IT in university level mathematics teaching and learning: a mathematician's point of view

Borovik, Alexandre

2010

MIMS EPrint: **2010.71**

ISSN 1749-9097

IT in university level mathematics teaching and learning: a mathematician's point of view

Alexandre Borovik[1]

Abstract. University mathematicians are often selective in their approaches to the use of IT in teaching. Although mathematicians systematically use specialist software in direct teaching of mathematics, as means of delivery e-learning technologies have so far been less widely used. We (mathematicians) insist that teaching methods should be subject-specific and content-driven, not delivery-driven. We oppose "generic" approaches to teaching, including excessively generalist, content-free, one-size-fits-all promotion of IT. This stance is fully expressed, for example, in the recent Teaching Position Statement from the London Mathematical Society [2]. This text is an attempt to explain, at an informal level, this selectivity and its guiding principles. The paper is addressed to our non-mathematician colleagues and is not intended to be a survey of the existing software and courseware for mathematics teaching—the corpus of existing solutions is enormous and any technical discussion inevitably involves some hardcore mathematics.

Selectivity: why?

Intensive use of computers and IT in everyday mathematics teaching makes university mathematicians selective in their approaches to the use of IT in teaching. Our position is not rooted in ignorance or arrogance; on the contrary, I argue that mathematics deserves special treatment not only because of its highly specific cognitive nature, but also because the mathematical community has accumulated much more experience of using computers and IT in teaching, learning, research and communication than most of our non-mathematician and non-computer-scientist university colleagues have attained in their considerably shorter exposure to IT.

Historically, mathematicians (and computer scientists) were the first to use IT in teaching. Even in the era of mainframe computers, green displays and dot matrix printers, some serious work was done in this area (for example, mass generation of random problems of controlled level of difficulty in linear algebra and differential equations).

University mathematicians form a professional community; it is global and transcends national boundaries, but at the same time it is closely knit and connected in an efficient network. And yes, mathematicians were some of the first to use email, too; it started at least 30 years ago; the Internet at the time existed as a set of ftp sites and was unknown outside of mathematics, physics and computer science departments. Therefore a discussion of accumulated experience, tradition and collective wisdom of the mathematics community is well justified.

In short, the mathematics community has experience and knowledge of what can and cannot be done with computers. In that respect, we are not alone: to name a few, similar experiences have been accumulated in computer science, or, say, in language teaching. But we differ from our colleagues in some other subject disciplines who are still on a path of discovery. And I sincerely hope that IT solutions that do not work in mathematics teaching can be happily used elsewhere. However whilst many available products are suitable for many disciplines, they are unsuitable and unworkable for mathematics.

A case study: TeX

In the late 1970s, the great mathematician and computer scientist Donald Knuth launched a revolution in scientific communication by creating TeX [3, 4], a cross-platform computer language for typesetting mathematical texts. In one step, he brought mundane mathematical scribbles—not only research papers but also lecture notes, exercise sheets, seminar handouts—to the highest reaches of typographic art. Since the early 1990s, TeX and its dialect, LaTeX [5, 6], have been international standards for mathematical typesetting. But the routine everyday use of TeX and LaTeX in teaching in every mathematics department remains unnoticed and unappreciated by the wider education community. This is unfortunate, because TeX/ LaTeX is a pedagogical success story: it allows us to present even the most complicated mathematical formulae as structured and logically justified shapes, optimised for visual processing by the human eye and brain. After all,

[1] University of Manchester, UK, Email: borovik@manchester.ac.uk. Version of 12 August 2010.

[2] Mathematics degrees, their teaching and assessment. http://www.lms.ac.uk/policy/2010/teaching_position_statement.pdf.

[3] D. E. Knuth, TeX and Metafont: New directions in typesetting. The American Mathematical Society and Digital Press. Stanford, 1979

[4] D. E. Knuth, The TeXbook. Addison-Wesley, Reading, 1986.

[5] L. Lamport, LaTeX—A document preparation system—Users guide and reference manual. Addison-Wesley, Reading, 1985.

[6] http://www.latex-project.org/.

Typography may be defined as the craft of rightly disposing printing material in accordance with specific purpose; of so controlling the type as to aid to the maximum the reader's comprehension of the text. [7]

TEX succeeded in part because Donald Knuth spent years studying the millennia long tradition of calligraphy and the art of typesetting (five centuries old) [8,9], in the process producing such masterpieces as *3:16 Bible Texts Illuminated* [10] with 60 original illustrations by many of the world's leading calligraphers.

Speaking about commercially available software systems for teaching and learning in HE, we can safely conclude that in 95% of these products the mathematical presentation lags 20 years behind TEX ; their developers have not done their homework with the same care as Donald Knuth did his. Too frequently, IT developers and promoters of e-learning invite mathematicians back to the Stone Age.

The whole of this paper is written in LATEX; for those who have never seen how LATEX typesets mathematical formulae, Figure 1 shows an example of LATEX output in font type, size, and column width suitable for viewing on narrow screens of iPhones (some of my students indeed use their iPhones for access to bite-sized learning materials like exercise sheets—although iPhones are less convenient for reading more substantial pieces of text like lecture notes).

Figure 2 demonstrates spatial positioning of complicated formulae.

For dyslexic students, I can easily meet disability consultants' recommendations by setting up my lecture notes in landscape mode, double line spacing and a huge sans serif font, see Figure 3.

If further enhancement is needed, a font handling facility of LATEX allows the incorporation of specialist fonts like Lexia Readable [11] for dyslexic readers, but so far my students, when given the choice, have preferred the standard Computer Modern font in its sans serif version.

Students are also different

I first started teaching computer-based courses in 1995—and was not a pioneer since I used off-the-shelf software packages and associated textbooks, already developed by my colleagues elsewhere, and then tested, published, and reviewed. In my courses, the media of electronic communication were web pages and email. In

7 S. Morison, First Principles of Typography. Cambridge University Press, 1951. Quoted from: R. Lawrence, Maths = Typography? TUGboat 24 no. 2 (2003).

8 D. E. Knuth, Mathematical typography, Bull. Amer. Math. Soc. 1 no. 2 (1979), 337–372

9 D. Knuth, Digital Typography. Cambridge University Press, 1999 (reissue edition). ISBN-10: 1575860104; ISBN-13: 978-1575860107.

10 D. E. Knuth, 3:16 Bible Texts Illuminated. A-R Editions, 1990. ISBN 0-89579-252-4.

11 `http://www.k-type.com/?p=884`.

Here is Schrödinger's Equation:

$$i\hbar\frac{\partial}{\partial t}\Psi = \hat{H}\Psi,$$

and here is Stokes' Theorem:

$$\int_\Omega d\omega = \oint_{\partial\Omega} \omega,$$

and this is an infinite product expansion for the Gamma function:

$$\begin{aligned}\Gamma(z) &= \lim_{n\to\infty}\frac{n!\,n^z}{z\,(z+1)\cdots(z+n)}\\ &= \frac{1}{z}\prod_{n=1}^{\infty}\frac{\left(1+\frac{1}{n}\right)^z}{1+\frac{z}{n}}.\end{aligned}$$

Figure 1. Example of LATEX output optimised for viewing on narrow screens of hand handled mobile devices.

$$\begin{aligned}\frac{1+\sqrt{5}}{2} &= \sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\cdots}}}}\\ &= 1+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{1+\cdots}}}}\end{aligned}$$

Figure 2. This is how LATEX handles nested roots and continued fractions.

$$\Gamma(\mathsf{z}) = \frac{1}{\mathsf{z}}\prod_{\mathsf{n}=1}^{\infty}\frac{\left(1+\frac{1}{\mathsf{n}}\right)^{\mathsf{z}}}{1+\frac{\mathsf{z}}{\mathsf{n}}}.$$

Figure 3. Some dyslexic students prefer formulae typeset in a large sans serif font .

one course (on mathematical logic), student assignments were marked automatically, by a computer (I used an early version of TARSKI'S WORLD [[12]]; an improved version is now available as [[13]]). In another course, on number theory and cryptography [[14]], students (whose identities were hidden online under aliases) were invited to attack each others' ciphers, and the ensuing fights provided the most rigorous form of assessment. I had a chance to observe pedagogical and psychological *mise en scènes* evolving from the constraints of a technological set-up.

Studies of students' attitudes to IT already exist, but yet I could not locate studies that are sufficiently detailed and subject specific—although [[15]] provides a useful survey. However, I believe that mathematics students differ from the general student population: in mathematics, students' attitudes to IT are much more diverse and complex. It sounds paradoxical, but quite a few mathematics students dislike computers (for otherwise they would study computing and computer science).

At the other end of the spectrum, we have the beginner hackers (or "script kiddies"); for whom the IT solutions offered at the university are primitive and boring. (In mass media, the term "hacker" has negative connotations; among computer enthusiasts, "hacker" is a term of respect, it means someone who can do clever tricks ("hacks") with computers and software [[16],[17]].) Also we have among our students a number of adrenalin charged gamers who cannot wait, for purely physiological reasons, if a VLE hangs for a few seconds. Comparing mathematics with other disciplines, I make an educated guess that gamers can be found, say, in Humanities—but not that many script kiddies.

In the cryptanalytic battles which I mentioned above, students revealed their psychological positions in their choice of aliases. Over the years, I had in my class several girls who called themselves Piglet. Alas, the outcome of an encounter between Piglet and, say, Darth Vader (a gamer) was entirely predictable. Interestingly, Tigger (another girl and a friend of Piglet) ferociously and successfully fought back. However, Tigger had fallen under a sustained onslaught from Xterminator (a script kiddie) who, unsatisfied by tools provided in the course, downloaded from the Web and compiled an industrial strength C++ code. Of course, my primary duty as a teacher was to give Piglet and Tigger not only technical help, but also moral support and encouragement.

These experiences made me sensitive and attentive to personal attitudes of my students to computers and IT,—and made me to believe that there is no one universal solution that suits all students.

What we want: windows in mathematical worlds

Let me formulate in one word the shared key feature of IT that finds modern uses in direct teaching of mathematics: this word is *virtualisation*. A computer is useful if it creates a new (virtual) reality that cannot be created by other means. [[18]] In mathematics, the word "reality" includes the ideal Platonic world of mathematical objects and structures. MATLAB [[19]], MAPLE, MATHEMATICA—mathematics software packages widely used in undergraduate teaching—are windows into this Platonic world. As a rule, software that provides such windows needs a powerful mathematical engine. MATLAB, MAPLE, MATHEMATICA and statistics packages such as SPSS and R are not just toys for learning—they are professional research tools; mastering them is a valuable transferable skill for graduates seeking employment in mathematics-intensive industries.

I can give less known and more specialised examples, like the already mentioned TARSKI'S WORLD—an expertly crafted courseware package for learning mathematical logic, and the wonderful visualisation and experimentation tools for elementary geometry, CINDERELLA [[20]] and GEOGEBRA [[21]].

Assessment of mathematics learning software inevitably involves a mathematical characterisation of its built-in mathematical world. For example, it matters that the interface language of TARSKI'S WORLD is "interpreted"—and serious implications of this fact for teaching logic with TARSKI'S WORLD had been pointed out in one of the first reviews of the package [[22]]—

[12] J. Barwise and J. Etchemendy, The Language of First-Order Logic: Including the IBM-compatible Windows version of Tarski's World 4.0. Cambridge University Press, 1993. ISBN-10: 0937073903; ISBN-13: 978-0937073902.

[13] D. Barker-Plummer, J. Barwise and J. Etchemendy, Tarski's World. Chicago University Press, 2008. ISBN-10: 1575864843; ISBN-13: 978-1575864846. http://ggww.stanford.edu/NGUS/tarskisworld/.

[14] This my course was close in spirit to the book of P. J. Giblin, Primes and Programming, Cambridge University Press, 1993. ISBN-10: 0521409888; ISBN-13: 978-0521409889. Instead of bespoke code in PASCAL, I was using off-the-shelf routines of MATLAB.

[15] Learner acceptance of on-line learning and e-learning, http://wiki.alt.ac.uk/index.php/Learner_acceptance_of_on-line_learning_and_e-learning.

[16] D. Thomas. Hacker Culture. University of Mineapolis Press, 2002.

[17] C. Legg, Hacking: The performance of technology? Techné 9 no. 2 (2005), 151–154. Available at http://waikato.academia.edu/CathyLegg/Papers/209879/Hacking--The-Performance-of-Technology-.

[18] A metaphor of mathematics as a virtual reality game is perhaps best formulated by Anna Sfard (A. Sfard, Symbolizing mathematical reality into being—or How mathematical discourse and mathematical objects create each other. In Symbolizing and Communicating: Perspectives on Mathematical Discourse, Tools and Instructional design (P. Cobb et al., eds.). Mahwah, NJ: Erlbaum, 1998, pp. 37–98.).

[19] A list of available learning resources—far from being complete—can be found at http://www.mathworks.com/matlabcentral/linkexchange/?term=tag:"mathematics".

[20] An earlier version of CINDERELLA can be downloaded for free from http://cinderella.de/tiki-index.php?page=Download+Cinderella+1.4&bl.

[21] GEOGEBRA is free and open source, http://www.geogebra.org/cms/.

[22] W. Hodges, *Review of J. Barwise and J. Etchemendy,* TARSKI'S WORLD *and* TURING'S WORLD, Computerised Logic Teaching Bulletin 2 (1) (1989) 36–50.

written in 1989!

If you wish to get hands-on experience, I invite you to have a look at CINDERELLA and GEOGEBRA—they are available for free, and their uncluttered minimalistic interfaces provide for an immediate usability. The next two paragraphs use Cinderella and GEOGEBRA to give a sample of a mathematician's approach to probing and testing the software; they can be skipped in the first reading.[23]

> It is interesting to compare the behaviour, in CINDERELLA and GEOGEBRA, of a simple interactive diagram: two intersecting circles of varying radii and the straight line determined by their points of intersection. In GEOGEBRA, when you vary the radii or move the centres of the circles and make the circles non-intersecting, the line through the points of intersection disappears—exactly as one should expect. In CINDERELLA, the line does not disappear, it moves following the movements of the circles, always separating them; when circles touch each other and start to intersect again, the line turns to be, again, the common tangent line of two circles or, in the case of two intersecting circles, the line through the points of intersection. (The line is called the *radical axis* of the two circles.)
>
> To a mathematician, the behaviour of this diagram suggests that the underlying mathematical structure of GEOGEBRA is the real Euclidean plane. In CINDERELLA, the underlying structure is the complex projective plane; what we see on the screen is just a tiny fragment of it, a real affine part. The radical axis of two non-intersecting circles is the real part of the complex line through two complex points of intersection. The intersection points of two real circles are complex conjugate, the line is invariant under complex conjugation and therefore is real and shows up on the real Euclidean plane. For a mathematician, this is a strong hint that CINDERELLA could work better than GEOGEBRA in accommodating non-Euclidean geometries: elliptic and hyperbolic (the Lobachevsky plane) since they happily live in the complex projective plane.

I have already mentioned CALL, Computer Assisted Language Learning, as an interesting development outside mathematics which in some aspects is parallel to CAL in mathematics. Software packages for learning languages frequently involve powerful engines that support a "virtual interlocutor", a software device that listens to the learner, recognises and analyses the learner's speech, corrects errors and gives feedback. Creation of such tools would be impossible without decades of development of computational and mathematical linguistic.

The unity of research and teaching

One interesting feature of MATLAB, MAPLE, MATHEMATICA and SPSS is that they were originally designed and developed for research purposes and only later fed into university teaching—mostly by mathematicians who transferred to their teaching the skills developed in their research. It was the mathematics research community who acted as a driver of technological change in mathematics teaching. This example, even taken on its own, demonstrates the futility of erecting a fence between mathematics research and mathematics teaching.

The situation with specialised teaching-only software packages is even more instructive. Returning to one of my case studies, TARSKI'S WORLD, I wish to comment that one of its authors—and the initiator of the project—was Kenneth Jon Barwise, a prominent mathematician, philosopher and logician.

Development of TARSKI'S WORLD and other programs that became part of the courseware package *Language, Proof and Logic* [24]: FITCH, BOOLE, GRADE GRINDER, required not only a pioneering re-assessment of methodology of teaching mathematical logic [25], but also the creation of a new direction in mathematical logic itself, [26], *heterogeneous reasoning*, which formed the core of the computer algorithms implemented as courseware. The first reviewers of TARSKI'S WORLD [27, 28] were fully aware of mathematical difficulties that its authors had to overcome.

The work of Jon Barwise and his collaborators is a manifestation of a phenomenon specific to mathematics: the central role of *didactic transformation*, that is, *mathematical reworking* of teaching material into a form suitable for students' consumption. The term *transformation didactique* was coined in 1852 by French philosopher Auguste Comte [29] and is well known in French education studies [30] but remains unused in English-language literature on education. Hyman Bass, a prominent mathematician and a champion of mathematics education, picked up from his French colleague Jean-Pierre Kahane [31] the term "didactic transformation" and an explanation of its role in relations between mathematics and mathematics education:

- In no other living science is the part of presentation, of the transformation of disciplinary knowledge to knowledge as it is to be taught (*transformation didactique*) so important at a research level.

[23] For the sake of formal completeness I have to mention other elementary geometry pachages: CABRI `http://www.cabri.com/` and THE GEOMETER'S SKETCHPAD `http://www.dynamicgeometry.com/`.

[24] J. Barwise and J. Etchemendy, Language, Proof and Logic. CSLI Publications, 2003. Distributed by the University of Chicago Press.ISBN 157586374X.

[25] J. Barwise and J. Etchemendy, Computers, visualization, and the nature of reasoning, in The Digital Phoenix: How Computers are Changing Philosophy (T. W. Bynum and J. H. Moor, eds.). Blackwell, 1998, pp. 93–116.

[26] S.-J. Shin, Heterogeneous reasoning and its logic, The Bulletin of Symbolic Logic 10 no. 1 (2004) 86–106.

[27] G. Boolos, Review of Jon Barwise and John Etchemendy, Turing's World and Tarski's World, J. Symbolic Logic 55 (1990) 370–371.

[28] D. Goldson and S. Reeds, Using programs to teach Logic to computer scientists, Notices Amer. Math. Soc. 40 no. 2 (1993) 143–148.

[29] A. Comte, Catéchisme positivist. 1852.

[30] Y. Chevallard, La transposition didactique—Du savoir savant au savoir enseigné. La Pensée sauvage, Grenoble, 1985.

[31] H. Bass, Mathematics, mathematicians, and mathematics education, Bull. Amer. Math. Soc. 42 no. 4 (2005), 417–430.

- In no other discipline, however, is the distance between the taught and the new so large.
- In no other science has teaching and learning such social importance.
- In no other science is there such an old tradition of scientists' commitment to educational questions.

You can read more about didactic transformation in my paper [32] or in Chapter 9 of my book *Shadows of the Truth* [33]. Here I will add only that, in Barwise's case, didactic transformation took the form of serious cutting-edge mathematical research in logic which was than fed into state-of-the-art software development.

A parallel universe: computer assisted language learning

I have already mentioned CALL, Computer Assisted Language Learning, and had a chance to say that creation tools for CALL would be impossible without decades of development of computational and mathematical linguistics. CALL benefited from the long standing interest and attention of computer scientists to linguistics which started in 1950-s and 1960-s, when machine translation of human languages was a Holy Grail of rapidly developing computer science. Grammar correcting software for written exercises in foreign languages which started to appear in 1980-s was an out-spun of the earlier attempts to develop natural grammar parsers for machine translation. Speech recognition modules were originally developed for wider non-academic applications, which, in their turn, had generous funding from the industry. A frequent complaint about language learning software is that it is expensive; this is not surprising, given the huge cost of development.

Paradoxical economics of education

So, mathematicians have developed, and systematically use, specialist software in direct teaching of mathematics—and find it very useful.

However, as means of delivery of mathematics teaching, IT and e-learning technologies have so far been unable to meet our expectations. There are several reasons for this.

One reason is that our expectations are high. Due to the level of sophistication already achieved, say, in MATLAB / MATHEMATICA / MAPLE or in TEX / LATEX, mathematicians' demands for functionality of IT are high and are not met by many software packages and VLEs currently promoted in British universities.

At a neurophysiological level, teaching / learning mathematics is a communication between two brains. It is best done one-to-one, or in a small group. Large class lectures are an unhappy compromise with economic necessity. From a pedagogical point of view, the right alternative to a large class lecture is not streaming-on-demand of video recordings; the true alternative is a small class lecture. Unfortunately, this alternative in most cases is financially infeasible.

There is a need to assess the efficiency of particular methods of teaching not only from the pedagogical, but also from a socio-economic point of view. Of course, "generic" technologies are very tempting to policy-makers because of their promise (mostly unrealistic) of economies of scale. But it would be wrong to reduce the all-important discussion of learning and teaching to deciding the choice of the cheapest variety of margarine as a substitution for butter.

And, last but not least, at the socio-economic level relations between mathematics and information technology are also paradoxical.

Mathematics, by its nature, is an open source phenomenon. A powerful formulations of this principle belongs to Joachim Neubüser, the initiator and leader of the GAP project, perhaps one of the most successful community projects in experimental mathematics [34].

> You can read Sylow's Theorem and its proof in Huppert's book in the library without even buying the book and then you can use Sylow's Theorem for the rest of your life free of charge, but—and for understandable reasons [...]—for many computer algebra systems license fees have to be paid regularly for the total time of their use. In order to protect what you pay for, you do not get the source, but only an executable, i. e. a black box. You can press buttons and you get answers in the same way as you get the bright pictures from your television set but you cannot control how they were made in either case.
>
> With this situation two of the most basic rules of conduct in mathematics are violated: In mathematics information is passed on free of charge and everything is laid open for checking. Not applying these rules to computer algebra systems [...] means moving in a most undesirable direction. Most important: Can we expect somebody to believe a result of a program that he is not allowed to see? [35]

It is almost a rule that open source software systems are friendlier to mathematics; perhaps this could be explained by the social and cultural background of the open source movement. A good illustration of this principle can be found in a comparison between MOODLE, a free open source VLE (it

---

32 A. V. Borovik, Didactic transformation in mathematics teaching, in The Teaching-Research Interface: Implications for Practice in HE and FE (Muir Houston, ed.). Higher Education Academy Education Subject Centre, Bristol, 2008, pp. 30–35. ISBN 978-1-905788-81-1.

33 A. V. Borovik, Shadows of the Truth: Metamathematics of Elementary Mathematics. A draft version is available for free download from `http://www.maths.manchester.ac.uk/~avb/ST.pdf`.

34 GAP – Groups, Algorithms, Programming – a System for Computational Discrete Algebra, `http://www.gap-system.org/`.

35 J. Neubüser, An invitation to computational group theory. Invited talk at the conference 'Groups St Andrews' at Galway 1993. Available in DVI: `http://www.gap-system.org/Doc/Talks/cgt.dvi` and PostScript: `http://www.gap-system.org/Doc/Talks/cgt.ps`.

provides for a decent rendering of LaTeX) and proprietary VLEs, some of which are completely unfit for use in mathematics courses.

Developers of quality proprietary software for mathematics and statistics (like MATLAB / MATHEMATICA / MAPLE, SPSS) have taken reasonable care to allow the users a sufficient degree of freedom in tinkering with the interface (and, at least in the case of MATLAB—with the computational core, too—MATLAB smoothly incorporates bespoke Fortran and C code). Also, MATLAB / MATHEMATICA / MAPLE allow the export of results (both symbolic and graphic) in formats directly usable in TeX / LaTeX documents.

And R, a very popular statistics package, is a GNU licensed open source product.

TeX, the true and unsurpassed masterpiece of the art of computer programming, is faced with a strange fate: it somehow does not show up on the radar of promoters of IT for HE. I believe this has a very simple explanation: TeX is free—thanks to the generosity of Donald Knuth—and open source. It exists like the air that we breath. For that reason TeX remains unadvertised and is not promoted, and therefore goes unnoticed by university administrators who make decisions about the acquisition of IT products. Paradoxically, these are the same administrators who hold the purse strings and are apparently on the quest for the cheapest IT solutions. I conjecture that MOODLE is also disadvantaged by being free, not promoted by vested commercial interests, and therefore may be less visible in the market.

We have to make free open source options visible—this will allow them to compete with commercial for-profit products. The problem is wider and concerns not only software and IT, but also textbooks.

From the next academic year, I will be teaching one of my lecture courses using an open source GNU licensed textbook [36]. Besides pedagogical reasons, my decision is motivated by new functionality provided by open source textbooks: it gives, for example, a possibility of global changes in the text (say, a uniform change of notation over the entire textbook). The nature of mathematics teaching makes this kind of open source functionality very useful.

It is a social imperative of our challenging times: open source teaching suits publicly funded universities best. But, because it is not promoted by commercial interests, it needs its champions. The textbook which I am planning to use finds an unexpected champion in Arnold Schwarzenegger. The book is endorsed by the Free Digital Textbook Initiative run by California Learning Resources Network [37]. CLRN is funded by the state of California. California is experiencing financial difficulties, and the webpage of the Free Digital Textbook Initiative proudly displays a message from Governor of California Arnold Schwarzenegger:

> This initiative will ensure our schools know which digital textbooks stand up to California's academic content standards – so these cost-effective resources can be used in our schools to help ensure each and every student has access to a world-class education.

The state of Texas recently launched a similar initiative [38]; together, the states of California and Texas control the market of high school textbooks in the USA. Also, public depositories of open source textbooks like CURRIKI [39] are becoming more prominent and influential.

Shopping List

Mathematicians do not want to work in isolation from the rest of the IT learning community; there are a number of issues (like support to users with disabilities) that need a coordinated effort.

Here is a brief list of our concrete wishes. It was suggested by my colleagues who read earlier versions of my notes. Any help and advice from the IT learning community would be warmly appreciated.

- Virtual Learning Environments:
  - Support for, and interfacing with, MATLAB, MATHEMATICA, MAPLE, SPSS, R.
  - Support for symbolic input and output in MATLAB, MATHEMATICA, MAPLE, and import of

[36] J. Hefferon, *Linear Algebra*, available for free download from `ftp://joshua.smcvt.edu/pub/hefferon/book/book.pdf`.

[37] `http://www.clrn.org/fdti/`.

[38] A. Vance, $ 200 Textbook vs. Free. You Do the Math. New York Times, 31 July 2010. `http://www.nytimes.com/2010/08/01/technology/01ping.html?_r=1&emc=eta1`.

[39] `http://www.curriki.org/xwiki/bin/view/Main/WebHome`.

mathematics graphics produced by these packages.
  - In SPSS and R—input and output of data files, import of tables.
  - As it was already explained before, VLEs are unusable in mathematics learning and teaching if they do not support for TEX / LATEX.
  - One of the benefits cited for VLEs is the ability for students to engage in discussions. They cannot do this if we have barriers to getting mathematics into a machine. Thus support for TEX / LATEXis essential.
- Provision for visually impaired students. Screen readers do not work with mathematics!
- Online computer-aided assessment. This is a big issue for those of us who want to formatively assess 350+ size classes without using very bland questions. In particular, assessment systems should allow easy and unconstrained entry of mathematical formulae and be able to interpret their meaning. Some obstacles to that are discussed by Sangwin [[40]].
- And last but not least—the role and status of free and / or open source software, courseware and textbooks deserve a thorough discussion.

Disclaimer and acknowledgements

Needless to say, all opinions expressed here are of the author and no-one else. However, I wish to express my thanks to my colleagues from the Research Committee of the Association for Learning Technology [[41]] and from the Education Committee of the London Mathematical Society for very useful discussions. My special thanks go to Chris Budd, David Mond, Morag Munro, Chris Sangwin and Seb Schmoller.

An earlier and shorter version of this paper has been published in *ALT News Online* [[42]].

[40] C. Sangwin, Assessing elementary algebra with STACK, 2006. http://www.open.ac.uk/cetl-workspace/cetlcontent/documents/4607d31d634fd.pdf.
[41] http://www.alt.ac.uk/.
[42] http://newsletter.alt.ac.uk/4edkkzb138s.