ModelicaXML

A Modelica XML representation with Applications

Adrian Pop, Peter Fritzson
Programming Environments Laboratory
Linköping University
Outline

- Introduction
- Why XML?
- XML, DTD
- ModelicaXML
- ModelicaXML Applications
- Problems with XML
- Semantic Web Languages
- Conclusions and Future Work
Modelica has a fast growing code base
Modelica adopters have to develop their own tools for tasks like:
- analysis of models (checkers and validators)
- pretty printing (un-parsing)
- interchange with other modeling languages
- query and transformation of models
- impose code style guidelines
- documentation

All these utilities
- need easy access to the structure of the language
- should interoperate.
Store the structure of the Modelica code using an alternative representation

Perform the tasks on this alternative representation

The alternative representation should
- be easy accessible from any programming language
- be easy to transform, query and manipulate
- support validation

XML has all these properties
Extensible Markup Language (XML)

- World Wide Web Consortium (W3C) standard
- XML represents information as trees using tags

```xml
<?xml version="1.0"?>
<!DOCTYPE persons SYSTEM "persons.dtd">
<persons>
  <person job="programmer">
    <name>John Doe</name>
    <email>email@none.ro</email>
  </person>
  ...
  <person job="manager">
    <comment>Classified</comment>
  </person>
</persons>
```
- DTDs specify the allowed content of the XML elements
- Are used for XML document validation

```xml
<!ENTITY % person-job-attribute "job (programmer|manager) #REQUIRED">
<!ELEMENT persons (person*)>
<!ELEMENT person((name+, email*)|comment+)>
<!ATTLIST person project CDATA #IMPLIED &person-job-attribute;>
<!ELEMENT name(#PCDATA)>
<!ELEMENT email(#PCDATA)>
<!ELEMENT comment (#PCDATA)>
```
`class Test "comment"`  
  `Real x;`  
  `Real xdot;`  
  `equation`  
  `xdot = der(x);`  
  `end Test;`  

```xml
<modelicaxml>
  <definition ident="Test"
    comment="comment">
    <component ident="x" type="Real"
      visibility="public" />
    <component ident="xdot" type="Real"
      visibility="public" />
    <equation>...</equation>
  </definition>
</modelicaxml>
```
ModelicaXML Applications

Interoperability and Transformation

Modelica code

Modelica Parser

Modelica XML

XML Tools

validate, read

validate, read

output

output

output

HTML Documentation

read

output

Parsers

Other XML formats

Other modeling languages

validate, read

output

read

output

Other XML formats

parse
The Stylesheet Language for Transformation (XSLT)
- is a declarative language for transforming XML trees
- selects parts of the XML tree using the XPath language
- uses templates to transform the selected parts
- outputs desired format
The Query Language for XML (XQuery)
- is a declarative language for query of XML trees
- selects parts of the XML tree using the XPath language
- uses SQL-like query language to further manipulate the results
Document Object Model (DOM)

- is a platform and language neutral interface for
  - access and update the content/structure/style of XML trees in XML documents
- implementation in almost all programming languages
Problems with the XML Representation

- XML can only express syntax
- No easy way to automatically handle semantics
- Possible solutions
  - use more expressive markup languages to express Modelica semantics
  - use available tools for these languages
Resource Description Framework (RDF)
- represents graphs
- describes classes and relations
- out of the box tools for graph query
- some type checking is available

Web Ontology Language (OWL)
- adds a Description Logics layer on top of RDF
- out of the box tools for:
  - type checking
  - model consistency
  - making the implicit sub-typing and equivalence relations explicit (inference)
Conclusions and Future Work

- **ModelicaXML**
  - offers *easy access* to Modelica language structure
  - helps in providing *more non-simulation applications* for Modelica
  - is *not enough* for handling Modelica semantics

- **Future Work**
  - formal definition of all the intermediate steps from Modelica to flat Modelica
    - (static semantics, interoperability)
      - tools will act on these level independent of each other
  - further investigation of the use of the Semantic Web Languages to express Modelica semantics is needed
Thank you!
Questions?
ModelicaXML and ModelicaOWL

- **ModelicaXML**
  - [http://www.ida.liu.se/~adrpo/modelica/xml](http://www.ida.liu.se/~adrpo/modelica/xml)

- **ModelicaOWL**
  - [http://www.ida.liu.se/~adrpo/modelica/owl](http://www.ida.liu.se/~adrpo/modelica/owl)