Binary XML and its Characterization

Robin Berjon, XML Prague, 25/06/2005

WHAT IS BINARY XML?

‣ **It's not XML**

- not in XML syntax

- doesn't comply to the XML specifications

- can't give it to an XML parser, it won't work

- your text editor won't like it

- it's even small and fast

WHY CALL IT XML THEN?

- **XML is cool!**

- **Marketing is fun!**

- **We're evil!**

Seriously...

▸ **It's *very* XML related**

- you can feed it into an XML application easily

- instances can map onto the data models used with XML

- when done right (or at least, not *too* wrong) it has many of the properties found in XML documents

▸ **People will call it "Binary XML" anyway**

WHAT'S THE DEBATE ABOUT?

- It's about Love

- But not love of listening

- The wrong reasons tend to be used on both sides

Typical Arguments

▸ **Examples:**

- **Pro (bad).** XML is always inefficient; text is always slow; human readable is useless; XML is too complex and over-engineered.

- **Pro (good).** Strong use cases; not too hard to get right; increased universality of XML.

- **Con (bad).** Just use gzip; all binary formats are evil and proprietary; it is impossible to obtain such gains; Moore's Law will lead to world peace.

- **Con (good).** Interoperability issues to consider; feasibility to be proven; decreased universality of XML

XML's Universality

- **Diverging opinions on the universality of XML**

- **Everyone is right:**

  - if you have all you need, it's there

  - otherwise it's not

- **XML is universal on much less than half the existing devices**

Use Cases

- **A small sample**

  - Mobile devices

  - Large multimedia documents

  - Web Services

- **This is just a small set, but hopefully they are different enough**

MOBILE DEVICES

▶ **More of these than fixed ones**

▶ **WAP mostly failed**

▶ **Bandwidth, but mostly parsing time**

▶ **Moore's law doesn't help**

- battery lifetime

- heat

LARGE DOCUMENTS

▸ **10k-200k pages documents don't scale easily in XML, especially with embedded multimedia**

▸ **Transcoding, splitting, etc. are painful and error-prone**

▸ **But documents *want* to be in XML**

▸ **Simple things are needed:**

  • random access, or accelerated access

  • random or at least efficient update

Messaging

- **Problems occur when the volume of messages is high, or when real-time is needed**

- **Jabber server handling thousands of messages per second**

- **Web Services for fast moving objects**

And Much More

- We saw typical if sometimes a bit extreme examples; there are more

- Ad hoc solutions are easy, generic ones harder (but possible)

- The verticality of use cases is illusory, in real life they are all increasingly mixed together

The XBC WG

- **Workshop in August 2003**
- **WG for a year starting March 2004**
- **Chartered with defining the problem**
- **Ended in success**
- **Produced four documents**
- **Follow-up being discussed**

Use Cases

▸ **XBC Use Cases**

- http://www.w3.org/TR/xbc-use-cases/

- those we've seen, and quite a few more

- not all are use cases in the same way

Properties

▶ **XBC Properties**

- http://www.w3.org/TR/xbc-properties/

- A vocabulary to speak coherently of the topic

- Not a shopping list

- Can be reused in other situations

Measurement Methodologies

‣ **XBC Measurement Methodologies**

- http://www.w3.org/TR/xbc-measurement/

- Describes how the properties are measured

- Makes the abstract discussion concrete

- Can sometimes be difficult to read as some properties are hard to measure

- Will be used to verify that a binary XML format actually works as promised

Characterization

▸ **XML Binary Characterization**

- http://www.w3.org/TR/xbc-characterization/

- Synthesis and conclusion

- Some may wish to read it first

- Requirements are defined using a mechanical process

- The result is surprisingly close to XML

- Conclusion: binary XML is both needed and feasible, therefore the W3C needs to produce a standard for it

W3C Involvement

- Based on the output from the XBC WG, the W3C is to decide whether it needs to define a binary XML format

- The decision hasn't taken place yet

- The W3C Team is investigating options

- But the decision will come soon

A Binary Future?

▸ **Sadly, binary XML seems unavoidable**

▸ **In fact it's already there**

▸ **It could go bad**

- no W3C standard, multiple conflicting and vertical formats, proprietary solutions, no interoperability, users suffer

▸ **Or not**

- a W3C standard, a few niche formats, general interoperability, a few problems at first but easily overcome, XML becomes more widespread

THANK YOU

- Any questions?

IMPLEMENTATIONS

▸ **Some of the many formats:**

- Efficient XML (Agile Delta)

- esXML (Stephen Williams)

- Fast Infoset (ISO ASN.1, Sun, OSS Nokalva)

- MPEG-7 BiM (ISO MPEG, Siemens, Expway)

- X.694 (ISO ASN.1, Sun, OSS Nokalva)

- Xebu (Helsinki Institute of Technology)

- XEUS (KDDI)

DECISION TREE

▸ **Does XML support the property?**

- Yes. Binary XML should support the property.

- No. Does XML support this property when combined with other parts of the XML stack?

  Yes. Binary XML should work with the other parts of the XML stack.

  No. Is it feasible for XML to support this property?

  - *Yes. The property should be addressed by a general approach (e.g., new recommendation) that works for both XML and Binary XML.*

  - *No. The property should be directly supported by Binary XML.*

Requirements

Directly Readable and Writable
Transport Independence
Compactness
Human Language Neutral
Platform Neutrality
Integratable into XML Stack
Royalty Free
Fragmentable
Streamable
Roundtrip Support
Generality
Schema Extensions and Deviations
Format Version Identifier
Content Type Management
Self Contained

Processing Efficiency
Small Footprint
Widespread Adoption
Space Efficiency
Implementation Cost
Forward Compatibility