
Didactical reduction versus references: How to better teach \LaTeX

Sarah Lang

Abstract

This paper discusses didactical principles (such as didactical reduction) and their relationship to common modes of technical (and \LaTeX) knowledge transmission, mainly the genre of technical references and documentation. It leverages ideas from personal development for suggesting new, more didactically suitable modes of knowledge transmission for \LaTeX education. It discusses how teaching concepts (like didactical reduction) and common modes of knowledge transmission (such as references and documentation) could be reconciled in \LaTeX teaching contexts.

The problem: Informal knowledge about how to act in a learning setting and how to procure the knowledge one needs is often linked to different forms of privilege. Using references to acquire knowledge requires a relatively big amount of tech literacy or familiarity with the medium. People with non-tech backgrounds often lack this tacit knowledge. This paper proposes ways to provide different sorts of \LaTeX education to different types of learners, using the efforts made on the \LaTeX Ninja blog as an example.

This paper is not an exact replica of the contents of the talk. The paper contains an example discussion of Kopka and Daly’s *Guide to \LaTeX* which came from the discussion after the original talk; and the original talk contains information about the \LaTeX Ninja blog and other examples which will not be discussed further in the paper.

1 What is didactical reduction?

Didactical reduction is the art of reducing unnecessary detail.¹ By this, I don’t mean at all that the difficulty level of materials necessarily needs to be reduced. Dumbing things down so much that there is no content left is not good practice either, and reducing information to the purely superficial is not at all what I mean. One can present difficult material but still leave out detail to make the contents more digestible didactically. This entails that producers of didactical materials need to become aware of their tacit knowledge and “tech privilege” before writing up learning materials. For example, passive knowledge of packages becomes a dominant issue when numerous packages are included in tutorials — packages which aren’t part of the topic and could have been left out. This overcomplicates the subject and provides examples which are far more than minimal.

2 Different starting points: “Tech privilege” and tacit knowledge in the \LaTeX learning setting

I argue there exists “tech privilege” which is a form of privilege analogous to other forms of privilege, such as male or white privilege, but concerns the subject matter of technology. It is known nowadays that we are taught lots of tacit knowledge implicitly according to certain stereotypes. For example, studies involving cross-dressing in small children, e.g., female toddlers dressed in stereotypically male clothes and vice versa, have shown that adults are conditioned societally to pass the “female passing” children dolls and such, whereas they give “male passing” children stereotypically “male connotated” toys, which tend to teach more logic, technical and mathematical skills. Such early conditioning causes children’s “talents” to develop much more in the area favoured by society, generating an apparent divide that we all know: It indicates that men are supposed to be better at logic and maths, whereas women are better at languages — a wholly incorrect stereotype according to science!

Children (and adults too) thus get more praise for engaging in stereotypically gender- and class-suited activities. They will also be trusted more to be successful and apt at those. It is known that praise and expectations, as in the so-called Pygmalion or Rosenthal effect have an influence on learning.² This means that those who already have a certain “tech privilege” or are in a socially accepted position as a tech nerd have a much easier path learning tech knowledge than those who lack this “tech privilege”. Also, those with “tech privilege” will implicitly learn certain skills pertaining to tacit knowledge — say, knowing how to post a successful question on Stack-Overflow or the simple fact of what a Lorem Ipsum is.

A person without a background in technology might never have come across such concepts which makes their learning process all the more difficult if learning resources don’t provide this information. To those who aren’t already part of a larger tech community, typical technical modes of knowledge transmission, such as references and documentation, are largely inaccessible.

3 The ability to abstract and the “superuser mindset”

Apart from that, references often require a large amount of abstraction which is also a skill outsiders might not yet have. They don’t know how to sum up their problem in an abstracted way. For example, in the talk, I mentioned that a newbie might not realize

their problem is spacing. They might just think, “I want this page to look like I know it from MS Word.” They also might not be familiar with non-GUI applications, having seen only applications which are extremely targeted to fulfilling a user’s (presumed) needs, giving them a user experience which requires no background knowledge at all. Moving from a passive user experience to — what we could call — a “superuser perspective” is a big step which is not at all an obvious or easy path for a newbie without a tech support network or in-person teaching.

4 The difference between reference and didactical reduction, with the example of Kopka and Daly’s *Guide*

I was asked about my opinion on the generally well-liked reference of Kopka and Daly.³ Therefore, I have chosen it as an example on which I can illustrate my point about references versus didactical reduction. I was asked whether I would recommend it and my opinion is: as far as references go, this a go-to resource — but it is still a reference. My argument is that for effective and efficient teaching, we need to detach teaching resources and technical reference documents. Kopka and Daly explicitly say their guide is a mixture of both, which is exactly what I take issue with.⁴ Thus, I would not fully recommend it. This does not mean I don’t like the *Guide* — I just think it was written under a different paradigm from the one I invite the (L^A)T_EX community to use in the future.

Let’s examine this in the context of the *Guide*. The first thing we notice: it’s over 600 pages long. So, either you’re not expected to read it all (i.e., it is a reference), or it’s going to be a long read. I think it’s a good beginner’s reference but I also find there’s an inherent problem with references. I would expect a beginner’s text to only cover things that they will ubiquitously need, so that’s either “theory” or a focus on the absolute basics. For example, if you know how a document class works, why would a reader be shown the example of letters which they might not even need? If they wanted to learn about them, they could just look it up with a quick web search. Thus, I think there would be lots of room for cutting out detail.

A suggestion for improvement would be to perhaps write a beginner’s part with things actually everybody will need and then diverging into “specializations”.

For example, as has been addressed elsewhere during the TUG 2020 conference, T_EX people tend to assume everybody needs math. But I never need

math. These little “divergences” might be starting points to find elements which a general reference should exclude or put into an appendix. Kopka and Daly for example put “Programming with L^AT_EX” into the appendix which I think should go into the main section, just like the text on “Error messages”. On the other hand, in my opinion, Kopka and Daly should have put lots of the material presented in part II “Beyond the Basics” into the appendix. A resource following the paradigm of didactical reduction would put all the relevant background information (“theory”) into the main part and move the information about packages or special purpose document classes into the appendix.

References are not usually didactically suitable. Apart from the format being less than optimal for teaching purposes, references also lack certain elements which are relevant to teaching, i.e., explaining theoretical backgrounds.

5 Leveraging concepts from personal development: Different goals

In theory, learners from every different skill level and background would each need their own learning resource, which is why I suggest that complete beginners should get extremely short resources with lots of practical applications, such as my attempt at a “3 minutes to L^AT_EX” tutorial.⁵

On the other hand, it can often be difficult to find information on the skills which lead to true advanced L^AT_EX skills or L^AT_EX mastery — because this also includes “soft skills” such as knowledge about the publishing industry, an understanding of typography or how to monetize your L^AT_EX skills. Leaning on different concepts of skill acquisition of expertise from personal development, I have argued that mastery might not be for everybody, which is why we shouldn’t make it the goal of learning.⁶ Josh Kaufman’s *The First 20 Hours: How to Learn Anything . . . Fast*, for example, points out that with popular skill acquisition discussions like the so-called “10,000 hour rule”, the implicit assumption was made that everybody wants to achieve real mastery.⁷ Kaufman argues that is quite the misconception; in reality, many people just want the skill level equivalent of being able to strum along with a four-chord song on the guitar. Certainly some will want to achieve mastery, but by no means all.

The (L^A)T_EX community should acknowledge this diversity not only of background in different learners, addressed in the discussion on tech privilege and tacit knowledge, but also the diversity of possible goals a (L^A)T_EX learner might have in mind.

Notes

¹ I have addressed the topic in the following blog posts:

latex-ninja.com/2018/11/05/on-didactical-reduction-especially-in-the-dh

latex-ninja.com/2019/01/12/didactical-reduction-part-ii

latex-ninja.com/2019/04/14/improve-your-teaching-10-simple-tricks

² The effect describes a self-fulfilling prophecy that learning effects get better with higher expectations and worse with lower expectations.

³ Helmut Kopka and Patrick Daly, *A Guide to L^AT_EX*, 4th edition, Addison-Wesley Professional, 2003.

⁴ “This *Guide* is meant to be a mixture of textbook and reference manual.”; page 10, Kopka and Daly.

⁵ latex-ninja.com/2018/12/11/jumpstarting-learn-latex-in-3-minutes

⁶ In the talk, I also brought up the concept of the Minimum Effective Dose (MED) from Tim Ferriss’ *The 4-Hour Chef*. This was discussed in our earlier *TUGboat* article: Sarah Lang and Astrid Schmörlzer, “Noob to Ninja: The challenge of taking beginners’ needs into account when teaching L^AT_EX”, *TUGboat* 40:1, pp. 5–9, <https://tug.org/TUGboat/tb40-1/tb1241lang-newbie.pdf>.

⁷ Malcolm Gladwell’s book *Outliers* made the so-called 10,000 hour rule popular. However, please note that this was an incorrect interpretation of research done by Anders Ericsson. The best resource about the science of expertise and skill acquisition using deliberate practice is Ericsson’s *Peak: Secrets from the New Science of Expertise*. A commenter also suggested the book *Range: Why Generalists Triumph in a Specialized World* by David Epstein.

◇ Sarah Lang
[the.latex.ninja \(at\) gmail dot com](mailto:the.latex.ninja@gmail.com)
<https://latex-ninja.com>