Syllable-based compression for XML

Katsiaryna Chernik,
Jan Lánský, Leo Galamboš

Dept. of Software Engineering
Faculty of Mathematics and Physics
Charles University
Content

- Motivation
- Syllable-based compression
- XMLSyl
- XMillSyl
- Results
- Conclusion
Motivation

- **XML**
  - Simple text format for structured text documents
  - Data exchange standard
  - High redundancy
Compression Methods for XML

- Character-based
  - XMill
  - XMLPPM
  - XGrind, ...
- Word-based ?
- Syllable-based ?
Syllable-based compression

- LZWL
  - Dictionary-based method
  - Syllable-based version of LZW
- HufSyl
  - Statistical method
  - Adaptive Huffman coding
  - Inspired by HuffWord
Syllable-based compression

Syllable-based compression is suitable for languages with rich morphology (Czech)

Syllable-based compression is suitable for small or middle-sized files
Syllable-based compression of XML

- Majority of XML documents are small or middle-sized
- Many text-like XML documents
  - news in RSS format
  - documentations or books in DocBook format

Syllable-based compression and XML?
XMLSyl

Idea

- Syllable-based compressor
  - XML tokens are divided to many syllables

- XMLSyl
  - XML tokens are treated as single syllables
XMLSyl
Architecture

XML Document → SAX Parser → Structure Encoder → Element Container, Attribute Container, Data and Structure Container → Syllable Compressor, Syllable Compressor → Compressed XML document
Example

XML doc:
<book>
  <title lang="en">XML</title>
</book>

SAX events:
startElement("book")
startElement("title",("lang","en"))
characters("XML")
endElement("title")
endElement("book")
XMLSyl
Example – Encoding process

SAX events:
startElement("book")
startElement("title,("lang","en")")
characters("XML")
endElement("title")
endElement("book")

<table>
<thead>
<tr>
<th>Element Container</th>
<th>Attribute Container</th>
</tr>
</thead>
<tbody>
<tr>
<td>book E0</td>
<td>lang A0</td>
</tr>
<tr>
<td>title E1</td>
<td></td>
</tr>
</tbody>
</table>

Data and Structure Container

```
E0  E1  A0  en  END_ATT
CHAR XML END_CHAR END_TAG END_TAG
```
XMLSyI
Implementation details

- SAX parser – EXPAT
- Syllable Compressor – LZWL and HufSyl
- Encoding was inspired by existing XML compression methods
  - XMLPPPM, XGrind, XPress, XMill
XMillSyl

- Based on XMill
- Main principles of XMill
  - Separating structure from data
  - Grouping Data values with related meaning
Architecture of XMill

Input XML file → SAX Parser → Path Processor

Structure Container → gzip
Data Container 1 → gzip
Data Container 2 → gzip
... → gzip

Compressed XML file
XMill – one container

Input XML file → SAX Parser → Path Processor → Structure Container, Data Container

Structure Container → gzip

Data Container → gzip

Compressed XML file
Syllable-based compression of XML
Experimental results

XMLSyl & XMillsyl vs. LZWL & HufSyl

- Non-textual XML data
  - 50-60% better
- Textual XML data
  - 10-20% better
Syllable-based compression of XML
Experimental results

Text-like XML documents

XMLSyl

XMillSyl one container

XMillSyl more containers
Syllable-based compression of XML
Experimental results

Text-like XML documents

XMLSyl
- XMLHuf is suitable for small-sized files
- XMLzwI is suitable for large-sized files

XMLSyl vs. XMill
- On average 10-15% worse than XMill
- On some documents the same performance or better
Conclusion

- New syllable-based compression methods of XML
  - XMLSyl (versions: XMLzwI, XMLhuf)
  - XMillSyl (versions: XMillzwI, XMillhuf)
- One of our method outperforms XMill on some documents
Conclusion

- Future work
  - extract and utilize the information in the DTD section
  - create a special syllable dictionary for elements and attributes
  - compress HTML data