Formal Mathematics in MathML

Andrea Asperti, Luca Padovani, Claudio Sacerdoti Coen, Irene Schena
Department of Computer Science
Via di Mura Anteo Zamboni 7, 40127 Bologna, ITALY.
contact: asperti@cs.unibo.it

According to Proof Theory, formal derivations are just yet another category of mathematical expressions. This simple remark opens the possibility to exploit MathML not only for logical statements or traditional mathematical expressions, but also for the more ambitious goal of encoding proofs, and then, by suitable extensions, whole mathematical documents. On the other side, entering in foundational issues, we must be very carefull to respect the multilingal environment of the possible, different formalization of Mathematics: each foundational system requires its own logical dialect, with a specific markup. From this respect, XML provides the right flexibility. Then MathML, and in particular its content markup, can be profitably used as an interesting intermediate, language between the low-level formal encoding (expressed in some specific XML dialect) and the target formatting language. At this intermediate level we could already get rid of a lot of annoying syntactical details that typically afflict formal mathematics, making it too constricting and too verbose for humans. In any case, formal content would be preserved by means of backward pointers to the low level specification.

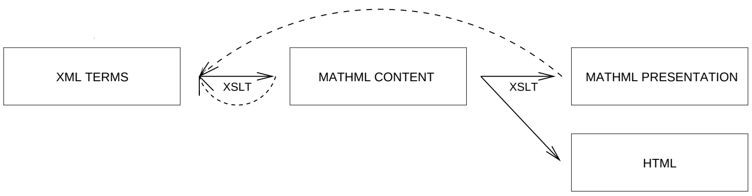

Figure 1: Transformations on XML files: the backward arrows represent links from the content and presentation files to the corresponding XML files.

This use of MathML as an (intermediate) archival and presentation format for large repositories of structured mathematical knowledge, opens new and exciting possibilities, and enlarges its field of application to the Proof Assistant and Logical Environments communities. As a matter of fact, existing logical systems are not suitable for the creation of large repositories of structured mathematical knowledge accessible via Web. In fact, libraries in logical frameworks are usually saved into representations which are clearly unsatisfactory, since they are too oriented to the specific applications. The information is not directly available, if not by means of the functionalities offered by the system itself, and moreover the same information lacks of a convincing form of presentation. On the contrary, XML is a natural, application-independent specification format for formal libraries, providing a central technology for storing, retrieving and processing mathematical documents, comprising sophisticated web-publishing mechanisms (stylesheets). In the same way, MathML can naturally be exploited as not only an interesting (intermediate) archival format, but also as the privileged target formatting language, yielding a natural bridge between tools for the mechanization of formal reasoning and the Web.

The choice of MathML is justified by several reasons:

- Even if content markup is restricted to the encoding of a particular set of formulas (the ones used until the first two years of college in the United States), it has been conceived as an extensible (and thus, potentially, infinitary) language.
- Passing through an intermediate semi-formal representation (such as MathML content) improves the modularity of the overall architecture: many specific logical dialects can be mapped into the same intermediate language (or into suitable extensions of it), and similarly the same intermediate representation can be transformed into different presentation formats.
- Being a standard, MathML may be exploited to cut and paste expressions from an application to another.
- MathML content captures well the *informal* "semantics" (or, better, the semi-formal *usage*) of well-known operators, such as, for instance, equality. At foundational level, there are a lot of different equalities (intensional, extensional, Leibniz higher-order equality, Type-specific equalities, ...), but there is little point to preserve the distinction at the intermediate level, if not as a pointer to the specific definition.

Once XML is chosen as the data description language for representing mathematical terms, XSL Transformations (stylesheets) provide a standard converter mechanism to automatically generate the associated MathML representations from a XML mathematical document, yelding a modular and application independent mechanism for associating notation to formal content. As depicted in Figure 1, there are two phases of stylesheets application: the first generates the MathML content representation from the XML one; the second generates from this intermediate representation the MathML presentation markup (or a different target markup).

The transformation from the low-level formal XML format to MathML content typically requires the extension of the latter language to cover, by means of suitable `csymbols`, the primitive constructors of the specific logical dialect which, in good foundational systems, are few. The power of MathML content starts to be fully exploited in the encoding of mathematical entities defined in the Logical System, and especially when they belong to the basic set of MathML content elements. In this case, the low level identifier can be directly mapped to the corresponding operator, automatically recovering its intended presentation. A pointer to the formal, low-level definition of the operator can be kept in the `definitionURL` attribute.

At presentation level, we have just to extend the (already available) MathML stylesheets to cover the new `csymbols` introduced at content level. Again, we keep pointers to the associated low-level formal information, in the `xlink:href` attribute, using the machinery of XLink.

We have tested the previous methodology on the standard library of COQ (a well known Proof Assistant developed at INRIA-Rocquencourt). The library may be consulted at the following url: `http://phd.cs.unibo.it/helm/coq`. For more informations on our project see `http://www.cs.unibo.it/~asperti/HELM/home.html`.