University of Washington
Department of Computer Science & Engineering

Managing XML and Semistructured Data

Lecture 19: Compressing XML Data

Prof. Dan Suciu

Spring 2001

In this lecture

- XML Compression
  - Motivation
  - XMill approach and results

Resources

- XMILL: An Efficient Compressor for XML Data by Liefke and Suciu, in SIGMOD'2001

Compression: The Problem

- XML for exchange (space or time)
- but XML is verbose
- users prefer application specific formats:
  - Web Server Logs
  - EMBL
  - G2
- is XML doomed to fail ?

An Example:Web Server Logs

ASCII File 15.9 MB (gzipped 1.6MB):

```
202.239.238.16|GET / HTTP/1.0|text/html|200|1997/10/01-00:00:02|-|4478|-|-|http://ww
```

XML-ized inflates to 24.2 MB (gzipped 2.1MB):

```
<apache:entry>
  <apache:host> 202.239.238.16 </apache:host>
  <apache:requestLine> GET / HTTP/1.0 </apache:requestLine>
  <apache:contentType> text/html </apache:contentType>
  <apache:statusCode> 200</apache:statusCode>
  <apache:date> 1997/10/01-00:00:02</apache:date>
  <apache:byteCount> 4478</apache:byteCount>
  <apache:referer> http://www.net.jp/ </apache:referer>
  <apache:userAgent> Mozilla/3.1${ja$}$(I)</apache:userAgent>
</apache:entry>
```

XMill

- specialized compressor for XML data
- makes XML look "small"
- Download:
  - Now: www.research.att.com/sw/tools/xmill
  - Soon: www.cs.washington.edu/homes/suciu/XMILL

How Xmill Works: Three Ideas

Compress the structure separately from the data:

gzip Structure

```
<apache:entry>
  <apache:host> </apache:host>
  . . .
</apache:entry>
```

+

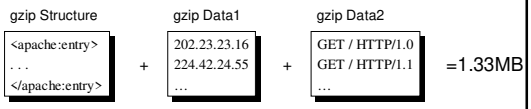
gzip Data

```
202.239.238.16
GET / HTTP/1.0
text/html
200
…
```

=1.75MB

How Xmill Works: Three Ideas

Group the data values according to their types:



How Xmill Works: Three Ideas

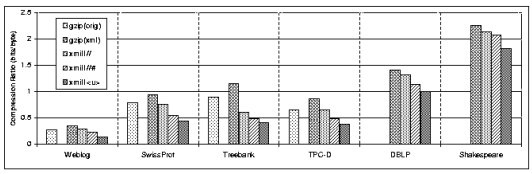
Apply semantic (specialized) compressors:

gzip Structure + gzip c1(Data1) + gzip c2(Data2) + ... =0.82MB

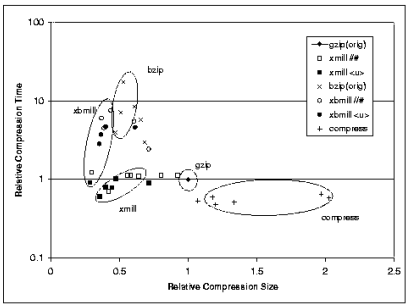
Examples:
- 8, 16, 32-bit integer encoding (signed/unsigned)
- differential compressing (e.g. 1999, 1995, 2001, 2000, 1995, ...)
- compress lists, records (e.g. 104.32.23.1 → 4 bytes)

Need user input to select the semantic compressor

XML Compression



Compression Tradeoff



Summary of XML Data Management

- XML =
  - old data type (trees)
  - with new interpretation (data)
- We discussed traditional management techniques for XML:
  - Data model
  - Query language
  - Optimizations
  - ...
- Many traditional problems still unsolved (storage, processing, optimization, ...)

Summary of XML Data Management

- More interesting question:
  - what are the novel applications enabled by XML ?

Some ideas:
- Approximate queries over unfamiliar data instances
  - "Search the database for a pattern *similar* to this one"
  - Rank results based on their similarity to the pattern
  - What is an appropriate query language for that ?
- Linking independent databases
  - We have Xlink, how do we use it ?