

thors of those mediæval manuscripts could have gained a lot from using T_EX: the average paragraph starred ‘Overfull Boxes’ on just about every second line.



Figure 4. The cloister that houses the library of Avranches.

Going home

Taco, Hans and Siep took the train back to Holland on Wednesday, and this time everything went remarkably smoothly. Erik, however, was tempted into another adventure. The Polish delegation had asked him if it were possible to stay overnight at his place, so they could split their journey back home (2000 km by car) in two equal parts. Naturally he said yes and in spite of lots of rain and fog we managed to reach Groningen at 5AM.

Conclusion

Though it was a very packed conference (only three days including the tutorials) it wasn't overfull. The organization was very well done, the accommodation was very good and the enthusiasm of the people attending the conference was high. All in all something to remember.

Expansion, what is that?

Let's start with a very plain T_EX file:

```
Hi there! \end
```

When T_EX reads this one line file, it typesets the words *Hi there!* and then ends the session. Here `\end` is not read as text to be typeset, but as a directive that tells T_EX to do something special. By default, this sequence of characters equals the built in primitive with the meaning *end this session*.

One cannot only call for build in primitives, but also define his or her own macros. So we've got primitives and macros!

```
\def\name{Hans} Hi there \name ! \end
```

Here we define a macro `\name`, that is said to expand into Hans. In most cases, one will use macros for defining more complicated things, but the principles remain the same.

In this small file, after accepting the definition of `\name`, T_EX inserts the two words *Hi there* plus a space. Next it meets `\name` and looks up its meaning. This macro expands into *Hans*, and after typesetting this word, T_EX reads on and sees the exclamation mark. Finally `\end` ends the session.

Here expansion means as much as *insert the meaning of a macro* and when doing so, *expand all macros that are part of this macro*, and so on. Such a macro can contain text or a program, which is a sequence of T_EX primitive operations, macros and/or text. Often those macros influence the typesetting process.

So in fact T_EX is constantly grabbing text and expanding macros. Although there is no sharp border between those activities, one should be aware that T_EX is constantly switching between typesetting text, interpreting character sequences, and executing the programs laid down in macros.

I have to admit that this is not the whole truth. Typesetting is a multi—step process in which T_EX is doing quite a lot. In the previous example T_EX first scans a paragraph, then determines the best line breaks, and finally adds the results to the current page, or in T_EX terms: the main vertical list. Then T_EX looks if and how the content of the current page should be split over more pages. If indeed a full page can be split off, T_EX calls the appropriate macros to complete the page, that is, add a page number, headers and footers, flush floating bodies, add footnotes, do some housekeeping etc. etc.

(Continued on page 43)

tention, some colors will have traveled from rgb to cmyk back to rgb and again to cmyk and will be much the worse for wear ;-).

Further reading

For online documents, dates and exact locations are omitted, because these are subject to too much change.

Adobe Systems, PostScript Language Reference Manual, second edition, Addison-Wesley 1990.

Adobe Systems, Portable Document Format Reference Manual. Available online from www.adobe.com.

Adobe Systems, PDF for Prepress Workflow and Document Delivery. Available online from www.adobe.com. Various other documents of interest on

pdf, PostScript Level 3 and color management can be found at this site.

David Carlisle and Sebastian Rahtz, Graphics package. Available from CTAN and included in most T_EX/L^AT_EX distributions. This is in docstrip format; just run `latex graphics.dtx` to get formatted documentation.

Michel Goossens, Sebastian Rahtz and Frank Mittelbach, The L^AT_EX Graphics Companion, Addison Wesley, 1997.

Thomas Rokicky, Dvips manual. Included in the dvips distribution; available from CTAN and included in most T_EX distributions.

Postprocessing software vendors, see e.g. www.lantanarips.com for Crackerjack, and www.imation.com for Trapwise and Presswise.

Expansion, what is that? (continued)

Although from the users point of view, T_EX expands whatever macro it meets while reading the paragraph, in practice expansion can be postponed or prohibited. References for instance can only get their meaning when the content is placed on the page, and this can be many paragraphs later. Postponing is needed to keep the references in tune with the page numbers. When you hear macro writers talk of whatsits, they are probably talking about such postponed expansion and hidden things.

The story is still not completed. Hidden for the user, and depending on the macro package in use, T_EX also inserts

things like color directives, indentation, skips and when asked for all kind of frills, like paragraph numbers. This means that what you type is not per se what you will get, in fact, what you see is what T_EX made of your input. Some things are quite complicated in T_EX. Take for instance multi-column typesetting. Splitting columns is to be programmed and is not part of T_EX. The same goes for adding line numbers. Such at first sight simple things ask for recalculating page breaks and/or reading back already typeset lines and post processing them.

Hans Hagen