Enabling Web-access to a Database of Calculus Problems Using LaTeX, PHP and LaTeX2HTML

Kalle Saastamoinen, Jaakko Ketola, Tuukka Kurppa and Liisa Torikka

Laboratory of Applied Mathematics
Lappeenranta University of Technology
P.O. Box 20, FIN-53851 Lappeenranta, Finland
tel. +358407606489; fax. +35856212898
Internet: `kalle.saastamoinen@lut.fi`

Abstract

In this paper, we present work which we have done for providing the material used in the teaching of calculus to students at Lappeenranta University of Technology. This procedure can be divided into three separate steps: 1) creation of the LaTeX files, 2) conversion of the base LaTeX files to the HTML files, and 3) implementation of these HTML files in the PHP database. So far, we have stored hundreds of exercises with hints and solutions on our Web server, in addition to some theory that we felt is necessary for the students. In this paper we will also discuss the benefits and future of our functioning Web environment [1].

Introduction

In the beginning of 2002, we initiated an investigation to develop possible solutions for providing the material used in the teaching of calculus to students at Lappeenranta University of Technology. We decided to write the pages in question in the LaTeX language. We also wanted to use database and network solutions for the finalized version of these pages. We chose to use network solutions because the material, which is available on the Internet: i) is easy to reuse, transform and combine with other materials, ii) can be easily accessed, at least with fast connections, iii) is accessible 24 hours a day, iv) can be accessed by many users at once. We wanted to use a database in order to provide students with tools for performing searches on the calculus information. The students can then study the material that they feel is the most interesting and then hopefully begin to carry out their own research and form a view of mathematics. Therefore we have taken a constructivist approach to the learning process. The LaTeX files, which are entered into the database with all the necessary keywords, form the core of the solution. Based on the keywords, the database knows where each file is located, including specifics, from the area of mathematics, to whether the file is an exercise, solution, hint or general information. The database alters the user's view of its contents by using the keywords that are included in the above-mentioned base LaTeX files. Now, any of our lecturers can contribute new material to the database by using a very simple interface and LaTeX. Lecturers can also easily combine weekly tutorials and even whole examinations from the contents of the database. So far, we have stored hundreds of exercises with hints and solutions on our Web server, in addition to some theory that we felt is necessary for the students. In this paper, we present the complete procedure from the creation of the LaTeX files to a functioning Web environment [1].

Our lecturers have found that this new interface saves a significant amount of time and is also flexible enough for their needs. Student satisfaction with this new material is also apparent, since the material is now always available, information is easy to find, and the material is clear.

Converting LaTeX files to a functioning Web environment

All of our LaTeX files are articles or exercises which are executed by a cover file. Article files include theory, exercise files include tutorials of different kinds, and cover files (`kuori.tex`) are files which call the previously mentioned document files. Cover files use `math.sty`, which includes macro definitions of mathematical formulas and all other necessary definitions. DVI files are generated by execution of a cover file. All of these LaTeX files were originally created by a lecturer, Simo Kivelä, from Helsinki University of Technology (HUT).

Kalle Saastamoinen, Jaakko Ketola, Tuukka Kurppa and Liisa Torikka

Creating LATEX files Base LATEX files, articles and exercises include categorization which defines the final destination of the file in our Web environment. Categorization is done in the following way and order:
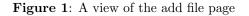
- At the beginning there is \begin{Artikkeli} or \begin{Tehtava}, which defines if the file is an article or an exercise.

- \tunniste{}: The name of the file, for example \tunniste{lta179}. (The value comes from the words Liisa, Torikka, article and the number of the article.)

- \alue{}: The highest categorization in our content dictionary; for example \alue{tavdy}, meaning ordinary differential equation.

- \luku{}: The next categorization area, a section; for example \luku{2kdy}, meaning second order differential equations.

- \kappale{}: The lowest categorization area, a chapter; for example \kappale {vali}, meaning constant linear coefficient.

- \otsikko{}: The name of the file, for example \otsikko{2ODYvaki lineaariset DY:t}, meaning second order constant linear coefficient, ODE.

- \luonne[]{}: Two arguments which characterize the file, for example \luonne[Matlab]{teoria}. Here, the first argument, 'Matlab', specifies that you need a computer program called Matlab to run the issued theory part; the second argument, 'teoria', means theory.

- \kuvaus{}: A short description of what the file includes.

- \tekija{}: The name of person who typed the file, for example \tekija{Liisa Torikka}.

- \pvm{}: The date, e.g., \pvm{29.8.2002}.

- \kieli{}: The language, with Finnish as the default.

- lahde{}: The original source, for example lahde {LTKK/Pekka Jauhon moniste, kevät 2002}.

- \kayttooikeus{}: Copyrights.

- \lahdekoodi{}: Source code, for example \lahdekoodi{LaTeX}.

- \laitos[]{}: The level of the material is the first argument, and the institution name is the second; for example, \laitos[hard]{LUT}.

- Then come keys to help database searches, for example \avain[Calculus] {course}, normally we use many keys to make files easy to find from our database.

- After these definitions comes the actual article or exercise.

All of these files which are written are then included in the previously mentioned cover file, executed and tested.

Conversion After we have a functioning LATEX file, let us say abc.tex, we will run a script that uses the LATEX2HTML translator to install this file into our Web database. The script is done in PERL and it transfers the original LATEX file into the package of files called abc.tar.gz which includes original file abc.tex in pieces of converted HTML files.

Add exercise or article
Modify exercise or article/Remove exercise or article

File:
(abc.tar.gz) [   ] Browse... | Send

Figure 1: A view of the add file page

Adding a file into the database After this, the package, here named `abc.tar.gz`, is ready to be sent into our server using a very simple form; see figure 1. Once a person pushes a button called Send a PHP-script does the following things for the file:

- it will be unpacked,
- the contents will be checked,
- the information will be added into the database and
- HTML-pages will be sent to the right directory

All this is done based on the information included in the original LaTeX file.

Finalization After all this is done we have our file in our Web page where the starting page looks like figure 2.

Figure 2: A view of the starting page of calculus

Now a user can perform different searches in our Web environment. A file which was added can be found from our categorization, from its place, or it can be found by doing searches with a word which can be any word included in the file categorization. For example, if we pretend that our file which we added had something to do with Euler, we could try to find it by entering the word 'Euler' into the search engine as in figure 3 and the outcome of the search looks like figure 4. This works well, at least if one knows an appropriate term for which to search.

Figure 3: A view of the search with the word Euler

Figure 4: Outcome of the search with the word Euler

The result of a successful search might look like figure 5.

Kalle Saastamoinen, Jaakko Ketola, Tuukka Kurppa and Liisa Torikka

Figure 5: An example of a file

Conclusions and future

We have gotten a lot of positive and very few negative comments from these pages that we have created. Our pages have been used in our normal basic courses and both lecturers and students have taken them as their own. We have also done both qualitative and quantitative usability testing for these pages and the results have been very promising.

In the future, we are, for example, going to automate the way the text will be caught into the paper in for example examinations. This will make usability in our pages better if one thinks for example of distance education. We will also continue developing these pages — animations, etc., will be included in these pages in the near future.

References

[1] Kalle Saastamoinen, Jaakko Ketola, Tuukka Kurppa and Liisa Torikka (2002). "Calculus Web pages", http://www.it.lut.fi/mat/virtuaali/matb/index.html