First International Congress on Tools for Teaching Logic Technical aspects of MetaBook: construction and use

Alberto Policriti
policriti@dimi.uniud.it
Dipartimento di Matematica e Informatica
Università di Udine, 33100, via Delle Scienze 206
Italy

May 24, 2000

Introduction

This document is a very short outline of the main technical (architectural, structural, functional, etc.) aspects of a tool that has been designed as part of the EC alpha project ARACNE: tools for teaching coordinated by Maria Manzano currently at the University of Salamanca.

One of the main goals (and challenge) of the alpha projects was to provide means to communicate and manage the ideas and instruments relative to a discipline (Logic in our case) in order to facilitate their circulation between Europe and Latin America. It was clear since the beginning that the challenge was not trivial and that many issues were involved. Here are some of those issues, the ones more related with the meta-book construction:

1. On what kind of tools should we concentrate? Should we simply talk about the ideas that could be exported/imported or should we get involved at a more practical level?

2. Who was the prospective user of our output?

3. To what extent should we get involved in the production/development of something sufficiently flexible to adapt to future needs?

Obviously, the answers to many of our questions were in fact independent from the fact that we were working with Logic as underlying discipline. However, Logic was somehow one of the best possible disciplines in this respect, at least as far as the following points are concerned:

- Logic is a basic discipline, it is taught all over the world and is widely recognized as founding the cultural preparation of a large array of typologies of students (mathematicians, computer scientists, philosophers, etc.).

- Logic can be taught at many different levels and the didactic choices made to respond to necessities of different curricula are usually independent from cultural factors relative to the country in which one teaches.

In view of the above points the meta-book was one of the tools characterizing our proposal as alpha project and many efforts have been devoted to its construction.

What is a meta-book

It is to answer to the (many) different needs that were apparent from the beginning of the project that Maria Manzano proposed the construction of a meta-book: a book on books (and possibly additional useful material) to help the teacher to design his/her Logic course. Such a meta-book should contain material and directions capable to guide/help both the teacher prepare his/her course and the student to make a better use of such course. Moreover, such a support should be provided in a flexible way: e.g. mathematicians should be assisted keeping in mind their specific needs, which are certainly different from the needs of computer scientists or from those of philosophers, etc.. Also, the meta-book should have a format allowing not only to convey rather general indications on the design of a curriculum but should be practically usable by the (Latin American/European) prospective teacher/student. New technologies should be taken into account in this respect and Internet should be exploited to favor the growth and maintenance of the meta-book with contributions (also in terms of software) from distant places.

The task of the Udine group was to design and build a program capable to answer to the above mentioned needs and what we will call MetaBook hereafter is the program that we produced. MetaBook is a program that works with a web interface and is essentially divided into two modules (MetaWriter and MetaReader that are briefly described below).

MetaWriter

The main functionality of MetaWriter is to support the writers working with MetaBook. The general idea is to provide to such writers the possibility to contribute to the construction of the meta-book in two ways:

1. by the introduction of parts, chapters, sections, etc. of the meta-book and maintaining an updated version of all the material;

2. by automatically producing an hypertext version of (one or more) selected paths of whatever available.

In other words the MetaWriter is the module in charge of supporting, organizing, and (to a certain extent) coordinating the activities of all the authors while working on all the parts that they are producing. This includes also lecture notes, exercises, and software, together with indications and experiences about their use and related with the project.

Two further activities of MetaWriter consist in maintaining a database of the writers—granting access and keeping backup copies of the files—and in producing the hypertext version of the meta-book available through the net.

MetaReader

The other module of MetaBook, MetaReader, is designed for the interaction with the non-writers users (that we will call the readers) of the tool through the web. It provides functionality very similar to those provided by the MetaWriter. As a matter of fact, as the reader can easily imagine, most of the functionality of MetaReader is a slight modification of that already provided by MetaWriter. The reader accessing MetaBook must be able to design, starting from the available material, his/her own Logic book (eventually in one of the available languages) with lecture notes, exercises, software, etc.. Moreover, the writers of the meta-book should provide predefined paths along the meta-book easily accessible by the reader through the MetaReader module and allowing to quickly design versions of the meta-book for the most common typologies of users.

The choices (the typesetting system)

Many choices have been made during the design and the implementation of meta-book (a prototype version of the MetaWriter is on-line while the MetaReader is still almost entirely under construction). We do not have neither the space nor the time to justify all of them but we comment on one that we consider as the fundamental one: the choice of the typesetting system that we assume every writer were using.

We have assumed that all the writers would use LaTeX as typesetting system. The principal motivations for this choice are the following:

- LaTeX is an exceptional typesetting system for mathematics oriented documents.
- the LaTeX ideas about how to build documents helped us in organizing the program, naturally providing a subdivision of the entire document that could be easily exploited by the part of MetaBook in charge of reconstructing selected paths among the whole bulk of the material.
- The format of LaTeX documents forces a natural and simple discipline in the definition and use of macros, special symbols, cross-references, etc..

Below we describe the general structure of the document as it is shaped by MetaBook and this should further clarify our above points.

The general structure

MetaBook (mostly MetaWriter) produces and deals with essentially three kind of objects:

1. the LaTeX files of the meta-book: what we collectively call the document,
2. the hypertext version the document: one (big) html file,
3. the tree structure of the document: again an html file.

The document is our source: is updated by MetaWriter whenever a new (or a new translation of a) part, chapter, section, etc. is written and downloaded by one of the writers through the web, MetaWriter updates the database of the document'files, updates the hypertext version of the meta-book, and generates backup copies.

The hypertext version is generated via a call to the `LaTeX2HTML` utility developed by Nikos Drakos at the University of Leeds.

The tree structure is an hypertext generated from the document to simplify the task of referencing material in other parts of the meta-book and extensively using the `label` and `ref` LaTeX-instruction to this end. The construction of the tree structure is necessary in order to allow to the different writers to exchange information on the choice (in fact made by MetaWriter) of labels used throughout the document.

The hypertext version

As we said, one of the activities performed by MetaBook is the production of the hypertext version of the material available. This choice was motivated by the fact we choose a non-WYSIWYG (What You See Is What You Get) editor. This does not allow to all the writers to have a clear view of what has been produced so far and the on-line hypertext version was the simplest way to provide such a complete view of the whole book to both the writers and the readers. Moreover, it is also the simplest way to support many of the features that we wanted to provide through MetaReader.

Future work

In a project like this one to say what still has to be done is at least as easy as to say what has been done. Many ideas and possibilities came up during the construction of the program and in the following we just briefly say a few words on the most important topics that would need further attention.

Translations

An extremely important aspect of the meta-book is the fact that should support users not only having different background and needs but also speaking different languages. This is particularly important if we expect MetaBook to be used by writers and readers interested in designing versions of the meta-book suitable for introductory courses in Logic.

Versions in different languages of parts, chapters, sections, etc. should be easily reachable from the main corpus of the meta-book. Moreover, predefined paths relative to translation in different languages should be provided and easily obtainable.

One further point on this topic is the fact that the module relative to the management of versions in different languages should be designed in such a way that even non-specialist writers (working on the translation only) should be put in condition to work under the supervision of some specialist checking the material before the introduction within the meta-book.

Bibliography

Another aspect that should be treated with special care is a more sophisticated treatment of bibliographic references. A good meta-book (whatever is the discipline for which it has been designed) should embody also a complete and well organized database of bibliographic references. To this end we plan to use extensively the BibTeX utility and to provide—besides the modules relative to the management of introduction, deletion, update, etc. of bibliographic references—also modules for the production of predefined and reasoned collections of references.

Parallelism and security

It is possible that many authors decide to work at the same time using MetaWriter: we should be able to allow some form of parallelism of activities. As the program is designed now, only one author can work at any given time, even though clearly there are many things that could be done at the same time. At this stage of the project we thought that the amount of work necessary to guarantee some sort of consistency while allowing parallel activities was not worth the effort.

Moreover, we did not put up any fancy security barrier: we only used a database of the author together with a password system for granting entrance to MetaBook.

Exporting the idea to other contexts

To conclude, let us briefly comment on the possibility of exporting the whole package to other application contexts.

We are convinced that, even though Logic is probably one of the most interesting discipline in connection to which MetaBook works, almost any other scientific subject could make use of our tool. Designing a tool sufficiently flexible to allow its re-use in other fields was not our initial objective. Hence, we did not think of any "initialization" process that could be added to MetaBook in order to set parameters that could take different values depending on the prospective use. Among such parameters we could have:

1. (significant) different sizes for the group of authors;

2. different levels of security;

3. multiple sites acting as servers (for both running MetaBook and producing hypertext versions).

In general, we are convinced that the meta-book idea (and large parts of MetaBook) could be "recycled" in much more practical environments in which a large manual configurable according to somehow different needs is to be constructed by people at different sites.