On Dynamically Presenting a Topology Course

Paul Cairns and Jeremy Gow
({p.cairns,j.gow}@ucl.ac.uk)
UCL Interaction Centre, University College London, 26 Bedford Way, London WC1H 0AP, UK (www.uclic.ucl.ac.uk/imp)

Peter Collins
(pjcoll@maths.ox.ac.uk)
St Edmund Hall, Oxford University, Oxford OX1 4AR, UK

Abstract. Authors of traditional mathematical texts often have difficulty balancing the amount of contextual information and proof detail. We propose a simple hypermedia framework that can assist in the organisation and presentation of mathematical theorems and definitions. We describe the application of the framework to convert an existing course in general topology to a web-based set of materials. Two user evaluations of the materials indicated both successful and unsuccessful aspects of the framework. We discuss further lines of investigation, in particular, the presentation of larger bodies of work.

Keywords: proof presentation, hypertext, Polya, structured proof

1. Introduction

When considering the organisation, retrieval and presentation of mathematical knowledge, it is important to consider the needs of those who use that knowledge. Necessarily, mathematicians are quite able to work with existing knowledge management provided by journals, textbooks, Mathematical Reviews [13] and so on. However, with the organisation of substantial quantities of mathematics in digital form, it may be that new ways of working will be made available to mathematicians. Indeed, it could be argued that without significant advantages from adopting new, digital bodies of knowledge, mathematicians are unlikely to want to move away from the tried and tested tools that already exist [12]. Thus, consideration of mathematicians as users of mathematical knowledge must play a key rôle in the uptake of mathematical knowledge management tools.

This paper addresses the possibility of providing a new concept of presentation for mathematics. Humans read mathematics not just for the satisfaction afforded by a complete, logical deduction but also for the new insights into the mathematical objects described — the ideas of mathematics are generally more important to humans than the bare logic. Using computers, we claim it is possible to provide interactive presentations that expose and explicate the central ideas of mathematical results without losing the necessary rigorous logic.

Given the centrality of proof to modern mathematics, we have focused on developing a new way of presenting mathematical proofs. The goal is to use hypertext as a means of balancing the motivation and intuition surrounding mathematical results and the details of their proofs. The paper begins by outlining the issues raised in traditional presentations of proofs. This leads to the proposed new presentational framework inspired by the work of Polya [16] and Lamport [11]. It is essential that we do not lose sight of the user in this work, so the framework was implemented in two versions and each version evaluated by mathematicians. As there is a huge range and diversity of mathematicians, we have chosen users for whom rigorous proof is a main feature of their work, namely, undergraduate and research level pure mathematicians. The paper describes the implementations and the results of both evaluation sessions. The consequences for future work in this area are also discussed.

At the outset, it is worth distinguishing the difference between the work done here and possible educational goals. Naturally, any new form of presenting proofs has implications for educational support for teaching mathematics, but education is far more than useful presentations [18]. Therefore, we do not assess the educational value of such presentations, but instead how this new approach compares or complements existing mathematical knowledge management, for example, books. We consider the target users' experiences of the new presentations and not on any consequent educational outcome.

2. Context and Detail

On paper, much of modern mathematics is organised into a standard form: definitions and axioms come first, then theorems that follow from them, then the proofs of those theorems, then further theorems and proofs, more definitions and so on. And even within a single proof, the statements of the proof follow in logical progression from the previous statements, theorems or axioms. However, there are well known problems with this format:

1. Without sufficient contextual content, the specific form of definitions and theorems may seem arbitrary[10]...
2. ...however such context is essentially irrelevant to the logical argument.
3. Too much detail in proof obscures the important insights [20] ...

4. . . . but with insufficient detail a reader will not be able to follow the argument [15].

With contextual knowledge and logical detail as orthogonal aspects of proof, there is clearly tension between presenting too much or too little of either. When authors prepare printed mathematical text, they necessarily take a cut through all the possible material that they could present in the hope of providing a balanced quantities of context and detail. Almost inevitably, the balance chosen is not appropriate for all readers. Indeed, it may not be appropriate even for a single reader at different times. For instance, a reader may require a detailed proof initially but after digesting the proof would no longer need the details and would look rather for the context and motivations behind the proof.

With hypermedia, mathematical authors need not be constrained to choose specific levels of context and detail. Readers could interact with on-line materials to manage the presentation to their satisfaction, without unnecessary materials being forced upon them.

Though logical detail is necessarily well defined, what constitutes context is not so clear. We take the context of a result to be basically any extra-logical material that might help a reader to understand what the result is and why the reader should need to know it. Whereas the strict logical statements of definitions and the arguments followed in proofs are somehow the actually substance of mathematics, insights come from a variety of sources. Some concepts seem obscure without the rationale that led to their particular form. A typical example of an obscure definition is the concept of bounded variation. It in fact evolved from being a technical condition in theorems related to Riemann-Stieltjes integration [10]. The importance of a particular result often comes through its relationship to other results. Wiles' proof of Fermat's Last Theorem, though sensational for the actual result, is more important mathematically for the unification of two previously disparate branches of mathematics [21]. This also shows that sometimes it is the proof itself that is significant about a particular theorem.

As well as external influences, there may be internal pressures that lead to the particular form of a theorem. For instance, why is a certain condition important to a particular theorem? And how does the theorem apply in specific situations? These questions can be answered through examples illustrating the theorem, alternate forms of a theorem, diagrams, and also examples that illustrate what happens when certain conditions are omitted from the statement of the theorem. All such extra-logical material, be it external or intrinsic to the theorem, we refer to as context. It is this context together with a theorem's statement and the logical details of it's proof that constitute the mathematical knowledge about a theorem, all of which we aim to manage with a good structured presentation.

3. The Polya-Lamport Framework

The aim of the framework is to provide a structure for presenting the detail and context of a theorem or definition in such a way that users can control the content. The outcome of the framework will be a way of producing web pages that structures and organises mathematical proofs and brings measurable benefits to users. As such a thing has not been attempted in this way before, it is not possible to ask users what they would like. Instead, we have motivated the framework from a novel synthesis and extension of two existing suggestions for improving the understanding of mathematics.

3.1. Polya's Four Steps

Contextual material is useful when trying to solve problems. This was carefully discussed and demonstrated by Polya [16, 17]. He proposed an approach to problem solving that actively applied heuristic thinking. Rather than considering a problem in isolation, the problem-solver is encouraged to make links with similar problems, address the problem in concrete or limiting cases and to reflect on the contribution of the problem to wider knowledge. Polya set out four steps that the solver should follow:

Understand	the problem with examples, diagrams and a careful examination of each of the terms and unknowns in the problem
Plan	how you intend to solve the problem
Execute	the planned solution with care
Reflect	on the result, how it relates to other results and how it might be proved differently

In other words, Polya promotes thinking about the context of a problem in order to solve it (**understand**) and having solved it to bring it back into the wider theory from which it arose (**reflect**). The solution is also explored at different levels of detail (**plan** and **execute**).

Just as Polya's steps are appropriate in understanding a problem, they are helpful in understanding a theorem too. After all, for a particular person every theorem is initially an unproven challenge. We propose to use these steps as a *framework for the presentation of theorems*. The **understand** step is used to clarify the statement of the theorem and to motivate it's formulation. This may involve links to other parts of the mathematical theory, e.g. a definition that is used in the theorem, or a previous result that is analogous to the current one. The **plan** step can be used to provide an overview of the key steps in the proof and also provide links to prior results that are essential to the proof. The **reflect** step can provide links to further theorems and definitions and so position the theorem within the theory. This leaves the **execute** step which must corresponds to the detailed proof of the theorem.

3.2. Lamport's Structured Proof

It is the **execute** step that poses the challenge of providing a sufficient but not overwhelming amount of logical detail. Lamport has proposed structured proof, an approach to managing proof detail which he has already employed to good effect organising proofs on paper [11].

A proof written in this format consists of lists of key proof steps, each recursively justified by a subproof. The proof bottoms out with an appeal to one or more previous proof steps. In this way, the high level steps give the outline of the proof and the low level steps the details. The obvious adaptation to hypertext is to hide or reveal the subproofs under the reader's control. This was first done by Grundy [7] for a calculational style of proof (i.e. a linear derivation). We propose to implement the **execute** step of our framework with an expandable/collapsible structured proof based on Lamport's original, more general design.

The framework for theorems, then, is to use the Polya steps to structure the presentation of a theorem, and within that to use the Lamport's format to structure the proof . In this way, the reader can control the amount of context and detail that they wish to see. The theorem presentation framework is shown in Table I.

3.3. Presenting Definitions

We also extend the framework to cover definitions. Definitions are an integral part of mathematics and, indeed, developments in definitions seem to drive and be driven by developments in theory [10]. As there is not generally a proof associated with a definition, the amount logical detail is not a problem, but it would still be useful to provide a context for a definition. We therefore adapted the Polya steps to definitions. Clearly, the **understand** step — that is, providing explanatory and motivating links with existing definitions and theorems, and giving examples and diagrams — carries over to definitions easily. Furthermore, some definitions are best understood by seeing the results in which they are used. Thus, it makes sense to also provide links from this section

Table I. A summary of the Polya-Lamport framework, for presenting theorems and definitions. Steps marked with * are not used for definitions.

Step	**Purpose**	**Links to**
Statement	Statement of theorem/definition	—
Understanding	Explanation of statement	Diagrams, definitions
	Motivation for theorem/definition	Theorems, definitions
Plan Proof*	Results used in proof High level proof description	Theorems
Execute Proof*	Interactively expand plan of proof	—
Reflect	Variations on statement	Theorems/definitions
	Use in subsequent theory	Theorems, definitions

out to key theorems that use the definition. The **reflect** step can also be useful here with the same meaning as for theorems. Both the **plan** and **execute** steps are omitted, as there is no associated proof.

3.4. The Whole Framework

Thus, the complete Polya-Lamport framework organises the presentation of both definitions and theorems, as summarised in Table I. There are consistent places at which to provide the various kinds of knowledge associated with a theorem or definition. This allows a reader to have an expectation about what a particular piece of text will say, giving greater structure to, and control over, their reading.

A link to another theorem or definition page may appear in a variety of positions within the framework. Each position corresponds to a different relationship the linked page may have to the current page. This allows a reader to navigate the mathematical theory in a more informed manner.

4. Applying the Framework

In order to evaluate the usability and usefulness of the framework, it was applied to converting a set of undergraduate course notes in general topology. This course, "Elements of Euclidean and Metric Topology" by one of us (Collins), is taught to all first year mathematics undergraduates at Oxford University. The course consists of six chapters (including a zero-th chapter containing necessary background material) covering basic notions of continuity, connection and compactness in metric spaces [22] as well as touching on more abstract topological ideas. It is typical of many mathematical texts, consisting of numbered theorems, definitions and proofs, some more general discussion of the topic and exercises for the reader.

The topology course represents only one type of proof presentations at which the Polya-Lamport framework is targeted. Other types could be textbooks or journal papers. Work is currently underway in adapting a journal paper to find out the response of a research community to this sort of presentation.

4.1. Adapting the Materials

To produce the on-line notes from the existing notes for the course, the notes must be re-structured in terms of the Polya-Lamport framework. This involves going through the entire materials and fitting the content into the different sections of the framework.

Like most mathematics textbooks, there are obvious definitions and theorems and these can be picked out immediately for the framework. However, in some places, there is text which is not under either heading. On analysing it, this tended to be extra-logical material that either helped in explaining what was to come, such as the form of a particular theorem, or extended what had been done to show its broader relevance. This material fitted naturally into the understanding and reflection sections, respectively.

The proofs of theorems were not in sufficient detail to satisfy all readers, nor were they organised in the hierarchical structure that Lamport proposed. When converting proofs into the hierarchical form of the framework, there had to be a decision as to how far to take the proof detail. Lamport's advice is to recurse "until the lowest level statements are obvious, then continue for one more level" [11]. Also there were decisions made about which parts of the proof were higher up the hierarchy and which lower. In some cases, this was obvious. I n others, it was a matter of choosing which parts of the proof required more emphasis and which less. This is a value judgement and it would not be trivial to automate.

The exercises of the original notes fell into two types. Some were questions that required applying the knowledge presented in the notes to new problems. As such, this fitted with our intention of the reflection step — developments of the basic statements. Such exercises were placed as questions in the **reflect** part of the most relevant theorem or definition. Answers to the exercises were not included in the original notes so were not represented in the framework.

Other exercises were actually ways of developing new ideas by making a new definition and extending proofs or results from the notes to make use of the new ideas. As such definitions were generally explicit, the exercise became a new definition in the framework and the question parts of the exercise went into the **reflect** step of the definition.

The framework also suggested further material that ought to be included for every theorem and definition such as an intuitive motivation for the theorem, a diagram and relations to other similar results. Wherever possible, such new material was taken from the notes but otherwise was made by the author.

For higher level organisation, above that of individual theorems and definitions, the chapter structure of the original notes was retained. The zero-th chapter of the notes was not included though as this was simply to introduce background material. This was adequately covered by direct references to background material from the theorems or definitions where it was relevant.

4.2. Encoding the Framework

The Polya-Lamport framework is essentially a breakdown of mathematical content into parts that better indicate the intention or rôle of that content for, say, a particular theorem or definition. As such, it makes most sense to consider the framework as a mark-up of mathematical content. For this reason, XML [2] was used as the internal representation of the framework.

The framework was thus captured in an XML DTD[1], most of which is straightforward. One complication comes from the grammar of Lamport's structured proofs. Lamport originally described this approach by example [11]. However, a corresponding grammar was extracted and formalised as part of the DTD.

There are other internal representations that could have been used that a more oriented to representing mathematics, for instance, OMDoc [8]. Although we choose to design our own XML DTD to meet our needs, OMDoc is itself an XML codification so, in principle, there is no reason why we could not have used it. We intend to look at generating Polya-Lamport style presentations from OMDOC.

4.3. Presenting the Course

The on-line notes exist in two versions, the second being produced after feedback on the first. However, there are core features that are constant between the versions.

[1] Available at `www.uclic.ucl.ac.uk/imp/imp.dtd`

Each version is essentially a set of pages coded in HTML based on the original versions in XML. Each page contains a single definition or theorem in the framework together with their appropriate steps such as **understand** and **reflect**, as in Figure 1. References from these sections are links to other pages or diagrams.

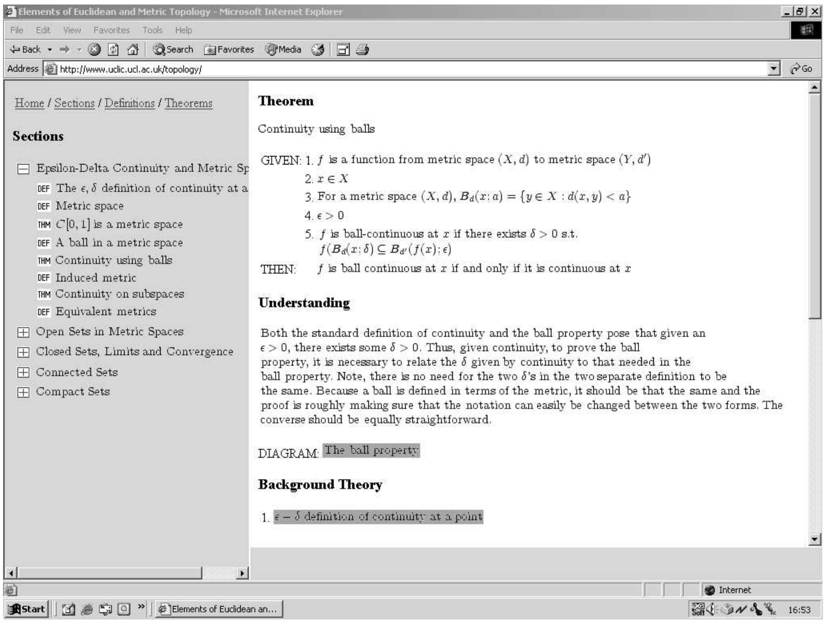

Figure 1. A page presenting a theorem.

For each theorem page, the structured proof is an interactive version of Lamport's hierarchy: the natural hierarchy forms a expandable/collapsible tree, akin to the tree representations found in products such as the Windows Explorer. This can be seen in Figure 2. The parts of the tree can be collapsed or revealed by clicking on the $+/-$ squares beside statements in the proof. A plus in the square means that clicking will reveal the steps the prove that statement. A minus means that the sub-steps are already revealed and clicking on it will hide them again. Thus, users are free to control the level of detail in the proof that they actually see.

The pages are themselves organised using HTML frames with a 'navigation side-bar' layout that is becoming standard on web-sites. Theorems and definitions are grouped into sections. The section titles and their contents can be examined in the side-bar, and clicking on a theorem or definition brings up the corresponding page in the main

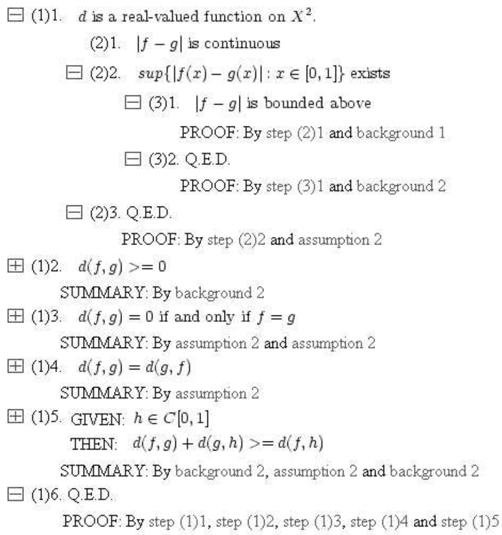

Figure 2. A partly expanded structured proof.

window. Additional links allow the side-bar to display all the theorems or all the definitions from all of the sections.

The second version of the online notes can be found at `www.uclic.ucl.ac.uk/topology`.

4.4. Differences Between the Versions

In the first version of the online notes, the structure proof was implemented using a Java applet. An XML description of each proof was produced by hand, and read by applet via JDOM, the Java document model for XML. The applet gave the interaction required for the expandable/collapsible proof that makes up the structured proof. In the second version, this was replaced by a combination of JavaScript [5] and HTML. The result was that the structured proof was an integral part of each page. Furthermore, the entire mathematical text was marked-up using XML, and transformed to a HTML/JavaScript presentation using XML stylesheet transformations, XSLT [4]. This considerably simplified the authoring process.

Server side implementations were also considered but it was felt that relying on server connections to read a single page (at least as it appears to the user) could cause frustrating delays and greatly reduce user satisfaction.

The structured proof of the second version also naively summarised hidden steps. Specifically, it simply gave any external links that were used to prove the hidden step. The idea was that it would correspond to a phrase like "which follows by the Heine Borel Theorem" and if the user really wanted to see how it followed, they could reveal those hidden steps.

In both versions, there were problems in producing mathematical symbols on web pages. While there is no real built-in support for mathematics in the common browsers, this will continue to be a problem. In the first version, TtH was used to produced the symbols outside of the structured proof and TeX4ht was used for producing GIFs of the symbols that the Java applet could use inside the structured proof [6]. With the second version, the distinction between the two parts was not made, so the single system TeX4ht was used, as it produced consistently better symbols. In fact, in order to get a very uniform appearance, all of the text was produced as GIFs so that the symbols did not obviously stand out, as is the case with TtH and Latex2HTML.

Coordinating the use of XSLT, the generation of GIFs for symbols and HTML content presented some challenging implementation problems. Though not relevant to the current paper, it is worth noting that the lack of native support for mathematics in commonly used browsers is an important issue. Until resolved, any such mathematical knowledge management system is going to have rely on either complicated implementations or third-party software to deliver its content using standard Internet technology. In fact, the latter is not really an option — requiring users to download special software is a common cause of user dissatisfaction with web-sites.

Because of time constraints and because it was only intended as a pilot study, the first version was not a complete version of the notes. Instead, it was made up of only the first four chapters of the original notes. The second version was complete.

5. User Evaluations

5.1. Pilot evaluations

For a pilot study, seven undergraduates were given tasks to perform with the first version of the web-based course notes. They were observed while they were performing the tasks. Afterwards they answered a SUS questionnaire [3] and were interviewed about their experiences of the interactive materials. Though seven undergraduates is hardly a large sample set, we have followed Nielsen's view that five or six users are sufficient to find all the major problems in a user interface[14].

As a quick and dirty measure of usability, the SUS questionnaire formed the basis of the evaluation. It consists of a set of fairly general questions about the users' subjective satisfaction with the interface. The pilot study gave us an SUS score of 78% which we understand to mean that the users are well satisfied with the system though there is some room for improvement. Unfortunately SUS does not give specific feedback on what those improvements should be!

From the observations of users working with the materials, it seemed that they were able to find materials reasonably quickly. However, the structured proof seemed to cause some navigational problems. Being an applet embedded in the page, it had its own set of scroll bars within the browser window and expanding and collapsing parts of the proof tended to move material off the screen. This was borne out by dislikes mentioned in the interviews.

Several of the users commented on the expandable proofs as a good feature of the materials. This is a good sign of the value of the hierarchical proof structure though it must be taken with a pinch of salt: the users may not have had sufficient time to make a balanced assessment of the proofs. And positive comments notwithstanding, the users had plenty of suggestions of how they might like to see the applet improved including: more use of colour; links in and out of the proof applet to the materials; and the ability to annotate proofs like a paper proof.

Another positive feature that users generally brought was the links to supporting material, in particular, the ease with which definitions could be found. This seems to support the breakdown of the materials into the Polya steps.

A significant comment made by a couple of the users was the unfamiliarity of the hierarchical proof structure. This comment has also been made in other presentations of the material in seminars and workshops.

5.2. Subsequent Implementation Changes

The second version of the on-line materials was prepared in light of feedback from the pilot study. The integration of the structured proof step into the 'flow' of the HTML page removed several of the problems. The applet has gone and with it the navigation problems that it caused. This also meant that the structured proof was integrated with the other steps, and it was consequently possible to link in and out of the proof itself.

Colours were used to give different meanings to the different types of link. It was decided not to include the possibility of annotating the pages partly because it was felt to be aside to the issue of presentation generally and partly because of practical problems with server support.

5.3. Second Evaluation

The second evaluation period was a year after the first and used the second version of the on-line materials. The method of the study was considerably different. This used a qualitative approach and has initially only involved six students.

In order to get better qualitative feedback, three students were given access to the on-line notes for a period of three weeks during which they were also studying the course through lectures and tutorials. They were then given a semi-structured interview aimed at eliciting their experiences of using the notes. For comparison with the pilot, they were also given the SUS questionnaire to fill out.

The other three students were also in the pilot study and we gave them an e-mail questionnaire made up of mostly open-ended questions based on the semi-structured interview given to the first three students. The aim was to see if there was any change in attitude or experience towards the newer version of the notes.

In both cases, the data from these sources is not fully analysed. However, some initial remarks are possible.

First, it seems that navigational problems apparent in the pilot were no longer present. Students when navigating through the pages and steps in the pages experienced no navigational problems or feelings of being 'lost'. This may of course be due to the different trial conditions — users had a longer time on their own in which to feel more comfortable with the navigation. In any case, this seems to have removed a distraction allowing them to focus more on the content of the pages and so there were many more comments about the particulars of the presentations themselves.

Secondly, there is a suggestion that the form of the structured proof step is too unfamiliar. It jars with the users' other experiences of mathematical proofs and they did not seem willing to spend time working through the unfamiliar style. This may be natural, but does not necessarily constitute a flaw in the presentation because more work invested in something tends to lead to deeper understanding. However, this is certainly a hurdle that needs to be over come if the presentations are to be widely accepted.

6. Discussion and Future Work

The pilot studies seem to show that the Polya-Lamport framework has at least solved some problems of balancing context and details of a proof. It was noticeable that no users commented on difficulties in using the Polya steps. We take this as a positive sign that the steps seemed natural and sensible or at the very least did not hinder users. There is more data still to be analysed from the second evaluations directly related to the value of the Polya steps.

The hierarchical proof seems to have a very ambiguous status in the framework. This is a bigger issue for mathematical knowledge management as the hierarchical presentation is an obvious way of presenting variable levels of details. Also, it falls out naturally from proofs produced by theorem provers and has already been implemented as a presentation mode of a particular theorem proving system [1]. However, implementational simplicity will not persuade mathematicians of its value. Thus, some improvements are essential. For example, whilst a hierarchical structure may be best for the internal representation, considerable pre-processing could be done to produce a more 'normal' sort of proof. However, there is then the question of how users might reveal the hidden details in a way which is meaningful in the context of the 'normal' proof.

It may be that a hierarchical presentation ends up being unacceptably different from traditional proof styles despite some initial enthusiasm for it. Or alternatively, if mathematicians really do want to see varying levels of detail, they may have to give up their familiar and well-worn styles of presentation! This would be a valuable direction for future research.

There are some interesting matters arising from the activity of authoring such notes. One matter of wider relevance to mathematical knowledge management is that of organising examples. In the Polya-Lamport framework, they sometimes did not have a natural fit with either definition or theorems being a combination of both. For small examples of the sort in this course, though, this was not insurmountable. But for larger examples, their rôle can be very difficult to classify. A single example at times can simply be defined, can be proven about, can serve as a motivation for new ideas, can serve as a counter-example to conjectures. Though it could be possible to include these into the internal representation of the particular example, it becomes difficult to maintain when new uses for the example are found. This suggests that even simply investigating examples as a interface objects between mathematicians and mathematics could be an important and fruitful area.

As with any hypertext system, there is the problem of maintaining a consistent, reliable and memorable way of producing a large amount of inter-linked documents. This is particular relevant for the framework as there are not only links between pages but also many links within the parts of a proof. And if user comments are to be followed, more links between pages and parts of the proof are needed! Without good authoring tools, it is unlikely that linking will become any easier.

The framework so far only covers the presentation of particular statements, be they definitions or theorems. As the topology course was reasonably small and self-contained, navigating through it was not too taxing for the user. This may not be the case with larger bodies of work, be they large theories and/or large proofs. The user will need to be able to find appropriate content easily through a corpus of possibly several hundred pages. Moreover, when considering whole theories rather than a particular, focused topic, the context and justification of the theory could be as useful as the context of individual theorems.

The Polya-Lamport framework provides no place for theory-level context. It will be necessary to look elsewhere for inspiration. Lakatos suggests revealing the context and evolution of a theory using a dialogue with different characters representing different philosophies that affected the theory [10]. Indeed 'Proofs and Refutations' is a fine demonstration of how the development of Euler's formula for polyhedra could be rationally reconstructed. But despite having been around for many years, there is not a stream of mathematicians repeating this process on other branches of mathematics. Our feeling is that the historical knowledge and literary skill required to develop such a presentation took an author so far away from mathematics that only exceptional mathematicians would be willing or even able to attempt it. We felt vindicated on finding only one other dialogue presentation of a theory – it was written by Knuth [9].

This does not mean that a rational reconstruction of the development of theory is impossible. Perhaps the dialectic element could be used to structure collections of hyper-linked materials. But this is clearly a matter requiring a great deal more thought and attention.

Throughout any future work, it is essential to ensure that what is produced is what users need. This can only be done with more in-depth and focused evaluations. We have plans to produce a larger topology course and also the proof of a large and complex theorem [19]. These will be evaluated in user trials in the course of the project. We also have interest from other mathematicians in the UK and we are hoping that they may provide a variety of theories, approaches and styles with which to investigate the range of applicability of the Polya-Lamport framework.

7. Conclusion

Presenting mathematical proofs has conflicting concerns in providing a balanced amount of context and detail. Our overall aim is to develop a conceptual hypermedia framework for interactively presenting proofs that real mathematicians would like to use. The novel Polya-Lamport framework goes some way to this goal, organising context through a series of steps and managing detail using an interactive, hierarchical proof structure. Early user feedback indicates that this provides a usable environment that certainly improves upon naive approaches. There are possibilities to improve the framework but for us the more interesting question is how to organise entire mathematical theories, and this will be a focus of future work.

Acknowledgements

Many thanks to the undergraduate mathematicians of St. Edmund Hall, Oxford University. This work was funded by EPSRC, UK, grant GR/N29280.

References

1. A. Asperti, L. Padovani, C. Sacerdoti Coen, F. Guidi, I. Schena, 'Mathematical knowledge management in HELM,' *First International Workshop on Mathematical Knowledge Management*, 2001
2. T. Bray, J. Paoli, C.M. Sperberg-McQueen & E. Maler (eds) , *Extensible Markup Language (XML) 1.0, 2nd edn*, W3C Recommendation October 2000 `http://www.w3.org/TR/REC-xml`
3. J. Brooke, 'SUS — a quick and dirty usability scale', `http://www.usability.serco.com/trump/methods/satisfaction.htm`
4. J. Clark (ed), *XSL Transformations 1.0*, W3C Recommendation November 1999 `http://www.w3.org/TR/xslt`
5. D. Flanagan, *JavaScript: The Definitive Guide, 3rd edn*, O'Reilly, 1998
6. M. Goossens & S. Rahtz, *The LaTeXWeb Companion*, Addison Wesley, 1999
7. J. Grundy, 'A Browsable Format for Proof Presentation,' *Math. Universalis*, 2, 1996 `http://www.ant.pl/MathUniversalis/2/grundy/mu.html`
8. M. Kohlhase, *OMDOC: An Open Markup Format for Mathematical Documents.* SEKI Report SR-SR-00-02, Universität Des Saarlandes, 2000.
9. D. Knuth, *Surreal Numbers*, Addison Wesley, 1974
10. I. Lakatos, *Proofs and Refutations: The Logic of Mathematical Discovery*, Cambridge University Press, 1976

11. L. Lamport, 'How to write a proof', *American Mathematical Monthly*, 102(7) p600-608, 1994
12. L. Lamport, 'How LaTeX changed the face of Mathematics: an E-interview with Leslie Lamport,' `www.cs.toronto.edu/pub/fanis/lamport-latex-interview.ps`
13. *Mathematical Reviews*, American Mathematical Society, `www.ams.org/mr-database`
14. J. Nielsen, *Usability Engineering*, Academic Press, 1993
15. L. C. Paulson & K. Grabczewski, 'Mechanizing Set Theory,' *Journal of Automated Reasoning*, 17 p291-323, 1996
16. G. Pólya, *How to Solve It*, Penguin Books, 1990
17. G. Pólya, *Mathematics and Plausible Reasoning*, Princeton University Press, 1968
18. P. Ramsden, *Learning to Teach in Higher Education*, Routledge, 1992
19. M. E. Rudin, 'Nikiel's conjecture,' *Topology and its Applications*, 116, p305-331, 2001
20. B. Russell, 'The Study of Mathematics' in *Mysticism and Logic*, Penguin, 1953
21. S. Singh, *Fermat's Last Theorem*, Fourth Estate, 1998
22. W. A. Sutherland, *Introduction to Metric and Topological Spaces*, Oxford University Press, 1985
23. Y. L. Theng & H. Thimbleby, 'Addressing Design and Usability Issues in Hypertext and on the World Wide Web by Re-Examining the 'Lost in Hyperspace' Problem,' *Journal of Universal Computer Science*, 4(11), p839-855, 1998