Standardize Binary Representation of XML?

Michael Rys
Shankar Pal
Jonathan Marsh
Andrew Layman
Microsoft Corporation, Redmond

“Text” XML vs “Binary” XML

- XML 1.0
  - Text representation, human readable
  - Successful as portable, platform-independent format
  - Uses more bits for encoding than theoretical min
- Ubiquitous format
  - All data can be rendered into textual XML form
  - All XML parsers can process
  - Text-processing tools available for manipulation
- “Binary XML” — encoded using fewer bits
  - Save parsing time
  - Saves transmission bandwidth

Problems of "Standard Binary XML"

- Complicates the XML landscape
- Plurality of new forms of XML
- Increases barrier of entry for working with XML
  - Vendors/users have to support text and binary forms
- Can splinter into multiple dialects addressing different requirements:
  - Infoset/XQuery Data Model Preservation
  - Memory Footprint
  - Parsing/Generating Speed
  - Random Access vs Streaming
  - Data-only Compression
  - Other Application-specific Needs
- Is "binary XML" a good candidate for standardization?

Infoset Preservation

- Infoset has weak conformance requirement
- Infoset/XQuery Data Model preservation for portability
  - Binary representation must preserve Infoset/DM
  - Or be isomorphic to Infoset/DM content of XML value
  - Note: Binary DOM format — not fully isomorphic to Infoset
- XML Schema or DTD should be optional
  - Use schema for optimizations
  - Encode PSVI in the binary representation
  - Can improve parsing speed
- Infoset or XQuery Data Model may be extended
  - Binary format will change
  - Continual maintenance of the standard

Memory Footprint

- "Binary XML" has smaller mem. footprint than text XML
- Compression techniques — Gzip, XMill, …
  - Very good compression
  - Decompress into text XML by recipient before consumption
  - Two passes of data required for parsing
  - Relatively large parse time
  - Whole XML must be compressed and decompressed
  - Chunking mitigates the issue to large extent
- Suitable when high compression ratio is required
  - Low bandwidth connection
  - Generation and parsing costs are less of concern
  - Storage and retrieval are predominant operations
    - Stored in files/database server, data caching, messaging, …
- Tradeoff between smaller memory footprint and higher parsing cost

… Memory Footprint

- On server, emphasis shifts to better usage of bandwidth
  - Server can exchange more information with clients
- Streaming useful for scalability of data server
  - If the data size is large single-pass parsing is desired (e.g. display data)
  - Lower memory requirement for parse/generation of XML
- Gain from hardware-based network compression (e.g. MNP-5) can be significant
  - Dilutes need for binary XML representation

Parsing/Generation Speed

- Binary form parsing can be faster than text XML
  - Up to one order of magnitude faster
  - Saves power on small devices
- Binary XML parsers
  - Can as simple as text XML parsers
  - Can be more complex with over-engineering
- Parsing and generation costs strongly correlated
- Low parsing/generation cost needs simple binary form
  - Create map from element and attribute names to numbers
  - Pretty good compression for multiple occurrences of long names
  - Binary values encoded in binary stream (schema is known)
  - No need of entity resolution or white space normalization
- Parsing cost optimization may yield little compaction
  - Conflicts with optimizations for small footprint

Random Access

- Random access during forward-only parsing
  - Significant speedup in some scenarios (e.g. XPath evaluation)
  - Additional structures must be encoded
  - Increases generation time, slows down parsing of whole XML
- True random access (i.e. not forward-only parsing)
  - Increase in size of XML
  - Punishes modifications of larger XML
- How much to speed up random access?
  - Slows down parse/generation
  - Determined largely by workload

Data-only Compression

- Sender, receiver know strict XML schema
  - Only data needs to be encoded
  - Yields very good compression ratios
- Benefits are large for large amounts of data
  - Applications can build in data-only compression
  - WSDL, WAP binary XML protocol
  - Individual vendors can provide such solutions
  - Encoding is no longer self-describing
- Suitable for inter- and inter-process data exchange
  - Can achieve extensibility of component architecture
  - Change schema ⇒ different behavior

Application Needs

- Parsing/generation speed important for server
  - Web server/DB sends data out in chunks
  - Buffering data for large transfers degrades scalability
- Client applications may want
  - Faster parsing speed
    - Visual rendering
  - Low memory footprint
    - Cached data (user looks only at first result of search query)
  - Optimization criterion depends upon application
- Greater compression increases parse time
  - Beyond a certain point, the parsing/generation cost outweighs the benefits

Multiple Binary Formats

- Different optimizations benefit different applications
  - Server wants faster generation speed
  - Mid-tier server emphasizes portability of data
  - Client desires small memory footprint over slow connections
- All together — perf. benefits might disappear!
- Standard would have to allow multiple binary representations
  - Standard set of “encodings” allowed in binary representations
  - Each optimizes one or more facets and application classes
  - Format must handle all encodings of XML for I18N
- Each side receives and processes all binary encodings
  - Sender gets to choose format to generate
  - Receiver must decode multiple representations
  - Increased complexity of software development

Conclusions

- Is “binary XML” a good candidate for standardization?
  NO
- Criteria for “binary XML” are different & conflicting
  - Minimize footprint or minimize parse/generate time
  - No single criterion to optimize all applications
  - Binary standard must allow a suite of representations
  - Goes against grain of portability goals of XML 1.0
  - Depends on machine and OS architectures on each end — translating between binary representations negates advantages
- Requires hitting 80/20 point: Not good enough for many uses
- Standard’s work can go on for years …
  - … stifle innovation (Research first, standardize later)
  - … ensuing standard can be burdensome on vendors
- Need ideas to build on advantages of XML 1.0
  - Promising — interleaved text/binary format preserving Infoset
  - Blobs of data (e.g. pictures) sent as binary attachments
  - Portable, improves parsing speed sufficiently

Questions?