XML Schema 1.0 and Relax NG Only Partially Support Context-Free Grammars

Roger L. Costello
January 1, 2014

Neither XML Schema 1.0 nor Relax NG are able to express this:

> *The content of BookStore shall be N Book elements
> followed by N Magazine elements.*

That is, BookStore must contain an equal number of Book and Magazine elements.

So this XML document:

```
<BookStore>
   <Book>...</Book>
   <Book>...</Book>
   <Book>...</Book>
   <Magazine>...</Magazine>
   <Magazine>...</Magazine>
</BookStore>
```

will not generate an error when validated, even though it is invalid (it is invalid because there are 3 Books and only 2 Magazines).

On the other hand, XML Schema 1.0 and Relax NG can express this:

> *The content of BookStore shall be N Book elements
> followed by N Magazine elements, provided each
> Book-Magazine pair is wrapped within a Merchandise
> element.*

Here is a sample XML document:

```
<BookStore>
   <Merchandise>
      <Book>...</Book>
      <Merchandise>
         <Book>...</Book>
         <Magazine>...</Magazine>
```

```
    </Merchandise>
    <Magazine>...</Magazine>
  </Merchandise>
</BookStore>
```

Notice that there are N Books followed by N Magazines and each Book-Magazine pair is wrapped within a Merchandise element.

Why does XML Schema 1.0 and Relax NG support the latter but not the former?

I have heard people say that the former is a context-free (CF) grammar and neither XML Schema 1.0 nor Relax NG support CF grammars. People say that XML Schema 1.0 and Relax NG can only express regular grammars. (Recall that context-free grammars are more powerful than regular grammars.)

It is certainly true that the former is a CF grammar, since abstractly it is simply an instance of a^nb^n, which is the classic example of a CF language.

But hold on!

The latter is also a CF language, as I informally prove in the following section.

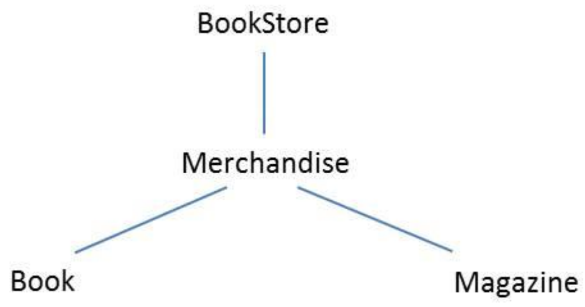
So, XML Schema 1.0 and Relax NG support *some* CF languages but not others.

It seems to me that XML Schema 1.0 and Relax NG should either support CF grammars or not support CF grammars. The existing partial support of CF grammars is just plain strange.

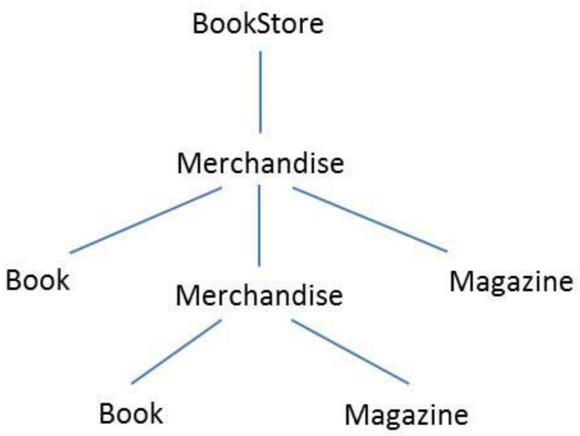
Recommendation: Extend XML Schema 1.0 and Relax NG to provide full support for CF grammars.

Proof That The Merchandise XML Language Is a CF Grammar

The field of formal language theory tells us how to prove that a language is context-free. There is a famous theorem, the *uvwxy* theorem (also called the *pumping lemma*), that characterizes CF grammars. Essentially it says that a CF grammar is one that generates XML instances containing parts which can be expanded by any amount. Some examples will help to explain what that means. Here is a valid XML instance (in tree form):



The Merchandise element contains a Book element followed by a Magazine element. But there is an optional Merchandise element which can be inserted in-between:



The Merchandise element at the lowest level of the tree contains a Book element followed by a Magazine element. Again we can insert an optional Merchandise element.

So we can expand (pump up) that Merchandise element as much as we please.

Therefore, the XML language is a CF grammar.