

Script for Interview about *ΛT_EX and Friends*

M. R. C. van Dongen

July 13, 2012

Contents

1	Introduction	2
2	Typography	3
2.1	Typeface Selection	3
2.2	Kerning	3
2.3	Ligatures	3
3	Languages	4
4	What More Can ΛT_EX Do?	5
5	How Does ΛT_EX Work?	6
6	Structured Writing	7
7	Bibliography Support	8
8	About the Book	9
8.1	Common Themes	9
8.2	Effective Presentation	10
8.3	Consistent Presentation	10
8.4	Planning and Development	11
8.5	Maintenance	11
8.6	Control	11

1 Introduction

1. My name is Marc van Dongen and I'm a lecturer in the Computer Science Department at University College Cork, in Ireland.
2. We're here because I wrote a book that is called *Λ_TE_X and Friends*.
3. The book is published by *Springer* and it's about publishing beautiful papers, reports, theses, books, and other documents with a software called Λ_TE_X.
4. Λ_TE_X is *free*, it's *very* stable, and it's available on Windows, on Unix, and on the Mac.
5. Λ_TE_X is used in mathematics, computer science, and physics, but also in other areas, including *humanities* and the *life sciences*.
6. I'll get back to that later.

2 Typography

2.1 Typeface Selection

1. \LaTeX provides state-of-the-art typesetting.
2. \LaTeX usually provides a *serif* typeface, a *sans serif* typeface, a *monospaced* typeface, and an *italic* typeface.
3. Of course \LaTeX also provides bold face versions of these style.
4. \LaTeX can access all features in your font files.
5. For example, if your font files provides them, \LaTeX may provide special letter forms, for example special letter forms for the start or the end of a word, *historic* letter forms, and so on.
6. \LaTeX can provide proper *small caps letters*, which are useful for typesetting abbreviations and acronyms.
7. Modern fonts provide several kinds of figures and \LaTeX can make them available, including (1) *proportional oldstyle figures*, which are useful for the running text, (2) *proportional lining figures*, which are useful in uppercase text, and their (3) *fixed-width figure counterparts*, which are useful for tables.

2.2 Kerning

1. \LaTeX provides proper *kerning*, which means that it knows when to override the default distance between certain pairs of letters.
2. Usually kerning means reducing the distance between certain pairs of letters.
3. For example, between an uppercase *T* and a lowercase letter.
4. Other combinations are also possible.

2.3 Ligatures

1. Likewise, \LaTeX provides *ligatures*, which are special shapes for certain combinations of letters.
2. Many typefaces have ligatures that involve the letter *f*, for example *fl*, *fi*, *ff*, *ffl*, and so on.
3. However, other ligatures are also possible.

3 Languages

1. \LaTeX can typeset a variety of languages.
2. \LaTeX provides bi-directional and vertical typesetting.
3. \LaTeX can typeset Hebrew.
4. \LaTeX can typeset Arabic.
5. \LaTeX can typeset Chinese.
6. If you're interested in Elves, Orks, Wizzards, Hobbits, and Tolkien's stories about Middle Earth, then you may be glad to learn that \LaTeX can also typeset Tengwar.

4 What More Can L^AT_EX Do?

1. There is a common misconception that L^AT_EX is used only by *mathematicians, computer scientists, and physicists*.
2. To a degree this is true because L^AT_EX is used a lot by these communities.
3. However, L^AT_EX is also used in the *life sciences, humanities, and outside academia*.
4. Let's give *some* examples.
5. L^AT_EX is used in *Chemistry*.
6. You can typeset *logic diagrams* with L^AT_EX.
7. L^AT_EX is used in *Molecular Biology*.
8. For example, you can typeset *DNA sequencing chromatograms and nucleotide and peptide sequence alignments* with L^AT_EX.
9. However, L^AT_EX is also used in other areas.
10. For example, L^AT_EX can typeset *Gantt charts*.
11. You can typeset *syntax trees* with L^AT_EX.
12. You can typeset *poetry* with L^AT_EX.
13. You can typeset *music* with L^AT_EX, and so on.

5 How Does L^AT_EX Work?

1. So how does L^AT_EX work?
2. Basically, L^AT_EX is a typesetting engine sitting on top of a Turing-complete procedural markup language.
3. Effectively, this means that you can tell L^AT_EX what to typeset, but also that L^AT_EX can *compute* the things you want it to typeset, just like a spreadsheet can compute the numbers in the spreadsheet's output columns.
4. You prepare your document in a L^AT_EX input file, which is a basic text file, which can be created with any editor such as vim, emacs, or notepad.
5. However, many users nowadays use a L^AT_EX IDE.
6. At the start of your input file you tell L^AT_EX about the *document class* of your document.
7. The document class determines the style of the output document.
8. Common document classes are *book*, *report*, *article*, and so on.
9. For me the document class is usually *article*.
10. After you've told L^AT_EX about the document class, you tell it about the *title* and the *author* of your document.
11. The main body of your output document is defined in a the *document* environment, which is defined in a separate section of the input file.
12. In the document environment, you can tell L^AT_EX to typeset the title of your output document, the table of contents, and so on.
13. Using markup commands you indicate the start of the *sectional units* of your output document and L^AT_EX will automatically number them for you in the output document.
14. The rest of the input file determines what's in the main body of your output document.

6 Structured Writing

1. One of the advantages of \LaTeX is that it forces you to focus on the content of the document, not on the style.
2. For example, a mathematician who's writing a theorem writes it in a dedicated theorem environment in the input file.
3. The theorem environment determines the style of the theorem in the output file.
4. Likewise, the mathematician provides a proof for the theorem in a dedicated proof environment that determines the style of the proof in the output document.
5. Writing a document like this relieves the user from the task of looking after the appearance of the document; it's done by \LaTeX .
6. This saves time, which the user can spend on the content of the document, which is what really matters.
7. Another advantage of this approach is the user can see the structure of the output document by looking at the structure of the input file.
8. For example, a theorem is provided in a theorem environment, a proof is provided in a proof environment, and so on.
9. This gives the user an extra handle on *writing* the document, *developing* the document, and *maintaining* the document.

7 Bibliography Support

1. Academics spend a lot of time writing papers.
2. These papers contain many citations and references to other academic works.
3. These citations are very important, so it is important to get them right.
4. \LaTeX provides excellent support.
5. This is how it works.
6. In the input file the user uses commands that typeset the citations in the running text.
7. Each command depends on a *label* that determines the cited work.
8. An external, textual database associates each labels with the required information about its work.
9. This required information typically consists of the *authors* of the work, the *title* of the work, the *year of publication*, and so on.
10. \LaTeX , in combination with a program called `bibtex`, reads each label, gathers the information about the corresponding work from the external database, and typesets the citations in the main text.
11. The task of preparing the reference section is also completely automated.
12. All the user has to do is tell \LaTeX *where* to print the reference section.
13. In addition they must provide the external textual `bibtex` database that associates the label of each cited work and its authors, its title, and its remaining properties.
14. In short, using \LaTeX helps saving time.

8 About the Book

1. The book explains all you need to know about writing articles, books, and theses with \LaTeX .
2. There are many books about \LaTeX and some explain everything.
3. What you end up with is something that's too big for a student who wants to write a thesis.
4. My book explains just enough.
5. The book is for novices who want to learn \LaTeX but it is also for users who are on the path of becoming an intermediate user.
6. At the same time the book can be used as a reference for seasoned users.
7. \LaTeX has been around for quite a while and there are many solutions for many problems.
8. Unfortunately, some of these solutions aren't compatible.
9. In my book, all solutions are compatible.

1. The book starts with a chapter about how to produce a basic pdf output document with \LaTeX , how to cross-reference a sectional unit, how to cite a work, how to create one or several bibliographies, how to create one or several indexes, and so on.
2. It proceeds with a chapter about typesetting the main text: how to typeset characters with diacritics and other special characters, how to typeset and avoid ligatures, how to emphasise text, how to typeset an abbreviation or an acronym, and so on.
3. Next there's a chapter about itemised, enumerated, and description lists.
4. A whole chapter is dedicated to producing technical diagrams.
5. There are two chapters about presenting data with tables and dataplots. These chapters explain how to do the typesetting but also give some clues about how to design effective tables and plots.
6. There are chapters about presenting mathematics and computer algorithms.
7. One of the chapters is about presenting computer ("powerpoint") presentation.
8. There are also more technical chapters about implementing user-defined macros and environments.
9. These chapters explain how to exploit \LaTeX 's computing abilities to automate more and less obvious typesetting tasks.
10. The book is an interesting mixture of topics.
11. I hope it has something for everybody.

8.1 Common Themes

1. There are a couple of reoccurring themes in the book.
2. **Effective presentation,**
3. **Consistent presentation,**
4. **Planning and developing the presentation,**
5. **Maintaining the presentation,** and
6. **Controlling the presentation.**
7. Let's have a look at these themes in more detail.

8.2 Effective Presentation

1. Let's start with effective presentation.
2. A common problem with reports and theses is that they have poor tables.
3. To a lesser degree you also see this in academic papers.
4. The reason why the tables are poor is not because the tables don't contain the information but because the presentation is poor.
5. For example, the format or the alignment of the numeric data in the tables may make it impossible to quickly compare the numbers in the columns.
6. As another example, it may be difficult to see the trend in the data because the data are presented in the wrong order.
7. The book has a whole chapter on presenting data with tables.
8. It explains how to present a table with rows that are easily scanned in horizontal direction, with columns that are easily scanned in vertical direction, and with numerical data that can be compared with ease.
9. It also explains how to design your table so the trend of the data in the table becomes clear.
10. There is a similar chapter about presenting data with data plots.

8.3 Consistent Presentation

1. The second theme is consistent presentation.
2. This is about (1) presenting elements that play the *same role* in the document in the *same, consistent style* and (2) how to achieve this with a minimum amount of effort.
3. Let's assume you have a document with lots of technical diagrams.
4. (I could have also given an example about text but this example is easier to understand.)
5. Some of the lines in the diagram draw the objects in the diagrams.
6. Other lines and arrows are used to indicate the sizes of the objects.
7. Since the different kinds of lines play different roles, it makes sense to give them a different style.
8. By choosing a proper style, we can make the lines that draw the objects stand out and the other lines sit in the background.
9. For the lines that draw the objects we may define a *normal* style that draws the objects in black.
10. For the other lines we may introduce a *size* style that draws these lines in blue.
11. By using the styles to draw the lines we ensure the presentation of the normal lines and the size lines is consistent throughout the presentation.
12. Using styles in this way is similar to using layers in a CAD program, where you draw certain kinds of objects in certain layers.
13. For example, construction lines in a construction layer.
14. Using layers like this pays off, so why not use a similar approach when writing a document with \LaTeX ?

8.4 Planning and Development

1. A consistent style is only one aspect of report, paper, thesis, and book preparation.
2. You also have to plan the writing of the document and develop the style of the presentation.
3. This brings us to the third theme: planning and development.
4. Let's continue our previous example and see how \LaTeX can help us develop/improve the style.
5. Remember that the size lines in the diagrams are drawn in blue.
6. Using blue really was a bad idea.
7. Let's say we want to make them red.
8. Since we've encapsulated the definition of the style, we can implement the *global* change from blue to red by making a single, *local* change in the definition of the style.
9. With a similar, simple change we can draw all size lines with a very thin line type.

8.5 Maintenance

1. The fourth theme of the book is maintaining the presentation.
2. This is related to developing the presentation.
3. To demonstrate how this works, I'll give an example of how *not* to do it.
4. Let's assume we're writing a paper about apples and pears.
5. The paper has four sections: an introduction, a section about apples, a section about pears, and a summary.
6. A good paper should briefly outline what's in the paper, so the introduction states that section 2 is about apples and that section 3 is about pears.
7. With \LaTeX you should *never* hardcode the numbers of sectional units in your input file, but let's suppose we do this anyway, so we write that section 2 is about apples and section 3 is about pears.
8. Next let's assume that for some reason, we decide it's better to change the order of the sections about apples and pears.
9. This on its own may require a substantial some editing operation that may introduce errors but let's say we manage to change the order of the sections without making errors.
10. Of course we also have to change the hardcoded corss-references to the sections about apples and pears in the introduction.
11. Let's say we also do this without introducing errors.
12. Maintaining a document like this still doesn't guarantee proper cross-references to the sections about apples and pears.
13. For example, we may have overlooked the hard-coded cross-references to sections 2 and 3 in the summary.
14. How to cross-reference sections properly is explained in the book.

8.6 Control

1. The last theme is about controlling the presentation.

2. Here we exploit \LaTeX computational capabilities—that it can compute things, which it can then typeset for you.
3. This is similar to the functionality of a spreadsheet.
4. Some of the columns in the spreadsheet are input columns and some of the columns are output columns.
5. What's in the output columns is computed from the input columns.
6. If you use \LaTeX 's computational capabilities cleverly, then you can produce *several* output documents from the same input file.
7. For example, let's assume you have a *modal* input file.
8. The style of the output document is determined by some *mode*—a style parameter that you provide when you tell \LaTeX about the document class.
9. For the *notes* mode, the output is a printable article.
10. For the *presentation* mode, the output is a computer presentation.
11. So far this is pretty much similar to letting the document class determine the style of the output document.
12. The book shows that you can do *much* more.
13. For example, you can use certain parts of the input file for the notes and *not* for the presentation, other parts for the presentation and *not* for the notes, and the remaining parts for both the notes and the presentation.
14. This lets you share some of the input and simultaneously develop the two output documents in the same input file.
15. Doing this helps you synchronise the presentation of the two output documents.
16. To give one concrete example, let's assume you want a table of contents in the lecture notes but not in the presentation.
17. By providing the right commands, you can tell \LaTeX it should only include the table of contents if the current mode is *notes*.
18. When you use the input in *notes* mode, the output document is an article *with* a table of contents.
19. However, when you use the input in *presentation* mode, the output is a computer presentation *without* table of contents.
20. You can use the same principle to ordinary text.