiso and Starner, "Energy scavenging for mobile and wireless electronics", In IEEE Pervasive Computing, vol. 4, pp. 1827, 2005.

[4] P. Gokran, "Making your low-cost handsets plans succeed", In in Telecom Era. Global Creations Group, India, November 2007, pp. 2332.

[5] H. Artail, M. Shihab, and H. Safa, "A distributed mobile database implementation on pocket pc mobile devices communicating over bluetooth", In Journal of Network and Computer Applications (JNCA), vol. 32, pp. 96115, January 2009.[Online].Available:http://portal.acm.org/citation.cfm?id=1460932.1461106.

[6] J. Kangasharju, T. Lindholm, and S. Tarkoma, "XML messaging for mobile devices: From requirements to implementation", In The International Journal of Computer and Telecommunications Networking, vol. 51, no. 16, pp. 4634-4654, 2007.

[7] H.-H. Lee and W.-S. Lee, "Consistent collective evaluation of multiple continuous queries for filtering heterogeneous data streams", In Knowledge and Information Systems(KAIS), vol. 22, no. 2, pp. 185210, February 2010.

[8] H. R. Elliotte, XML Bible. New York, NY, USA: John Wiley Sons, Inc., 2003.

[9] Maletic. J. I, Collard. M. and Kagdi. H, "Leveraging XML technologies in developing program analysis tools", In Proceedings of 4th International Workshop on Adoption-Centric Software Engineering (ACSE'04), Edinburgh, Scotland, vol. 25, pp. 80-85, May 2004.

[10] M. P. Michael, "Energy Awareness for Mobile Devices," In Research Seminar on Energy Awareness, University of Helsinki, 2005.

[11] R. Potlapally, Ravi and Niraj, "Heterogeneous grid computing for energy constrained mobile device," In IEEE transactions on Mobile Computing, vol. 5, IEEE, 2006, pp. 128-143.

[12] S. S. Nair, "XML compression techniques: A survey", Citeseer, June 2007, [Online] Available: http://citeseer.ist.pst.edu/ .

[13] Tanakorn Wichaiwong and Chuleerat Jaruskulchai, "XML Retrieval More Effcient Using Compression Technique", In Proceedings of the International Multi Conference of Engineers and Computer Scientists 2011 Vol I, IMECS 2011, March 16 - 18, 2011, Hong Kong.

[14] Ericsson. M,"The e_ects of xml compression on SOAP performance," World Wide Web (2007), Springer, vol. 10, pp- 279-307.

[15] Augeri, C.J, Bulutoglu, D.A, Mullins, B.E, etal, "An analysis of XML compression efficiency", In Proceedings of the 2007 workshop on Experimental computer science, ACM, pp. 7.

[16] Snyder, S.L, "Efficient XML Interchange (EXI) Compression and Performance Benefits: Development, Implementation and Evaluation", DTIC Document, 2010.

[17] Bayardo, R.J, Gruhl, D, Josifovski, V. and Myllymaki, J, "An evaluation of binary xml encoding optimizations for fast stream based xml processing", In Proceedings of the

13th international conference on World Wide Web, ACM, pp. 345-354, 2004.

[18] Bruce Martin, Bashar Jano, "WAP Binary XML Content Format, W3C NOTE 24 June 1999, [Online]. Available: http://www.w3.org/TR/wbxml/ .

[19] S. Jacobs, Open node syntax version 0.6.9, June 2005, [Online] Available: http://www.seairth.com/web/onx/onx.html/ .

[20] J. Kangasharju, T. Lindholm, and S. Tarkoma, "Requirements and design for XML messaging in the mobile environment", In Anerousis, N., Kormentzas, G., eds.:Second InternationalWorkshop on Next Generation Networking Middleware. (2005) 2936, 2005, pp. 29-36.

[21] Simple outline xml: SOX, 2002, http://www.langdale.com.au/SOX/.

[21] O. Ben-Kiki, C. Evans, and I. dt Net, "YAML aint markup language, YAML version 1.2, 3rd edition," January 2010, Available at http://www.yaml.org/spec/1.2/spec.htm/

[23] P. P. Abdul Haleem and M. P. Sebastian, "An Energy Conserving Approach For Data Formatting and Trusted Data Exchange in Resource Constrained Networks", Knowledge and Information Systems, Volume 32, Issue 3, pp 559-587, September 2012

[24] T. Devadithya, Zongde Liu, etal. "BXSA for Fast Processing of Scientific.Data", Proceedings of the 2007 spring simulation multiconference, Vol. 2, pp. 441-446.