What to teach about (LA)TEX

Jean-Michel HUFFLEN

Abstract

We summarise several experiments of introducing LaTeX to different audiences of students: in Computer Science, Mathematics, Physics, Chemistry, Digital Editions. Each time we started from different requirements. We show what common points for all audiences may be emphasised and how we adapted to features of specific audiences.

0 Introduction

We have been an Assistant Professor in Computer Science since the fall of 1994. In 2002, we were asked for the first time to teach LaTeX. Subsequently we have taught to many varied audiences. Hereafter, we propose a survey of such a variety and explain which experience we have got. First we summarise all our teaching units related to LaTeX, then Section 2 exposes the common points of our programmes. We went thoroughly into some specific features intended for Computer Science students in Section 3, whereas the main points of the programme given to other students are shown in Section 4. If the time allotted allows a teacher to go further than a short introduction, Section 5 points out some interesting directions. Some words about evaluating students are given in § 6.

1 Our experience

1.1 Teaching units

As mentioned above, we have been responsible for numerous classes about LaTeX [24]. All these teaching units have been based on *lab classes*, possibly preceded by *lectures* — in which case they include *demonstrations* —:

- in faculty of letters: my first teaching unit based on LaTeX — and including other points related to XML[1] — from 2002 to 2006, for 1st-year students in the Master of *Digital Edition* programme;[2]
- in faculty of science:
  - Mathematics students (2nd-year university degree), from 2005 to 2008, we wrote [12, 13] for these students;[3]
  - Computer Science students (1st- and 2nd-year university degrees), from 2013, [17] is devoted to these students;
  - Physics and Chemistry students (2nd-year university degree), from 2018: these two sections were sometimes grouped, sometimes separated, depending on the number of students.[4]

In addition, we helped people write big documents outside of 'formal' teaching units: PhD[5] dissertations in Computer Science, and in Medicine, nurse dissertations to get the final degree.

1.2 Requirements

We have been free to decide how our unit runs, provided that some requirements are included *a minima*:

- a focus on multilingual aspects for Digital Edition students, that is, some language-dependent typographic features, and an introduction to the babel package [27, Ch. 9];[6]
- an emphasis on the ability of developing *large* documents, possibly managed by *several* people, possibly reworked in several successive versions, for Computer Science students;
- a focus on (LA)TeX's math mode and some advanced features such as the amsmath package [25, Ch. 11], for Mathematics students;
- a deep introduction to bibliographical features — including the management of large bibliography data base (.bib) files using tools such as JabRef [20] — for Digital Edition, Physics and Chemistry students; let us remark that these students have had to do bibliographic research involving many documents.

1.3 Diversity

There cannot be a unique schema suitable for any teaching unit about LaTeX, beyond the bases of this typesetting system; that might be obvious. As a particularly accurate example, let us consider the way to do *bibliographies*. In France — because we have been facing students registered in a French university —:

- bibliographies are *alphabetically sorted* w.r.t. authors'/editors' names for most scientific documents, including PhD dissertations in Mathematics and Computer Science,
- but for the PhD in Medicine, they are sorted w.r.t. the *order of first citations* throughout the document,[7]

[1] E**X**tensible **M**arkup **L**anguage.

[2] Later, this curriculum was deeply revised and moved from Besançon to Belfort, under another title.

[3] Originally, they were available on the Web. Since the corresponding site was closed down several years later, the URL addresses are not given in this article's bibliography.

[4] These students were also given [17].

[5] ***Ph****ilosophiæ* ***D****octor*.

[6] At this time, engines like XeLaTeX [29] or LuaLaTeX [8] had not yet come out; the polyglossia package [26, § 13.7.2] had not, either. The babel package's present state is described in [26, § 1.3.6].

[7] That corresponds to *unsorted* styles, w.r.t. BibTeX's and biblatex's terminology. If such styles are used, some change

doi.org/10.47397/tb/46-2/tb143hufflen-teach

- they may also be implemented by *footnotes* — possibly abridged after the first citation —:
  - with a *global* bibliography section in Physics,
  - *no* global bibliography section for nurse dissertations.

More details about possible citation styles can be found in [26, Ch. 16]. Besides, we can observe that more and more LaTeX documents are designed using the duo biber/biblatex [26, §§ 15.2.2 & 16.7] as a modern bibliography processor. Within PhD dissertations, that is particularly true about Letters and Human Sciences. On the contrary, many Mathematics and Computer Science students are still using the 'old' BibTeX, probably because most Web sites for automatic submissions to conferences only process source files with LaTeX and BibTeX.[8]

2 Generalities

As reported in [2], *[it has been told] that secretaries should be given* short *texts, whereas we have to take care of* long *texts ourselves since we will have to rework them.*

Echoing this quote, we think the first contact of students with LaTeX should be performed by handling a middle-sized text. A document for an auto-learning class can start with a short LaTeX source text, a small text to be typed and compiled, but doing the same at the beginning of the first lab class encompassing a group of several students is useless and even counterproductive, as already mentioned in [1]. If you do that, you will spend a considerable amount of time fixing typing mistakes within command names or explaining what a compilation process is.[9]

So, we start from a document about 10 pages: first, students can compile it without any error, then they have to try some variants, e.g., two-column option, changing characters' sizes, ... Likewise, we very quickly introduce *new* commands, by means of the `\newcommand` command, as a kind of *semantic markup*. In particular, such new commands are very suitable for cases subject to change, e.g., a category of words that will be typeset using italicised characters or boldface fonts or even surrounded by '*' characters...; such examples should be present within the initial source file. Besides, this source file's text should include *several* sections: that will ease the introduction of the commands for *cross-references*: `\label` and `\ref`. So, the first demonstration can consist of first referencing a section, then adding a new section and studying how a reference to a previous one is changed.

With a powerful tool such as LaTeX, mastering its bases is needed, but learning to use its documentation is important too, due to numerous extensions, most implemented by packages. So students are given packages' names and are asked to look into LaTeX's documentation. In addition, our faculty's library bought several copies of the French translation [28] of the second edition of the LaTeX Companion [27]. Some examples of packages handled by students are shorttoc [3], numprint [9], multirow [25, § 6.7].

3 Teaching Computer Science students

All my classes have started from scratch, except for this academic year (2024–2025), for Computer Science students. In France, at regular time intervals, university *curricula* have to be revised and assessed. Within this framework and some reorganisation in 2024, the teaching unit about LaTeX, initially designed for 1st-year university students in Computer Science, moved to the 2nd-year programme. As a consequence, 2nd-year students of this academic year were 1st-year students of last year. So, we decided to teach advanced features to them, even if this was a *one-shot*, as explained in [18].

Since they were Computer Science students, we asked them to program some examples using the Lua language [19]. That was the first time they practised a *pair* of complementary programming languages: TeX for typesetting texts, Lua for other calculation.[10] They can observe that TeX's commands are very suitable for formatting directives — a very specialised language — but using it for programming purposes may be tedious or even nightmarish [11, 30]. We introduced these students to LuaLaTeX and the fontspec package [25, § 9.6] at the same time.

Another interesting point was to use TeX as an *old-fashioned* language, as sketched in [11]. That is not a pejorative point of view, but rather a comparison of the techniques used within TeX and more modern implementations. For example, modern students practise modern programming languages based

within a document's text may induce an order change within the bibliography. Besides, they are not really suitable for large bibliographies. So many associations of medicine students campaign for changing this convention.

[8] The same point slows down the adoption of modern engines such as LuaTeX.

[9] We confess that we did not fall into this trap, because a colleague warned us about it. Roughly speaking, he said: "if you would like to convince your students, do not let them type their first examples themselves. Otherwise, you will be screwed up!"

[10] In the three-year university degree, Computer Science students practise another example of such an association, by means of JRuby [4], which allows programmers to mix Ruby [31] and Java [21]. We are also in charge of this teaching unit.

on modern compilation techniques, using a clear separation between *lexical* and *syntactic* analysers, whether the syntax is Java's or Ruby's, these examples being not limitative, of course. On the contrary, TEX's analyser is *monolithic*, which causes other conventions to be put into action. An illustrative difference of these two techniques is given by the dynamic processing of conditional constructs, that is, balancing "`\if...\fi`" [22], in comparison with static abstract trees resulting from parsing within modern languages, as shown in [15].

Likewise, we explored some meaningful subtleties about the notion of *function*, described in [16] and some constructs given in [14]. As another instructive comparison, the source program of TEX only uses static — and bounded — structures [23]; no dynamic allocation is performed anywhere in TEX's source files, contrary to what is practised within modern object-oriented languages.

Last but not at least, TEX's implementation may introduce Computer Science students to the notion of *legacy programs*. That is an important point, because these students difficulty in perceiving this idea, especially at their studies' beginning. They are taught that they should program *nicely*, but for most of them, that means *programming like their teachers*. Concerning TEX, its source files have been frozen according to Donald Knuth's wishes, unless new bugs are discovered. However, this program's monolithic organisation makes very difficult any evolution. Showing some extracts of the TEX program and explaining why they are difficult to maintain is not a betrayal of TEX's spirit: TEX's programming techniques belong to the past, and may be shown as such. Anyway, the door is not completely closed to further evolution: we can explain why and how *hooks* have been added [26, § A.4], as has the communication with Lua. In other words, TEX can be extended, but is not reworkable in a usual way.

4 For other science students

The specialised packages for Physics and Chemistry used by our students include siunitx [25, § 3.3.4], chemfig [33], mhchem [10], and the CircuiTi*k*Z package, based on Ti*k*Z,[11] well-known as a powerful tool to create graphic elements in LATEX [32]. In fact, introducing LATEX as being a WYSIWYM[12] program, as opposed to interactive (WYSIWYG)[13] systems such as Microsoft Word, is important, in order to emphasise the difference between the two 'philosophies'.

Likewise, showing a language for drawing figures, in comparison with interactive drawing programs, driven by mouse clicks and drags, is of interest for students, especially science students. First, XY-pic [6, Ch. 7] was shown to Mathematics students in the 2000s, then we have used Ti*k*Z, as a macro language based on the low-level graphics language PGF[14] [32].

5 Going further

The times that have been allotted for these teaching units based on LATEX have varied considerably, from 10 hours to around 30 hours. It is considered that an introduction to LATEX, even with a short duration, can include the bases — that is, sectioning commands, changing the layout and size of characters —, the list-like and `tabular` environments, and perhaps math mode, depending on whether or not students are supposed to use complicated mathematical formulas. If the duration permits, the beamer package [6, § 11.4] shows and solves some difficulty related to writing *slides*, whereas introducing the ecltree package [27, § 2.1.3] is of interest for science students in general, since drawing trees is used throughout many scientific topics, starting with Computer Science. Depending on contexts, we sometimes finished our units with the packages graphicx [25, § 8.1.3] and color [6, § 11.2]. Besides, the former allows students to cope with *floating figures* and problems related to their placements.

We explained in § 3 that we introduced Lua to 'more advanced' students, in order to emphasise the complementarity of TEX and Lua. In particular, Lua allowed us to extract some data given within separate files. Without going that far, we think that a demonstration of the pgfplotstable package [5] may hold audiences, in the sense that this package allows data of an Excel file to be used within a `tabular` environment of LATEX. Such data may be *accounts records* extracted from an activity report, or *measurements* performed during an experiment. Appendix A gives some examples of interfacing Excel files with LATEX.

6 Evaluating students

Sometimes we had to organise a 'classical' examination, as a graded lab class. In this case, students had to reproduce a formatted text from raw data. They had to use packages, too, some of which had been introduced during the classes, others not. In this last case, we gave students the packages' names; they had to look for its documentation, possibly on

[11] Ti*k*Z ***I**st **K**ein **Z**eichenprogramm*, recursive acronym for "Ti*k*Z is not a drawing program"

[12] **W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **M**ean.

[13] **W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et.

[14] **P**ortable **G**raphics **F**ormat.

the Internet. So, these examinations were some of the rare cases where the Internet was permitted to students.

For our 'quite advanced' students of this academic year (cf. § 3), we thought that an examination would not have been suitable for a practical topic, involving LaTeX packages or Lua statements subject to possible future evolution. So, we proposed a final project for grading them. This project's description is given in Appendix A, as an example of an advanced project using LaTeX & Cie. It seemed to us that students greatly enjoyed doing it.

7 Conclusion

Many documents may be used to support a deep introduction to LaTeX, possibly in *self-learning*. As far as we know, the most complete series is [7]. However, direct contact allows better understanding of publics' immediate reactions. Over the years, many students have asked us for advice, especially about resumés — in French or English. So we can think that teaching such units was successful.

From our point of view, teaching LaTeX should be something *philosophical*:

(i) easing possible change,

(ii) avoiding information redundancy,

(iii) getting high-quality output texts.

It may be strange that (iii) appears last. We personally think that it was pre-eminent in TeX's early years, around 1980. At this time, documents handled by students were often typed laboriously on typewriters with ugly fixes or formulas crossed out. Even if many present publications do not have as beautiful typography as they could, they are typeset better. In other words, the situation is not the same nowadays; current students are used to some *nice* typography, since many teachers give exercises and lab class subjects formatted with LaTeX. So arguments other than beauty should be given in order to convince modern students.

Acknowledgements

I thank the students of these teaching units about LaTeX. Some of them sometimes asked me questions that pushed me to delve deeper into a subject; some were difficult to convince, too, but I am satisfied with the results I can share with you now without pretense. Last but not at least, many thanks to Karl Berry and Barbara Beeton, who proofread a first version and suggested improvements.

A A project for advanced students[15]

[This appendix is an English translation — somewhat abridged — of the description of the project done by 2nd-year students in Computer Science from April to June 2025. Some footnotes have been added within this translation, they are signalled by the "[⋆⋆]" sign.]

A.1 Files to be downloaded

In addition to this description, given as a PDF[16] file, the *ten* files associated with this collective project are:

accounts.xlsx
arraydata.txt
Effectifs CJDCI 2024-2025.{csv,xlsx}
lualatex-arrays-plus.tex
split-plus.lua
the-superreport-ch-{1,2,3}.tex
the-superreport.tex

[Then we give specifics on where these files are located and how they must be organised; omitted here.]

A.2 Description

When this project is finished, compiling the master file the-superreport.tex shall produce a report describing the activity, accounts and staff of the company CJDCI.[17] This company sells the physical formulas given in Fig. 1 to anyone who knows how to use them. The past year's financial accounts are given in the Excel file accounts.xlsx and information about this company's staff in the file:

Effectifs CJDCI 2024-2025.xlsx

The final report must consist of the following three chapters:

(1. Activity) this chapter must include Fig. 1's formulas, *as exactly as possible*; in particular, pay attention to ensuring that the different formulas are well spaced, as are the successive components of a vector or determinant;

(2. Accounts) this chapter will format all the information present in the accounts.xlsx file as `tabular` environments: *any change* to this last Excel file must be applied at the report's next compilation by LuaLaTeX; to do that, you can use the pgfplotstable package after converting the Excel file into a CSV[18] file — as shown in the

[15] This appendix is also given in [18].

[16] **P**ortable **D**ocument **F**ormat.

[17] ***C**ompagnie des **J**oyeux **D**rilles du **C**MI **I**nformatique*. In English: *company of the joyful chaps of the CMI section of Computer Science*. "CMI" stands for ***C**ursus de **M**aster en **I**ngénierie*: they are enhanced training programmes, close to engineer schools'.

[18] **C**omma-**S**eparated **V**alues.

The CJDCI sells applications of vector calculus' operators. Let us recall that we are placed within an orthonormal basis, whose vectors are so-called $(\overrightarrow{i}, \overrightarrow{j}, \overrightarrow{k})$. In the following, u is a *scalar*[I] expressed as a function of the three coordinates (x, y, z) in space and $\overrightarrow{V}$ a three-dimensional vector, whose coordinates are (V_x, V_y, V_z). The formulas of the vector calculus' operators — *gradient*, *divergence*, *curl*,[II] and *Laplace operator*[III] — are:

$$\overrightarrow{\operatorname{grad} u} = \begin{pmatrix} \dfrac{\partial u}{\partial x} \\ \dfrac{\partial u}{\partial y} \\ \dfrac{\partial u}{\partial z} \end{pmatrix}$$

$$\operatorname{div} \overrightarrow{V} = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} + \frac{\partial V_z}{\partial z}$$

$$\overrightarrow{\operatorname{rot}\, \overrightarrow{V}} = \begin{vmatrix} \overrightarrow{i} & \overrightarrow{j} & \overrightarrow{k} \\ \dfrac{\partial}{\partial x} & \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial z} \\ V_x & V_y & V_z \end{vmatrix}$$

$$\Delta u = \operatorname{div} \overrightarrow{\operatorname{grad} u}$$

$$\overrightarrow{\Delta\, \overrightarrow{V}} = \overrightarrow{\operatorname{grad} \operatorname{div} \overrightarrow{V}} - \overrightarrow{\operatorname{rot} \operatorname{rot} \overrightarrow{V}}$$

The *gradient* of a scalar-valued differentiable function of several variables is the vector field giving the direction and rate of fastest increase. The *divergence* of a vector field produces a scalar field giving the rate that the vector field alters the volume in an infinitesimal neighbourhood of each point. The *curl* of a vector field is defined as the circulation density at each point of the field. The two versions of the Laplace operator[IV] are based on *second-order derivatives*.

[I] In Physics, *scalar quantities* are physical quantities that can be described by a single pure number, accompanied by a unit of measurement. Thus scalar quantities are opposed to *vector* and *tensor* quantities in Physics.

[II] [⋆⋆] This operator's French name is *rotationnel*, whence the notation used in the following.

[III] As you can see in this figure, *two* versions exist for this operator, for scalars and vectors.

[IV] Named after the French mathematician Pierre-Simon de Laplace (1749–1827).

Figure 1: Formulas sold by the CJDCI company.

pdflatex-arrays-0.tex file[19] — or process the CSV file by means of Lua statements — commented examples are given in the three files arraydata.txt, lualatex-arrays-plus.tex, and split-plus.lua;[20] give also *vertical bar charts* for *income* and *expenses*: incomes should appear above the x-axis and expenses below it; you could draw bars by means of the package Ti*k*Z's commands; we will assume that there cannot be more than *four* rubrics for income, the same for expenses: as a consequence, *eight* bars at most will have to be drawn; to do that, you can define an eight-argument command, some of these arguments may be bound to an `\empty` command, modelling that this argument is a *dummy*; a comparable example is given in the two files lualatex-arrays-plus.tex and split-plus.lua, where eight commands are bound to a dummy value first, then some of these commands are replaced by values retrieved from a text file;[21] this example should inspire you;

(3. Staff) this chapter will present all the information about this company's staff and is included in the Excel file:

Effectifs CJDCI 2024-2025.xlsx

as `tabular` environments; then we calculate the average age — in years and months as of June 1, 2025 — of the company's members;[22] here also:

- any change to the Excel — or CSV — file must be applied at the report's next compilation by LuaLaTeX,
- examples given in the three source files cited previously:

  arraydata.txt, lualatex-arrays-plus.tex, split-plus.lua

  should inspire you, too;

Also, person names should be typeset throughout the report only by means of the following command:

```
\newcommand{\cjdcipersonname}[2]{%
 #2\ #1%
 \index{\cjdcipersonnameindex{#1}{#2}}}
```

— where the first (resp. second) argument is the last (resp. first) name. As you can guess from this command's definition, an index of person names is built, typesetting these names being homogenised by means of the command:

[19] [⋆⋆] This file was demonstrated during lab classes ...

[20] [⋆⋆] ... so were preliminary versions of these three files.

[21] ... which may be a CSV file.

[22] [⋆⋆] In fact, the corresponding Excel file contains the list of this teaching unit's students: for each of them, the first and last name were given, followed by the birth date. We added ourselves, in order for the average age to be increased.

```
\newcommand{\cjdcipersonnameindex}[2]{%
 #1, #2}
```

these two commands:

```
\cjdcipersonname{...}{...}
\cjdcipersonnameindex{...}{...}
```

being defined within the file the-superreport.tex's preamble. Of course, you can build this index by means of the `makeindex` program, as shown in [17].

A.3 Terms

[Here we gave some information about the deadline and the procedure for submitting the project.] Since your class has 9 students, tasks will be shared as follows:

- a first three-student group will be in charge of Chapter 2,
- a second three-student group will be in charge of Chapter 3 and put the index of person names into action,
- a third three-student group will typeset Chapter 1 and integrate work by the other groups.

At each chapter's beginning, you must make precise its contributors, by means of the command:

```
\cjdcipersonname{...}{...}
```

as with any occurrence of person names.

We recall that different groups could work in parallel on the three chapters: for example, the group in charge of the first chapter could put:

```
\includeonly{the-superreport-ch-1}
```

in its main file's preamble. Later, the group in charge of the final integration will put:

```
\includeonly{%
 the-superreport-ch-1,the-superreport-ch-2,
 the-superreport-ch-3}
```

in the final preamble.

References

[1] Anne-Marie Aebischer, Bruno Aebischer, Jean-Michel Hufflen and François Pétiard: "Introducing New French-Speaking Users to LaTeX Quickly and Convincingly". *MAPS*, vol. 39, pp. 99–104. In EuroTeX 2009 conference. August 2009.

[2] Laurent Bloch : «LaTeX ou Word : qui l'eût cru ? Un exemple de falsification scientifique». *Cahiers GUTenberg*, Vol. 59, p. 5–12, Novembre 2024. doi : 10.60028/cahiers.v2024i59.174.

[3] Jean-Pierre F. Drucbert: *The shorttoc Package*. 10 April 2002. ctan.org/tex-archive/macros/latex/contrib/shorttoc/shorttoc.pdf.

[4] Justin Edelson and Henry Liu: *JRuby Cookbook*. O'Reilly. November 2008.

[5] Christian Feuersänger: *Manual for Package pgfplotstable*. 15 May 2021. ctan.org/tex-archive/graphics/pgf/contrib/pgfplots/doc/pgfplotstable.pdf.

[6] Michel Goossens, Frank Mittelbach, Sebastian Rahtz, Denis B. Roegel and Herbert Voss: *The LaTeX Graphics Companion*. 2nd edition. Addison-Wesley Publishing Company, Reading, Massachusetts. January 2009.

[7] The GuIT group, ed.: *Meeting, Sessione mattutina*, Turin. ArsTeXnica 28, pp. 8–134. October 2019. www.guitex.org/home/images/ArsTeXnica/AT028/arstexnica28_low_res.pdf

[8] Hans Hagen: "LuaTeX: Howling to the Moon". *Biuletyn Polskiej Grupy Użytkowników Systemu TeX*, vol. 23, pp. 63–68. April 2006.

[9] Harald Harders: *The numprint Package. Version v1.39*. 28 August 2012. ctan.org/pkg/numprint.

[10] Martin Hensel: *The mhchem Bundle. Documentation for the LaTeX Packages mhchem v4.10, hpstatement v2.1.0 and rsphrase v3.11*. 29 January 2024. ctan.org/tex-archive/macros/latex/contrib/mhchem/mhchem.pdf.

[11] Jean-Michel Hufflen: "TeX's Language within the History of Programming Languages". *Biuletyn GUST*, vol. 22, pp. 23–32. In *BachoTeX 2005 conference*. April 2005.

[12] Jean-Michel Hufflen : *Premier contact avec LaTeX*. Document de support de travaux pratiques. Janvier 2006.

[13] Jean-Michel Hufflen : *Mode mathématique*. Document de support de travaux pratiques. Février 2006.

[14] Jean-Michel Hufflen: "Using TeX's Language within a Course about Functional Programming". *MAPS*, vol. 39, pp. 92–98. In EuroTeX 2009 conference. August 2009.

[15] Jean-Michel Hufflen: "Which Success for TeX as an Old Program?". *ArsTeXnica*, vol. 30, pp. 24–30. In Proc. GUIT 2020 meeting. October 2020.

[16] Jean-Michel Hufflen: "Functions in TEX and Elsewhere". *ArsTEXnica*, vol. 32, pp. 51–53. In Proc. GUIT 2021 meeting. October 2021.

[17] Jean-Michel Hufflen : *Outils périphériques à LATEX — Millésime 2023*. Cours polycopié. Octobre 2023.

[18] Jean-Michel Hufflen: "Advanced Lectures Using TEX & Co.". In: *TEX in Education. Proc. BachoTEX 2025*. April 2025.

[19] Roberto Ierusalimschy: *Programming in Lua*. 2nd edition. `lua.org`. March 2006.

[20] *Stay on Top of Your Literature. The Efficient Way to Collect, Organize & Discover*. July 2025. `jabref.org`.

[21] *Java Technology*. March 2008. `java.sun.com`.

[22] Donald Ervin Knuth: *Computers & Typesetting. Vol. A: The TEXbook*. Addison-Wesley Publishing Company, Reading, Massachusetts. 1986.

[23] Donald Ervin Knuth: *Computers & Typesetting. Vol. B: The Program*. Addison-Wesley Publishing Company, Reading, Massachusetts. 1986.

[24] Leslie Lamport: *LATEX: A Document Preparation System. User's Guide and Reference Manual*. Addison-Wesley Publishing Company, Reading, Massachusetts. 1994.

[25] Frank Mittelbach with Ulrike Fischer: *The LATEX Companion*, vol. 1. 3rd edition. Addison-Wesley Publishing Company, Reading, Massachusetts. 31 August 2023.

[26] Frank Mittelbach with Ulrike Fischer: *The LATEX Companion*, vol. 2. 3rd edition. Addison-Wesley Publishing Company, Reading, Massachusetts. 31 August 2023.

[27] Frank Mittelbach and Michel Goossens, with Johannes Braams, David Carlisle, Chris A. Rowley, Christine Detig and Joachim Schrod: *The LATEX Companion*. 2nd edition. Addison-Wesley Publishing Company, Reading, Massachusetts. August 2004.

[28] Frank Mittelbach et Michel Goossens, avec Joannes Braams, David Carlisle, Chris A. Rowley, Christine Detig et Joachim Schrod : *LATEX Companion, 2e édition*. Pearson Education France. Traduction française de [27] par Jacques André, Benoît Bellet, Jean-Côme Charpentier, Jean-Michel Hufflen et Yves Soulet. Octobre 2005.

[29] Will Robertson, Khaled Hosny and Karl Berry: *The XqTEX Reference Guide*. 11 February 2025. `ctan.org/pkg/xetexref`.

[30] Denis B. Roegel : «Anatomie d'une macro». *Cahiers GUTenberg*, Vol. 31, p. 19–27. Décembre 1998.

[31] *Ruby, a Programmer's Best Friend*. July 2025. `www.ruby-lang.org/en/`.

[32] Till Tantau: *The TikZ and PGF Packages. Manual for Version 3.1.10*. 11 April 2025. `tikz.dev`.

[33] Christian Tellechea: *chemfig. A Package for Drawing Molecules*. December 2023. `ctan.org/tex-archive/macros/generic/chemfig/chemfig-en.pdf`.

◇ Jean-Michel HUFFLEN
Université Marie et Louis Pasteur,
FEMTO-ST
16, route de Gray
25030 BESANÇON CEDEX
FRANCE
`jmhuffle (at) femto-st (dot) fr`
`http://members.femto-st.fr/¶ Hufflen-Jean-Michel/en`