

Preface

With \LaTeX being a voluntary effort,
it seems quite appropriate that
TLC also stands for “tender, loving care”
(Concise Oxford Dictionary)!
David Rhead, 1994

I have now been involved in computer based typesetting for nearly four decades, three of them as the technical lead for the development of \LaTeX . During that long period there have been impressive technical advances in many different areas.

When I started there was no Internet to speak of — there were no browsers and there was no World Wide Web as we know it today. To book a hotel on my first trip to California to meet with Leslie Lamport, I had to resort to a travel agency that used fax machines to arrange the trip; on the flight I was served free alcoholic drinks (bad idea); and my computer at home was an Atari with two floppy drives (younger people probably only know these as the strange “save icon” in many software programs and perhaps have wondered what that represents) and an impressive external hard disc with 100mb of storage (that cost me a fortune at that time).

However, already back then \LaTeX had existed for some time and worked fine, though a lot of today’s functionality was unavailable or, even if available, impossible to use, because computer processing speed was simply too slow.¹ As explained in more detail in the history section in Chapter 1, most of our enthusiastic ideas back then for a new and improved \LaTeX were simply two decades too early, and while we had a fully working first version of the L3 programming layer in the early nineties our

¹The first simple \TeX documents I produced on a large university mainframe took about half a minute per page — you could literally watch the progress as [*wait*, 1, *wait*,], *long wait*, ... — and we still thought it was great and fast.

users would have died of caffeine consumption waiting for the results of processing their documents if we had dared to inflict it on them.

This all has changed since that time and today my smartphone is faster than the mainframe power available in the nineties. As a result, many new packages appeared over time and a lot of our dormant ideas and concepts envisioned in 1990 were finally integrated into L^AT_EX on the memorable day of February 2, 2020.

Since then, this programming environment has been used by the L^AT_EX Team to offer new functionality and also by many package authors developing new packages. All these developments — the recent as well as the older — are covered in this book.



The Companion editions — setting the standard for a dozen years each

When Michel, Alexander, and I wrote the first edition of *The L^AT_EX Companion* [56] in 1993, we intended to describe what is usefully available in the L^AT_EX world (though ultimately we ended up describing the then-new L^AT_EX 2_ε standard and what was useful and available at CERN in those days). As an unintended side effect, this first edition *defined* for most readers what should be available in a then-modern L^AT_EX distribution. Fortunately, most of the choices we made at that time proved to be reasonable, and the majority (albeit not all) of the packages described in the first edition are still in common use today.

During the following decade the *Companion* (nicknamed the “doggie book” because of its cover) became a core resource for many L^AT_EX users, with several reprints and translations into German, Japanese, and Russian.

Our approach was to provide comprehensive coverage for typical L^AT_EX documents so that for most users the *Companion* would serve as the only reference needed to get “the job” done. More esoteric package features or features still under development were not described. Instead, pointers to the package documentation were given if we thought such a feature was worth mentioning. This approach worked well, so at the turn of the millennium one reviewer wrote, “while the book shows its age, it still remains a solid reference in most parts”.

The second edition in the new millennium ...

Nevertheless, much had changed and a lot of new and exciting functionality had been added to L^AT_EX during that decade and it became clear that a revised edition was necessary. This second edition [145], published in 2004, saw a major change in the authorship: I took over as principal author (so from then on I am to blame for all the faults in the *Companion* editions) and several members of the L^AT_EX Project Team joined in the book’s preparation, enriching it with their knowledge and experience in individual subject areas.

We ended up rewriting 90% of the original content and adding about 600 additional pages describing impressive and useful new developments. As a result, the second edition was essentially a new book — a book that we hoped preserved the positive aspects of the first edition even as it greatly enhanced them, while at the same time avoiding the mistakes we made back then, both in content and presentation (though, of course, we made some new ones). From the reception in the user community, I think it is fair to say that we largely succeeded — in fact, that book served even longer as a useful resource.

However, a decade or more is an awfully long time for a technical book, even given the longevity and stability of \LaTeX and the *Companion's* approach of describing a coherent and well-established set of packages. So in 2017 I started discussing with Kim Spenceley (my new editor at Addison-Wesley/Pearson after Peter Gordon's retirement) plans for a third edition of *The \LaTeX Companion*. One question to solve up front was that of authorship. Initially, it looked as if I would have to do any necessary work all by myself this time, because none of the previous co-authors was available to help for one reason or another, making it a very daunting task indeed.

... and nearly two decades later, the third

Fortunately, this impression was wrong! In the end I got great help from Ulrike Fischer, who wrote Chapters 15 and 16, the sections on `hyperref` and `tikz`, and helped with numerous tasks during the production of this edition.

Javier Bezos and Johannes Braams took on the task of revising Chapter 13 on localizing documents, and Joseph Wright helped with describing `siunitx` and the section on source control support. Thanks to all of them — without their help the book would have been much more difficult to finish.


Furthermore, Nelson Beebe kindly offered to read *all* chapters, checking them for accuracy as well as doing a first pass on copyediting. He provided numerous suggestions for improvements and I cannot thank him enough for undertaking that enormous task! The professional copy editor and the two proofreaders found additional boo-boos of mine, and I then found a few they missed while entering their corrections. I am sure our readers will find even more — it is a never-ending task, but we all did our best and, on the whole, I think we delivered a solid result.

Big thanks to our volunteer copy editor Nelson!

The new edition

Initially, when I discussed plans for a third edition of *The \LaTeX Companion*, I expected the need for a large number of updates to the existing material, but not many additions. Thus, my naive estimate was that the book would perhaps grow by 10–15%.

However, after researching in depth the new material that had been developed since 2004, it became crystal clear that to remain faithful to the core promise of the *Companion* — to be a solid reference for the majority of \LaTeX users to get their work done — we had to include a much larger amount of new material:

 *What's in it for you?*

- Descriptions of highly useful, large-scale packages that appeared in the meantime or were substantially updated in the last decade, e.g., `biblatex`, `fontspec`, `hyperref`, `mathtools`, `siunitx`, `tcolorbox`, `unicode-math`, and `tikz`, to name a few.
- A larger number of smaller packages that cover new ground and are useful for day-to-day work or for specialized (but not too esoteric) tasks.¹
- Two new chapters on the exciting possibilities offered by using high-quality fonts for text *and* math — yes, \LaTeX is no longer restricted to Computer Modern fonts or a few PostScript fonts that were set up for use with \LaTeX in the nineties.

You can now choose from a large number of high-quality, free fonts for both text and math; the only serious remaining problem is finding the ones you like. These chapters help with that, by showing samples of more than one hundred

¹Give or take the odd exception, e.g., `sillywalk`, which I found just too lovely to bypass.

Big thanks to Adam, helping me with my two favorite “coffee table book” chapters

text fonts and more than forty alternative math font setups. I am very grateful for the help I received from Adam Twardoch (president of GUST, the Polish T_EX users group, and the designer of the Lato fonts) on this, who spent many hours with me during two BachoT_EX conferences, guiding me through the fonts available today and helping to select those of high quality for inclusion in the book.

- We also had to cover newer engine developments, e.g., the use of Unicode engines with L^AT_EX, across all chapters of the book. There are often subtle differences that you need to be aware of if you use these engines.
- Finally, there have been very important changes to L^AT_EX itself, which is undergoing a transformation that started in 2018, to keep it relevant in the years to come. Examples are the new hook management system for L^AT_EX, the extended document command syntax, and the inclusion of the L3 programming layer into the L^AT_EX format. All this is covered in the appropriate places — just take a peek at the term “L3 programming layer” in the index to see how much of the new material is already based on it.

In that sense the third edition is like the first: both have been written just after L^AT_EX itself had seen major changes and these exciting changes and additions are covered.

All this relevant information is now part of the new edition, but as a result we ended up with 1700 pages, not including the index — clearly too much to be printed as a single book that you can reasonably use as a day-to-day reference on your desk.

Two parts — one (virtual) book

For that reason the decision was made to split the book into two parts of roughly equal size and market them as a unit.¹ The chapter progression follows more or less the successful order of the earlier editions, starting with elements and concepts that you need quite often in nearly all documents (the first few chapters in Part I) followed by topics you also usually need in most documents but not necessarily all the time (remainder of Part I and most of Part II).

Of course, if you are a mathematician you might end up keeping Chapter 11 open all the time, but if your interest is typesetting novels or company reports, it might be the chapter least often touched.

Part II also contains three important appendices on core L^AT_EX commands for defining your own little commands or applications, one on resolving errors (not that you would make any, would you?), and one on getting further help if this edition is not answering your questions — unlikely I’m sure, but then who knows?



As David Rhead observed in the quotation at the beginning of the preface, TLC is not only an acronym for *The L^AT_EX Companion* but also stands for “tender, loving care” and this is most certainly an accurate description of the efforts and lifeblood poured into the works described on the pages of this book.

¹For technical and accounting reasons that still means three separate ISBNs, one for each part and one for the bundle. From my perspective this is far from perfect, but it is how the world of publishing and logistics works. It means that while it is theoretically possible to buy only one part, there is little sense in that — unless you have used it so much that it is worn out and you want a fresh copy.

There are millions of L^AT_EX users out there (the online service Overleaf alone reported ten million accounts in 2022) and most of them use L^AT_EX because they love its typesetting capabilities and its superior quality. With L^AT_EX being easily extensible, users often adapt L^AT_EX to their needs in new fields and for new applications, and some of them go one step further, packaging their solution and making it available to others — usually supporting it long after they have a private need for it.

This is why we now have close to 5000 packages for use with L^AT_EX on the Comprehensive T_EX Archive Network (CTAN) and why its catalogue lists nearly 3000 contributors. It is because of their dedication and “tender, loving care” that T_EX and L^AT_EX stayed relevant for nearly four decades, offering unsurpassed quality and, likely, continuing to do so for several decades to come.

*A standing salute
to all these dedicated
developers in the
L^AT_EX world!*



Looking back, it took roughly five years and several thousand hours to write this book, which sounds like an awfully long time and a huge effort — both of which are true — but the effort was rooted in the complexity and size of the task.

The first phase of the production was reading through the documentation of nearly 5000 packages available in today’s L^AT_EX distributions, classifying them according to functionality, usability, and correctness. This included testing all packages initially considered as candidates for inclusion, to see if their documentation actually matched reality (often it did not — mine included ☺) and to come up with relevant use cases and examples. Often, alternative solutions provided by different packages existed, in which case a more in-depth analysis was necessary to decide which packages to recommend. That phase took somewhat more than a year.

Research

After this initial survey I started with documenting the selected packages or, in the case of packages already in the previous edition, revising and updating the existing material, describing new functionality, or rearranging the documentation to provide better access.

Describe

Frequently, while thinking up useful examples, I found some errors in a package or in its documentation or identified valuable but missing functionality, in which case a discussion with the package author started. As a side effect, this process more than once messed up the text that I had already written about the package, because afterwards I had to account for the new or changed functionality that I had requested. Thus, in several cases I rewrote whole sections or provided new, improved examples when further features became available. However, a real headache proved to be the larger, complex packages that sometimes come with several hundred pages of documentation. The task in such a case is to work through all this material and figure out what from it is needed by the majority of our readers, describe it adequately, and point out which areas I had left out or only skimmed over.

Of course, in many other cases the situation was reversed; i.e., the package functionality was good, but the documentation difficult to understand or incomplete, so here the task was to provide a different, possibly expanded, and hopefully better description. In either case, a strong focus was on providing useful, ready-to-apply examples, of which this edition has more than 1550. They have all been handcrafted to cover the typical use cases and support the accompanying documentation in the book.

This phase took close to three years, which means roughly writing two pages per day if working without break — going at it each and every day, including weekends (and for large periods it was like this).


Produce

The final phase, which started in spring 2022, was to pass all the work, chapter by chapter, to the professional copy editors engaged by Pearson, enter their corrections, and then do the layout of the chapters.¹ Once a chapter was in its final form I passed it back to a proofreader, who verified that the corrections had been correctly entered (not always), followed by a final pass by another proofreader who checked the final version once more. In parallel, Keith Harrison and I worked through all chapters to compile a useful concept index. The overall process took nine months, and I completely underestimated the amount of work necessary for this, even though I should have known better from previous books.

Many thanks to Kim and Julie for making the book a reality

My sincere thanks to Kim Spenceley, my editor at Pearson, and Julie Nahil, my senior content producer, who steered me patiently through the whole process, putting up with my idiosyncrasies while keeping me on track, and at the same time making sure that my quest for quality was supported as much as possible.



Was it worth the effort? 

Maybe you are asking yourself was that worth it, in the days of the Internet, where you can search for almost anything in a matter of seconds or watch a video that explains how to do something?

My personal answer to that question is a clear yes, because while there is a huge amount of information out there, it is of very varying quality, from extremely good to horrendously bad, misleading, or even plain wrong. This makes it very difficult for a user to sort the wheat from the chaff and, as a result, this overflow of information is not helpful unless you get good guidance.

And this guidance, we trust, is what the *Companion* is offering you, by providing you with a curated set of packages covering all areas of document production, showing you suitable solutions to various problems, and explaining the limitations when using one or the other approach.

We hope that this new edition will become a good companion for you for many years to come — just like the previous editions in the last decades. If it turns out that we achieved that, it will certainly be something to be proud of.

Frank Mittelbach

November 2022

¹The nightmarish but also oddly satisfying task to lay out a book like this is described in the Production Notes at the very end of Part II.