German Hyphenation and *Umlauts* in TEX

Bernd Schulze
University of Bonn
Institute for Applied Mathematics
Wegeler Str. 6, D-5300 Bonn, W. Germany

Since we first heard of Donald Knuth's plans for writing a typesetting system for mathematical texts, we have observed the development of TEX with great interest because our Institute regularly publishes a great number of mathematical preprints. As soon as we could get hold of the first Pascal version we installed it and started gaining experience. Since then we have been able to help several German institutions with their TEX installation. While doing this, we noticed that there were two main obstacles to the applicability of TEX to German publications, namely hyphenation and coding of characters particular to the German language.

So there was the need to make the hyphenation capabilities of TEX available for the German language. Our solution to the problem also simplifies typing of the special characters. We took a different approach than the University of Milan (TUGboat Volume 5, No. 1, pages 14–15) where the TEX program itself was modified to incorporate Italian hyphenation rules. In order to maintain compatibility we rejected that idea.

In the German language, there are some hyphenations that can only be handled by their explicit specification in an exception dictionary. The provisions made by the discretionary hyphen are sufficient (cf. The TEXbook, p. 96). But these cases are rare; satisfactory hyphenation is indeed attainable by making use of nothing but the possibilities Donald Knuth has already built into his program. To explain our approach we have to make a short excursion into the hyphenation algorithm.

When a word has to be hyphenated, TEX searches a given set of character sequences (hyphenation pattern set) for all the patterns that occur in this word. Positive numbers are associated with each of the character pairs of each pattern. For all character pairs of the given word, we use the highest number of all the patterns we have found. A simple rule now allows us to decide where hyphenation is possible: if this number is odd, we may hyphenate here, if it is even, hyphenation is forbidden at this position.

It is clear that the algorithm itself is language-independent. In order to apply it to a specific language you have to generate a pattern set for this language. In his thesis at Stanford (1983), Frank M. Liang took a probabilistic approach to convert a hyphenated dictionary of any language into such a pattern set. Based on Webster's Pocket Dictionary and an additional list of more than 1000 words, he created the file `HYPHEN.TEX` that is now part of the TEX distribution tape. With these patterns TEX is able to reproduce about 90% of the hyphens of the original input dictionary, and it will generate no wrong ones. As a consequence, with high probability TEX will also hyphenate correctly English words that are not in the dictionary. But although English and German are quite similar languages, faulty hyphenations will occur frequently in German words, e.g. `ko-r-re-la-tion-s-anal-yse` vs. the correct `kor-re-la-ti-ons-ana-ly-se`.

We had access to a hyphenated German dictionary with more than 127,000 entries, and, as Liang's program `PATGEN.WEB` is on the TEX distribution tape, we were able to generate a hyphenation pattern set for the German language (`GHYPHENU.TEX`). This file is larger than Liang's (6082 vs. 4447 entries), so it is necessary to enlarge the variable *trie_size* in TEX. These patterns give satisfactory results, e.g. TEX hyphenates `kor-re-la-ti-ons-ana-lyse`.

Using a little trick in our hyphenation patterns, we were able to integrate the German *Umlauts* ä, ö, ü, Ä, Ö, Ü and the letter ß into the TEX system. At the same time, we made typing and reading easier.

In plain TEX, *Umlauts* have to be entered in a cumbersome way. `\"` is a macro that sets a trema atop the following letter, so `\"a` gives ä. This is acceptable if needed only rarely, but hard to type in longer German texts. And a word with an *Umlaut* will not be hyphenated because TEX views it as a special character. By catcoding " as an active character (which calls the same macro as `\"`) the input sequence for ä is shortened to `"a`. But ergonomically it would be favourable to interchange both characters to `a"`. We can do this if we follow the advice of the TEXbook, p. 46, and specify in the font metric files that `a` and " should be ligatured, giving ä.

All that remains is to prevent TEX from hyphenating a word between `a` and a following ". Liang, using only the numbers 0 to 5 as hyphenation values, got an acceptable number of breakpoints and, on the other hand, limited the pattern set to a reasonable size. Starting with the same range of numbers, we coded *Umlauts* in the

hyphenation patterns as `a"` (as in the input file) and inserted a hyphenation value of 6 between the two characters. So TEX will never hyphenate at this position.

Of course we now need different pixel files with bitmaps for the *Umlauts* at positions where the ligature information points to. Currently, we can use this method only for output devices with built-in character sets and manually-generated font metric files. We have a version of our pattern set where all entries containing an *Umlaut* are deleted (`GHYPHEN.TEX`). As a temporary solution, we load this for use with the standard TEX fonts until Metafont gives us the ability to create the necessary pixel files.

Although the German character ß (pronounced as sharp s) is not an *Umlaut* it can be handled in the same way. It is coded in the hyphenation pattern set as `s6"`, it can be entered as `s"`, and a ligature specification in the font metric files results in ß. With `\def\3{s"}`, we defined a macro that is easier to type and looks almost like the real ß. This circumvents also the problem of the blank character after control sequences because a ß may appear in the middle of a word and at its end.

Both `GHYPHENU.TEX` and `GHYPHEN.TEX` are now used by many German TEX installations. If you have questions or propositions concerning German hyphenation and *Umlauts*, feel free to contact me.