Aw, SCHUCS! An Approach to Creating XML Schemas from UML Class Diagrams

Michael Hucka
mhucka@caltech.edu
Control and Dynamical Systems 107-81
California Institute of Technology, Pasadena, CA 91125

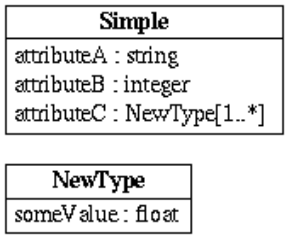
30 October 2000

XML Schema (Fallside, 2000) is likely to become the standard for defining structure, content and semantics of XML documents. An XML Schema specifies the kinds of objects allowed in an XML data stream, as well as how the objects and their properties are to be organized and the types of values that can be assigned to the object attributes. However, while it is powerful and flexible, the XML Schema language (and for that matter, the DTD language) is complicated and not suitable as a descriptive specification language for humans communicating about data structures. A more suitable language is UML, the Unified Modeling Language (Eriksson, 1998; Oestereich, 1999). Unfortunately, there currently does not exist an agreed-upon approach for translating object classes represented with UML class diagrams into XML Schemas. Here I describe one approach, SCHUCS ("Schemas from UML classes"), which aims to provide a simple, direct and systematic methodology for taking data classes defined in UML and translating them into XML Schemas.

There are three main advantages to using UML class diagrams as a basis for defining data structures in bioinformatics. First, compared to using other notations or a programming language, the UML visual representations are generally easier to read and understand by readers who are not computer scientists. Second, the visual notation is implementation-neutral—the defined structures can be encoded in any concrete implementation language, not just XML but other formats as well, making the UML-based definitions more useful and flexible as a communication medium. Third, UML is a de facto industry standard, documented in many books and available in many software tools including mainstream development environments. Readers are therefore more likely to be familiar with it than other notations.

SCHUCS has grown out of ongoing work on a software framework for simulations in systems biology; the framework uses XML-based data structures (Finney et al., 2000) and is being designed to interact with bioinformatics databases. SCHUCS maps a subset of UML class diagrams to XML Schemas. Here is simple example of its application, showing first a UML diagram, followed by the resulting XML Schema, followed by an example XML document fragment:

UML:



XML Schema:

```
<xsd:complexType name="NewType">
  <xsd:attribute name="someValue" type="xsd:float"/>
</xsd:complexType>
<xsd:complexType name="Simple">
  <xsd:attribute name="attributeA" type="xsd:string"/>
  <xsd:attribute name="attributeB" type="xsd:integer"/>
  <xsd:element   name="listOfAttributeCs">
    <xsd:complexType>
      <xsd:element name="attributeC" type="NewType"
                   minOccurs="1" maxOccurs="unbounded"/>
    </xsd:complexType>
  </xsd:element>
</xsd:complexType>
```

Example XML:

```
<simple attributeA="foo" attributeB="42">
  <listOfAttributeCs>
    <attributeC someValue="1.0"/>
    <attributeC someValue="2.0"/>
    <attributeC someValue="3.0"/>
  </listOfAttributeCs>
</simple>
```

Compared with other related efforts at mapping UML to XML, SCHUCS has the following features. First, it maps UML object classes directly to XML Schemas definitions, such that the Schemas can be used to represent data objects. Some other efforts such as XMI (OMG, 1998) are aimed at representing the *models* implied by UML, not the data structures. Second, SCHUCS does not define auxiliary XML constructs. Some other approaches such as the *UML for XML Schema Mapping Specification* (Booch et al., 1999) require the use of certain constructs such as UML stereotypes that complicate the data structure definitions. Third, SCHUCS produces compact XML representations that emphasize the use of attributes over elements; this leads to small XML data objects/documents, an advantage when exchanging large numbers of messages between databases, search engines, and other software.

The approach does have limitations. First, the scope of the notation is limited to classes and their attributes, not class methods or operations. This is appropriate in the context of databases and data objects, but may not be suitable for other applications. Second, the translation approach is not bidirectional. It is only designed to handle the direction of UML to XML, not the reverse. And third, the method only handles a subset of UML class diagram constructs. The limited scope is consistent with the goal of maintaining a reasonably simple notation and UML-to-XML mapping; SCHUCS is explicitly *not* intended to cover the full power of UML or XML.

SCHUCS is currently being used in two efforts, the Modeler's Workspace (Hucka et al., 2000) and the Systems Biology Workbench (Finney et al., 2000). However, it is a general-purpose methodology that can be applied to other areas as well. It could be especially useful for representing schemata in bioinformatics, where UML-based diagrams of data structures would serve as a better communication tool than RDF, straight XML Schemas, and other similar notations.

References

Booch, G., Christerson, M., Fuchs, M., and Koistinen, J. (1999). UML for XML Schema Mapping Specification. Available via the World Wide Web at `http://www.commerceone.com/xml/sox/index.html`.

Eriksson, H.-E., and Penker, M. (1998). UML Toolkit. New York: John Wiley & Sons.

Fallside, D. C. (2000). XML Schema Part 0: Primer (W3C Working Draft, 7 April 2000). Available via the World Wide Web at `http://www.w3.org/TR/xmlschema-0/`.

Finney, A., Sauro, H., Hucka, M., and Bolouri, H. (2000). An XML-Based Model Description Language for Systems Biology Simulations.. Available via the World Wide Web at `http://www.cds.caltech.edu/erato`.

Hucka, M., Beeman, D., Shankar, K., Emardson, S., and Bower, J.M. (2000). The Modeler's Workspace: A Tool for Computational Neuroscientists. Soc. Neurosci. Abstr. 26.

Oestereich, B. (1999). Developing Software with UML: Object-Oriented Analysis and Design in Practice. Addison-Wesley.

OMG. (1998). XML Metadata Interchange (XMI). Available via the World Wide Web at `ftp://ftp.omg.org/pub/docs/ad/98-10-17.pdf`.