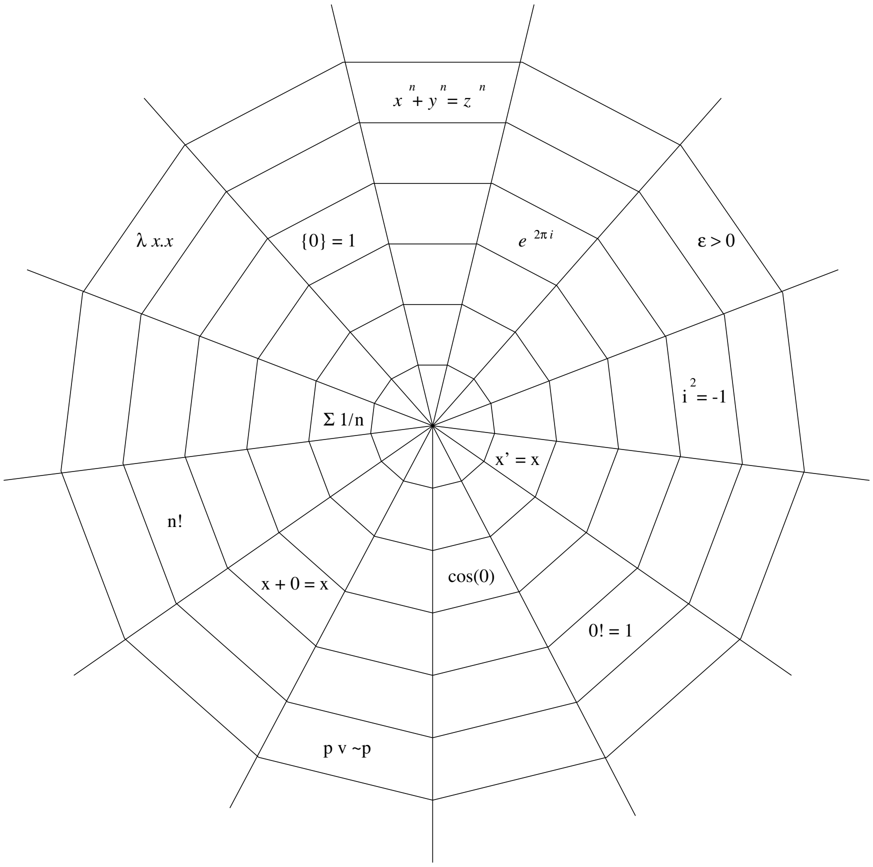
Publishing Formal Mathematics on the Web



Pavel Naumov

June 9, 1998

Publishing Formal Mathematics on the Web

Pavel Naumov
Computer Science Department
Cornell University, Ithaca, NY 14850
pavel@cs.cornell.edu

Abstract

Paper describes the design of the Nuprl Web Publisher - an automated tool for converting formal, computer-generated, mathematical texts into a set of hyper-linked HTML pages that preserves original, non-linear, text structure.

The current version of the Web Publisher, also developed by the author, provides access to term structure of the displayed formulas and links to definitions of abstractions used in these formulas.

1 Non-linear structured documents

For thousands years people have been writing texts in consecutive order: sentence was followed by sentence, paragraph by paragraph, chapter by chapter. Although this way to present material may be the best for a novel or a history book, where text has natural chronological structure, it is less acceptable for more technical writing, where non-linear structure of the document is more appropriate. Such structure may provide access to more detailed information or include reference to another part of the document. Different techniques of presenting texts with non-linear structure have been developed: page reference, index, table of contents, footnote, figure reference, list of figures, formula reference, etc. All this is an attempt to adopt sequential paper-based medium to handle non-linear texts and as a result they provide only partial solutions to the problem. In real life nobody is satisfied with the fact that after so much time has been spent on search for a word in the index, and the index itself, one still needs to look through the book to find page with the definition and look once again through the page to find needed sentence. Ideally, each word in the book should have be connected to at least two different objects: the next word and the definition of the word in this book or some standard dictionary. Similarly should be handled the other existing usage of cross-referencing.

Some of the most interesting objects for testing ideas about non-linear text structure are computer-generated texts with formal mathematics because

1. Mathematical texts have one of the most complicated structure. Traditional math texts usually are full of cross-references and are normally

accompanied by an extensive index.

2. Adding even simple non-linear structure to formal mathematics representation will make big difference since formal mathematics are among the hardest texts to read.

3. Since formal mathematical texts are normally created based on some form of computer representation of the definitions, theorems, and proofs, structure of the text can be provided by the computer. There is no need to create such structure or to reconstruct it from a linearly-organized document.

The Nuprl Proof Development Environment was one of the first computer system to use extensively non-linear structured texts in formal mathematics([1], [2], [3]). Supplemental information about used in the text mathematical objects, such as abstraction definitions, can be accessed in Nuprl by selection appropriate text fragment and clicking it. Unfortunately, these features were available only to users of the system. Once a formal theory has been developed and published by Nuprl as a LaTeX document, the only non-linear structure that was left was the tree-like presentation of the proof for which a special style had been developed.

Another approach to publishing non-linearly structured texts has became possible only in the last decade of the twentieth century when a new, non-linear, medium has been developed and has been made accessible to substantial number of people. This medium is the World Wide Web. The fact that some document is presented as a set of hyper-linked HTML pages does not mean yet that it is using all the possibilities, provided by the new medium. In fact, most of links on current HTML pages are generated manually, so they cannot use extensively this feature. Popular converters from old paper-oriented text formats such as LaTeX, $Word^{(\mathrm{TM})}$, or $WordPerfect^{(\mathrm{TM})}$ into HTML are mainly concerned with adequate representation on the Web of old documents with paper-oriented design. Probably it will take some time before new tools assisting authoring big HTML documents with complicated structure will be created.

This paper describes the design of the *Nuprl Web Publisher* - an automated tool for converting formal, computer generated, mathematical texts into a set of hyper-linked HTML pages that preserve original, non-linear, text structure[1].

2 Formula presentation in HTML

We have just seen that formal mathematical text is an appealing material for converting to HTML. At the same time, because such text usually consists almost entirely of formulas, its presentation on the Web has some special problems. In this section I will discuss these problems and possible approaches to their solutions.

[1]Examples of HTML pages produced by the converter can be found at Nuprl Project Web Page http://www.cs.cornell.edu/Info/Projects/NuPrl .

The most obvious problem are the special symbols used in mathematics. Such symbols are normally not included into fonts, available to a web browser. The most elegant solution is to use loadable fonts available in dynamic HTML. Any existing font can be converted into loadable font format and be used by the HTML document author. The browser supporting dynamic HTML will download fonts together with a web page and use them to display the page. Unfortunately, dynamic HTML is supported not on all platforms by $Netscape$ $Navigator^{(\mathrm{TM})}$, the major Web browser at this time. Hence, usage of loadable fonts will reduce the audience of potential readers[2] . Another solution, that is used in the `latex2html` converter as well as in Nuprl Web interface developed by R. Vaughn and D. Svitavsky ([4]), is to present mathematical formulas by graphical images. The biggest disadvantage of this method is that formula-intensive pages take too much time to download. Web Publisher implements an intermediate solution - regular font is used everywhere in a formula except for special symbols. Such symbols are displayed using tiny images. Since there is only a couple of dozens of such symbols in Nuprl formal mathematics, they are cached by the browser and re-used many times, radically cutting the downloading time.

Another obvious problem with presenting formulas on the Web is that formulas are not linear. In mathematical texts one can easily find formulas like

$$\frac{\partial^2}{\partial b\,\partial a}\int_a^b \begin{vmatrix} x & e^x \\ 1/x & \ln x \end{vmatrix} dx$$

New standard of MathML ([5]), that has been recently proposed, will provide mechanism to display such formulas in HTML. Fortunately for the Web Publisher, Nuprl formal mathematics does not have such formulas because of the limitations imposed by the system.

The other, less obvious, problem with presenting math formulas on the Web is: mathematicians do not always mean exactly what they write on the paper. Sometimes they do not display some parts of the formulas. One example of this would be the multiplication term $x \cdot y$ commonly written as xy. In such short form, there is just no space to click onto if we want to access extra information about operation "product". Another example, discussed in [2], of partial formula displaying is derivative f'_x that is commonly written as f' when variable x is clear from the context. Similarly, some arguments of the function are often called *parameters* and normally are not displayed when they have some "fixed" values. For instance, statement "X is an open set" actually has the topology $\mathcal{T}$ as its second and hidden argument, since the same set can be open with respect to one topology and can be not open with respect to another. Unfortunately, because in all of these cases there are some hidden formula parts, we are not able to provide hyper-links from these parts. In the Web Publisher this problem is solved by providing, when requested, two different representations of the formula: easy-to-read *display form* and more formal *term structure*.

[2] Future versions of Nuprl Web Publisher will be expected to use loadable fonts as soon as they become accessible on wider range of platforms.

THM	int upper properties	$\forall i{:}\mathbb{Z}.\ \forall j{:}\{i...\}.\ i \le j$
ABS	int_lower	$\{...i\} == \{j{:}\mathbb{Z}\mid j \le i\}$
ML	int lower ml inc	add_set_inclusion_info 'int_lower' 'int' AbSetDForInc Auto ;;
THM	int lower wf	$\forall i{:}\mathbb{Z}.\ \{...i\} \in \mathbb{U}_1$
THM	int lower properties	$\forall i{:}\mathbb{Z}.\ \forall j{:}\{...i\}.\ j \le i$
ABS	int_seg	$\{i..j^{-}\} == \{k{:}\mathbb{Z}\mid i \le k < j\}$
ML	int seg ml inc	add_set_inclusion_info 'int_seg' 'int' AbSetDForInc Auto ;;
THM	int seg wf	$\forall m,n{:}\mathbb{Z}.\ \{m..n^{-}\} \in \mathbb{U}_1$
THM	int seg properties	$\forall i,j{:}\mathbb{Z}.\ \forall y{:}\{i..j^{-}\}.\ i \le y < j$

Figure 1: Theory overview page fragment

Term structural representation is hyper-linked to additional information about formula parts.

3 Nuprl theory representation

All formal mathematics in Nuprl is divided for convenience into segments called *theories*. Each theory consists of definitions and theorems about some specific topic such as integer numbers, lists, booleans, etc. Although related theorems and definitions are normally placed in the same theory, still there exists inter-theory dependency. For example, rational number theory is using theorems and definitions from integer number theory. Hence, different theories cannot be published on the Web independently. Hyper-links should be provided from theory to all other theories on which it depends. It means that URL of such theories should be known at the time of converting the theory from Nuprl internal representation to HTML.

Components of a formal theory, such as definitions and theorems, are called *objects*. In addition to mentioned above *kinds* of objects, Nuprl theories have many other kinds of auxiliary objects to store information about tactics[3] enhancements that were made, abstraction display forms, etc. On the theory overview web page (Figure 1), objects are presented by rows in an HTML table. Each row corresponds to some object. Different columns are used to display kind, name, and content of the object.

To make this HTML table more readable, some objects are not represented in HTML. Among such objects are specifications of abstractions display forms and all objects, called *hidden*, that are not normally displayed by the Nuprl system. In Nuprl, list of hidden objects is determined by the author of the theory.

[3]Tactics in Nuprl are programs that consecutively apply different inference rules in order to proof a theorem. Tactics help reduce proof size and make proof more readable.

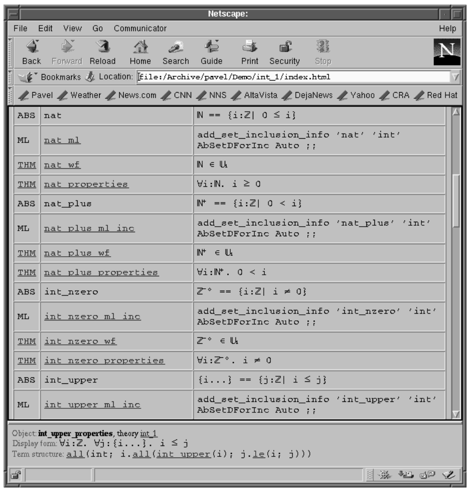


Figure 2: Netscape window

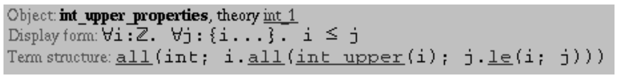


Figure 3: Theory help page content

When displaying HTML pages created by the Web Publisher, the browser window is always divided into two frames (Figure 2). One of these frames, called *main frame*, occupies most of the window space. It is used to display theory overview pages as well as theorem proof pages. Another frame, called *help frame*, is located on the bottom of the window. It provides, on-demand, extra information about content of the main frame, a click on the object name in the main frame results in displaying supplemental information about the object in the help frame (Figure 3). It consists of name of the theory to which object belongs, object content, already presented in the main frame, and, the most importantly, the structural representation of the object. Each abstraction name in structural representation, unless it is one of primitive abstractions, is hyperlinked to an object in a theory where this abstraction is defined. As a result, the reader can go through several help pages by clicking on abstraction names, until (s)he fully understands content of the main frame. During this review of the supplemental material the main frame will stay unchanged, so it is easy to switch back to the main "line" of the information at any moment.

4 Formal proof representation

Theory objects of the kind *theorem* also have proofs associated with them. Clicking on `THM` tag on the theory overview page results in *top proof page* (Figure

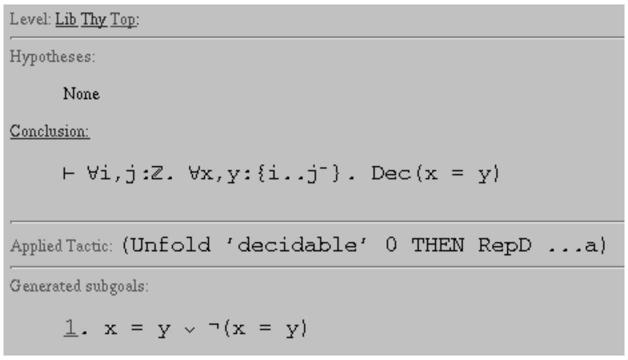


Figure 4: Top proof page content

4) being presented in the main frame. Formal proof representation is another example of extensive usage of non-linear structured texts. The top proof page displays the theorem statement, the tactic that was applied to it, and the subgoals to which theorem was reduced by the tactic.

Each subgoal in Nuprl consist of a hypothesis list and a conclusion. Experience shows that subgoal hypothesis list is usually slightly modified hypothesis list of the original goal. In order to keep HTML page short, Web Publisher displays subgoal in *suppressed* form that includes only subgoal conclusion. The subgoal hypotheses are presented on the *subgoal proof page* that will be discussed later.

In many cases information provided on the top proof page is sufficient for understanding of the proof. In such situations the reader will just go back to the theory overview page to study other theory objects. If extra information is required for better proof understanding, it can be accessed from the top proof page on as-needed basis. All hypotheses and the conclusion on the top proof page are clickable. They are hyper-linked to structural formula representation that will be displayed in the help frame. Just like in the case of a theory objects, any abstraction in such structural representation is linked to the object with its definition.

If the proof of any of the subgoals to which original theorem was reduced by the tactic is not obvious, additional information about proof of this subgoal may be accessed by clicking on the subgoal. *Subgoal proof page* will be displayed in the main frame and will be organized identically to the top proof page: statement of the subgoal, tactic that was applied and the sub-subgoals to which subgoal was reduced by the tactic. It also has the same hyper-link structure. Of course, a tactic may prove the goal completely. In this case no subgoals would be generated.

5 Related Works

R. Stärk ([8]) converted to HTML formal theories of the Logic Program Theorem Prover. He represents theory content similarly to our main frame of the theory overview page, but the only provided hyper-links are pointing to linear structured theorem proofs.

HTML interface to Isabelle formal library [9] includes applets that represent theory dependency graphs and HTML pages with theory ML sources.

L. Mikušiak and M. Adámy ([6]) have developed Netscape plug-in for displaying formal mathematics in Z notations. It can display special symbols in different color and provide, on-demand, auxiliary information about them. Unfortunately, this plug-in available only for *Windows* 95^(TM) and *Windows NT*^(TM) and is not free ([7]).

The first time Nuprl formal theories became available on the Web as a result of the Web interface to the system designed by R. Vaughn and D. Svitavsky ([4]). This presentation was based on interaction with a copy of Nuprl running on the server. It provided access to theorem and abstraction definitions, including map-based access to term structure. Unfortunately, interface was not very stable because of the internal bugs. It also worked slowly since it relied on running copy of Nuprl and was using big image files to represent formulas. Paul Jackson has used similar technique to publish a "static" library of his theories without interaction with any server script. He provided more stable access to the library, but his presentation lacked term structure information and proofs. In addition, web pages took too much time to download.

The initial version of the Web Publisher, developed by the author in 1996, was the first to provide static access to formal proofs and to use special symbol images instead of converting the entire formula to a graphic file. It was used by several members of Nuprl group to publish their own theories on the Web. S. Allen has modified this converter to supplement each proof page with the lemma statements and and definitions of abstractions used on it.

The new version of the Web Publisher, presented in this paper, adds help frame to provide access to term structure and definitions of the abstractions, used in the term.

6 Acknowledgments

Author thanks Stuart Allen for several discussions that had a large influence on the final design of the Publisher, Robert Constable for comments on the draft of this paper, and Richard Eaton for helping make the converter much faster.

References

[1] R.L. Constable et al., *Implementing Mathematics with the Nuprl Proof Developing System*, Prentice Hall, 1996

[2] C.L.T. Mannion and S.F. Allen, *A Notation for Computer Aided Mathematics*, Cornell CS Technical Report TR94-1465, January 1995

[3] S.F. Allen, *From $\partial y/\partial x$ to $\Box P$: a matter of notation*, Proceedings of User Interfaces for Theorem Provers '98, *to appear.*

[4] R. Vaughn and D. Svitavsky, *Nuprl/WWW Browser Design.* http://www.cs.cornell.edu/Info/Projects/NuPrl/html/design.html

[5] *Mathematical Markup Language (MathML) 1.0 Specification.* World Wide Web Consortium Recommendation. (available at www.w3.org)

[6] L. Mikušiak and M. Adámy, *Publishing Formal Specifications in Z Notation on World Wide Wed*, TAPSOFT '97: Theory and Practice of Software Development, M. Bidoit and M. Dauchet (Eds.), Springer 1997, pp. 871-874

[7] *Z Browser Plug-in* http://www.ora.on.ca/z-eves/zbplugin.html

[8] R.F. Stärk, *A Logic Program Theorem Prover* http://www2-iiuf.unifr.ch/tcs/robert.staerk/lptp/index.html

[9] Isabelle Online Theory Library http://www4.informatik.tu-muenchen.de/ isabelle