TEX for TEXnical Typists

Charles R. Martin
National Biomedical Simulation Resource, One University Place Suite 250, Durham, NC 27707
(919) 383-2256. Internet: `crm@nbsr.duke.edu`

Abstract

Many TEX users are not programmers or mathematicians but technical typists, practitioners of a skilled craft. These users often find existing TEX texts intimidating and cryptic. *TEX for TEXnical Typists* is intended especially for these users. Each unit introduces a few concepts, then immediately reinforces those concepts with practical experience using a short document. Students see visible results immediately, which leads to rapid progress and greater confidence. The course encourages an experimental attitude that serves well in practice. *TEX for TEXnical Typists* appears to be an effective way to teach technical typists to create attractive documents.

Teaching TEX to nontechnical staff. People who prepare documents using TEX find themselves not just typing, but setting type. Doing so means that they have an opportunity to exercise considerably greater creativity and craft. It also means, however, that they must learn new skills; these skills require effort and time to learn. Mathematicians and programmers often find acquiring these skills to be relatively easy, because they have background knowledge on which to draw.

Many people use TEX who are neither programmers nor mathematicians. These people often have difficulty getting started in TEX. *TEX for TEXnical Typists* is an introductory course directed toward nontechnical users of TEX, who are neither programmers nor mathematicians, but who have experience with technical document preparation. These users are sometimes known as "technical typists", "technical editors", or "technical writers". All of these groups have their own expertise and their own qualifications; for simplicity and because the course is intended to be as broadly applicable as it can be, we can consider the technical typists as our example audience.

TEX for TEXnical Typists is based on an operational approach to TEX. In this approach, the students are exposed to a very few concepts at one time; these concepts are then immediately reinforced with exercises. This quick reinforcement has two effects: it helps the students learn each concept, and reduces confusion; more important, each successful exercise reinforces the students' confidence. With greater confidence, the students are more willing to experiment on their own, and more willing to brave the mystic realms of *The TEXbook*.

The audience: Technical typists. Document preparation is a craft in itself. As with most crafts, it has aspects of labor, skill, and art: the simple mechanical labor of keying a text; the skills needed to handle complexities of the language, of punctuation and capitalization; and artistic aspects like those of book design. The people we speak of as technical typists are the tool and die makers of the crafts of the secretary: they must not only be masters of the skills of typing manuscripts, they must be aware of all the other issues involved in preparing drafts, "clean copy" for submission, and camera ready copy for publication. Surprisingly often now, photo-offset printing means that these people also prepare the final designs and camera-ready copy for entire books.

Technical typists are usually accustomed to using complex terminology, and those who prepare scientific texts are skilled in using difficult symbology. We can expect them to have considerable skill with the mechanics of typing as well as aesthetic sense and a sense of good style. We can't expect them to be skilled programmers, or to be more than very slightly acquainted with mathematics.

Intimidation. Someone reading *The TEXbook* is immediately presented with stylish but dense prose, and confronted with the dread "double-bend" sign. From the first time I tried to teach a technical typist to use TEX, I have heard some variant of the sentence "I'm just too dumb to learn this." This is a sign that they are intimidated by the material. The course is based on the idea that this intimidation is the most important impediment to learning to use TEX.

Why are they intimidated? In my opinion, there are several reasons:

- As a reference, *The TEXbook* must present a lot of material; *The TEXbook* at once presents a typesetting system, a tutorial on typesetting, and the reference for a complex programming language. The presentation is necessarily dense.
- *The TEXbook* is not structured as a cookbook; rather than presenting recipes, it presents the material in a form more similar to mathematics texts. Learning the material requires more than one reading.
- The material is directed toward the particular typesetting problem Knuth found most troubling: typesetting mathematical text. Unlike TEX's technical users, many technical typists are not mathematically skilled, and often they feel a certain distaste for mathematics in general.

The course is particularly intended to avoid intimidating the students at any time. The key to this is to structure the course always to instill confidence. Some part of this is attitude: how many people have taken courses in which the instructor's managed with every word and every sentence to imply how much smarter the instructor was? But beyond questions of attitude, there are many specific steps that can be taken to instill confidence in the students.

The instructor should always be careful to praise the students repeatedly, even though I think many people feel that repeated praise is somehow patronizing. As long as the praise is sincere — which it can be with a little thought on the instructor's part in order to recognize the praiseworthy parts of any sincere effort — then students are quite willing to accept it.

The dual of the exhortation to praise repeatedly is never to punish. The greater the intimidation level of the students to start, the more easily any punishment can be interpreted as a sign that competence in TEX is beyond the student. Once students believe they can never become competent, that belief becomes prophecy. Similarly, the more often the instructor reinforces the belief that competency can be achieved, the more likely that belief will become self-fulfilling prophecy.

The general rule should be "to instill confidence, ensure success; to build competence, ensure success in successively more difficult problems."

The course: Structure and philosophy. *TEX for TEXnical Typists* is organized around a series of small tasks, chosen so that each of the tasks is slightly more than trivial, but not much more. For example, the first task is to type two paragraphs of text with no font changes and no TEX commands whatsoever, and process it through TEX to a printed document. These small tasks I call "units" for want of a better term.

Each unit in the course is structured around a small number of concepts. The first unit, for example, is intended to ensure that the student understands the basics of using whatever editor and operating system is used in the course, and that the student has succeeded in producing a document. These are intentionally small goals; the size of the succeeding steps, and the number of concepts included, will increase, but a starting point this simple makes it certain that the students start the course with a success.

The concepts are introduced through a specific document that serves as the focus and assignment for the units. These documents are chosen (directly influenced by Knuth) in the attempt to be light and amusing. Since I don't trust my comedic writing skills, early texts are stolen from collections of jokes and small, humorous essays. Later texts, of course, must be chosen to illustrate specific skills such as typesetting multiple integrals; the number of jokes using multiple integrals being small, the examples are composed specifically for the course.

Each unit has a few minutes of lecture, followed immediately by an experiment; this experiment first uses the concept, then extends the concept in some way that requires the student to use some creativity and thought to reach a solution. Later units necessarily build upon earlier units; this can be extended to make certain that concepts from previous units are always re-used in some later unit.

I cannot emphasize enough that *each unit is directed toward success*; those students who have some problem with a step get help from the instructor and from each other until they successfully complete the unit. (As an aside, I believe that allowing and encouraging students to help one another has several benefits. It improves the environment for everyone, not least the harried instructor. It leads the better students to help bring the less-talented along, and it also improves the more-talented students' competence.)

This approach has a number of features that can be explained psychologically. First, the structure of the units means that there is a short cycle between the introduction of a concept and its actual use, and hence little separation between the first introduction of a concept and reinforcement in correct use of the concept. From the time of Skinner's pigeons, it's been well-known that this encourages quick learning.

Second, each unit leads to the successful completion of a task that the student perceives as a difficult one; the pleasure of doing this is hard to beat. This in itself reinforces the student in the concepts used, and makes the class rather a pleasure for both teacher and student. Since the student is generally motivated for one reason or another to learn the material, sucessively greater achievements are very rewarding for everyone. (There is little that is better than teaching a topic to happy students.)

Finally, when concepts introduced previously are reused, and these reuses are pointed out, the interactions between the concepts is emphasized. To ensure this for all concepts, it is necessary to include some review units at the end of a course in order to make several reinforcing uses of the final concepts. As more and more such associations are built up, the students are increasingly likely to build up their own associations. This leads to greater comprehension, as the students' cognitive models are expanded.

To summarize quickly, a successful unit is built using these guidelines:

- Don't be afraid of asking questions that are too easy. Any unit that introduces anything new leads to learning. People prefer getting the right answer to the pleasure of any noble effort that doesn't lead to a right answer.
- For that reason, target the units' contents to ensure that all students achieve success. Use the talents of any more-talented people in the class to help the less-talented, in order to balance the load and prevent boredom.
- Ensure that later units use concepts from previous units. (Even the very first unit can use concepts from the students' experience as typists or compositors.)
- Plan for several reinforcing units as the last units in a course, which introduce no new concepts or few new concepts, but which ensure that all previous concepts are reinforced at last.

Discussion and conclusions. I've now taught this course many times; much of the content of this paper comes from self-examination of my own mistakes. (Much of the rest comes from remembering the most horrible experiences of my own checkered academic career: there is a lot to be said for remembering the things one hated most in courses, and trying *not to make the same mistakes!*)

Here are some of my own mistakes:

- Require that the prerequisites be met. One of my worst experiences in the course was the time I agreed to teach an experienced group an abbreviated version of the course, only to find that the students not only were *not* experienced with TEX but were not in general computer literate. If one cannot ensure similar competence on the part of all students, one is reduced to starting at the level of the least competent member of the group. Nothing can keep the course from being boring for some part of the rest of the group.
- Don't do requests. The course I mentioned above was an amazingly productive source of mistakes. The other major one was agreeing to abbreviate the course. A one-day course is a much different thing from a four-day course, and must be planned as thoroughly. If it's worth doing, then plan it out as carefully and call it another course.

  Similarly, any extensions can be troublesome. The worst of these was attempting to add serious macro programming to the course: students without a programmer's algorithmic skills find macros completely opaque, and these skills cannot be taught in an hour.
- Provide the notes. Any course in TEX will necessarily include a lot of details of various coding and markup conventions. You can cover lots more material if you don't expect the students to take careful notes; instead provide copies of your slides or detailed copies of your notes for each unit.
- Insist that every student have a copy of *The TEXbook*. The students *must* be able to look things up on demand. (Plus, they must become comfortable with the book, and introducing them to it in class again reduces the intimidation.) They will need their own copy eventually, in any case.
- Ensure sufficient resources. Without sufficient resources, the delay between concept and reinforcement is longer. As the delay grows longer, the benefits of the short concept-use cycle grow less. In particular, students *must* be free to make mistakes, which means that they must

be insulated from whatever cost there may be in producing many printed results. This is particularly true of environments where there is no adequate previewer available.

Conclusions. *TEX for TEXnical Typists* has proven to be effective and successful as a way to teach composition using TEX to students for whom *The TEXbook* and other texts are not very effective: students with little technical background. These students — particularly technical writers, technical editors, and technical typists — can be taught to be extremely effective TEXnicians; they bring to the job skills and taste that technically-educated users have never had reason to develop. But to become effective, they must be presented the material in a special way.

One objection that might be made to this paper is that it is simply an advertisement for the particular TEX course I have designed. It is even in part true: I am happy to teach the course, and do charge a fee. But I think the points that I am making are essential ones for anyone trying to teach TEX; I hope that the concepts in this paper will make it possible for others to construct effective courses along the same lines as *TEX for TEXnical Typists*. The difficulty of getting started in TEX is, in my opinion, one of TEX's major problems; if we don't solve it, the seductive ease of use of "what you see is what you get" may doom us to years of poorly designed and poorly typeset documents. As with so many other problems, the solution is education.

A fit of philosophy. Document processing is moving away from a separation between the author, the person keying the document, and the person composing the typeset page, and toward a separation between the roles of creator of the text and creator of the attractive text on a page. More and more, those who compose the text on the page will take on a more creative, and less mechanical role. We will eventually need to recognize these people as craftspeople. TEX is a part of this process; TEX skills allow these people greater creativity, as they exert greater control over the results of their work. More creativity and more control over the results means more power and inevitably more professionalism. Teaching TEX to "technical typists" means teaching these skilled craftspeople ways to enhance the parts of their craft that require the most skill and are most like an art.

Acknowledgement. The course *TEX for TEXnical Typists* grew from one I originally wrote for the Physiology Department (now Cell Biology) at Duke University and its then-chairman, Edward A. Johnson, M.D. I've been able to continue to develop this course, and to present it, at least in part through the forbearance and assistance of J. Mailen Kootsey, Ph.D., of the Departments of Cell Biology and Computer Science at Duke University. Facilities for the preparation of this paper were provided through the kind assistance of the National Biomedical Simulation Resource, supported under NIH Division of Research Resources grant 5P41-RR01693.