Type-based compression of XML streams

by Christopher League — Assistant Professor
and Kenjone Eng — B.S. 2004, M.S. 2006
Computer Science Department
Long Island University

eXtensible Markup Language represents structured data.

- The <tags> help humans and computer programs interpret the data properly.
- Used in every field, and as a file format for some applications (*e.g.*, MS Office 12).
- Ideal for transferring data between different database systems.

Extensibility comes at a cost.

- The tags substantially increase the size of the stream; therefore we will consume:
- More network bandwidth to transmit it,
- More disk space to store it, and
- More time to parse it.

A schema specifies what tags to use, and how to combine them.

- A program can automatically *validate* XML documents against the schema.
- Several schema formats are available; we will use "RELAX NG" [Clark & Murata]

```
start = element search-results { item* }
item = book | audio-cd
book = element book
    {id, title, author+, cover?, price}
audio-cd = element audio-cd
    {id, title, artist, cover?, price}
id = attribute id { text }
title = element title { text }
author = element author { text }
...(etc.)
```

Figure 2: a RELAX NG schema for a set of product search results, as in figure 1.

a?	element a is optional
b*	b appears zero or more times
c+	c appears one or more times
d | e	choose d or e (not both)

```
<search-results>
 <book id="067003472X">
  <title>Breaking the Spell</title>
  <author>Daniel C. Dennett</author>
  <cover>/9j/4AAQSkZJRgABAQAA..</cover>
  <price>$16.35</price>
 </book>
 <book id="0553345842">
  <title>The Mind's I</title>
  <author>Douglas Hofstadter</author>
  <author>Daniel C. Dennett</author>
  <price>$12.32</price>
 </book>
 <audio-cd id="B0000808B3">
  <title>Bach: The Art of Fugue</title>
  <artist>Emerson String Quartet</artist>
  <cover>FhcYGRolJicoKSo0NTY3..</cover>
  <price>$14.98</price>
 </audio-cd>
</search-results>
```

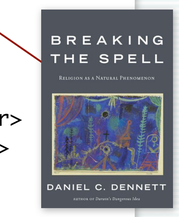


Figure 1: a sample XML stream, representing a product search result.

Using a schema, we can vastly abbreviate the document.

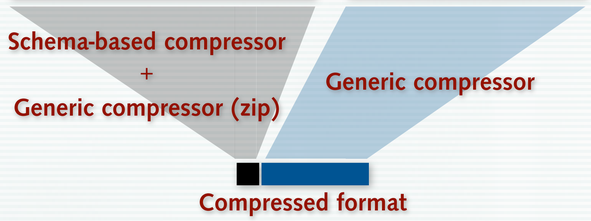
- The sample schema (figure 2) says that a book *must* have an ID attribute and a title,
- So we can *omit* both of those tags and just send the data itself.
- Cover art is optional, so just store *one bit* (true or false) to say whether it is present.

First, separate tags from data and encode each separately.

- Tags will be compressed using knowledge from schema.

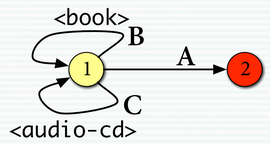
```
<search-results><book id="•">
<title>•</title><author>•
</author><cover>•</cover>
<price>•</price></book>  ...
```

```
067003472X
Breaking the Spell
Daniel C. Dennett
/9j/4AAQSkZJRgABAQAA…
$16.35                    ...
```

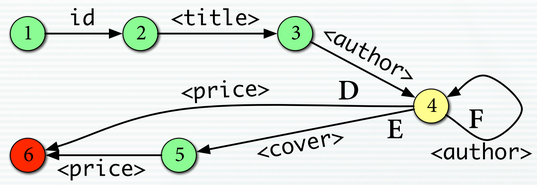


Schema can be systematically converted to state diagrams.

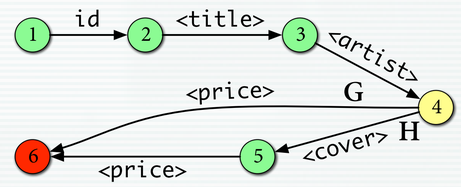
- There are three different kinds of states:
  - No choices, ∴ nothing to record.
  - Choice point, record path taken.
  - Final state, end of this element.
- State machine for <search-results>:



- For the <book> element:



- And the <audio-cd> element:



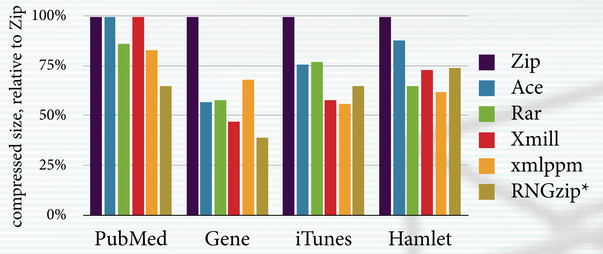
Path through state diagram is sufficient to infer tag structure.

- "B E B F D C H A" is all we need to record; from this we can reconstruct all the tags:

```
<search-results>
 [B]<book id="•"> <title>•</title> <author>•</author>
    [E]<cover>•</cover> <price>•</price> </book>
 [B]<book id="•"> <title>•</title> <author>•</author>
    [F]<author>•</author> [D]<price>•</price> </book>
 [C]<audio-cd id="•"> <title>•</title> <artist>•</artist>
    [H]<cover>•</cover> <price>•</price> </audio-cd>
[A]</search-results>
```

Preliminary experimental results

- We implemented this algorithm in Java, using the *Bali* library [Kawaguchi].
- Four examples were tested: bibliographic & genome data from NIH, meta-data from a music library, and a Shakespeare play.
- Compressed sizes are shown, so **smaller bars are better.**



- Zip is the popular *deflate* algorithm—all sizes are relative to this, so it is 100%.
- Ace & Rar are newer general-purpose compression algorithms.
- Xmill & xmlppm are specialized for compressing XML documents.
- **RNGzip is our type-based compressor.**

Conclusion

- Our technique really shines on the NIH examples—the ratio of tags to text is high.
- We may be able to *combine* our strengths with those of xmlppm, to do even better.

Selected references

- J. Cheney. *Compressing XML with multiplexed hierarchical PPM models. Data Compression Conference,* p. 163–172. IEEE, 2001.
- J. Clark & M. Murata. *RELAX NG specification.* Dec. 2001
- K. Kawaguchi. *Bali—RELAX NG validatelet compiler.* Sep. 2002.
- H. Liefke & D. Suciu. *Xmill: an efficient compressor for XML data. Proc. Conf. on Management of Data,* pp. 153–164. ACM, 2000.