Software

Another Approach to Multiple Changefiles

Klaus Guntermann and Wolfgang Rülling*
Technische Hochschule Darmstadt

As reported by W. Appelt and K. Horn in TUGboat Vol. 7, No. 1, pp. 20–21, there are several reasons to allow multiple changefiles in the development of a WEB program. These need not to be repeated here. But we did not follow the same approach when we faced the problem. We decided to develop a separate program which we called TIE, since it ties several parts of a WEB together.

Furthermore we allow that a changefile modifies parts that were just changed. The general strategy is that the addition of changefile f_{i+1} behaves as if the changefiles f_1 to f_i had been merged into the WEB program before.

We use a separate program because of the following reasons:

- This single simple program makes additional changes to two programs (namely TANGLE and WEAVE) unnecessary.
- A WEB software developer needs a tool to incorporate frozen changes into a new release of his WEB program from time to time. If the preprocessor program can either create a single changefile or merge all changes into a new WEB file modifications can be written and tested via an additional changefile without touching the released source file. Finally the changes are added ("tied") to the new release.
- TIE can be used for other WEB like systems, too, e.g. for a C version of WEB we created recently. Furthermore TIE allows the application of the changefile method to "plain" Pascal (or even FORTRAN, or whatever programming language you like). One can just merge the changes into the program by selection of the "create new WEB file" option. This is possible since TIE just knows about the line oriented structure of changefiles and has not to deal with the WEB control sequences for sections and so on in the main WEB file. Even data files — as long as they contain textual data — might be changed this way.

The only drawback compared to the method suggested by Appelt and Horn seems to be that it introduces another preprocessing step. This takes some time when the changefiles for large programs like TEX or METAFONT have to be tied since the complete WEB source must be read once more.

Comments to our approach are welcome. TIE is available as a WEB program and can be obtained for a handling charge from Klaus Guntermann at Technische Hochschule Darmstadt.

WEB Adapted to C

Klaus Guntermann and Joachim Schrod
Technische Hochschule Darmstadt

In the UNIX environment the programming language C usually is best developed. For systems programming it seems even to be more suitable than Pascal. This led us to the development of CWEB that allows literate programming in C, giving the full documentation tools that WEB adds to Pascal.

The CWEB processors (simply named by a C- prefix, as is the whole system — there was no common relative whose initials the implementors could choose) CTANGLE and CWEAVE are derived from their WEB counterparts.

Changing TANGLE to build a C program instead of a program for the Pascal compiler was rather straightforward. The only problems occurred with the C preprocessor statements that must be allowed in a CWEB program. These statements are supposed to start on a new line, may span several lines ending with a backslash, and in the last line (which may be the first) no other text is allowed to follow. With new tokens designating start and end of a preprocessor statement it was rather easy to add the necessary rules.

Adapting WEAVE to parse C was a heavier task. One of the reasons is that the beginning of a C function declaration cannot be detected very easily since declaration and call of a function look similar if one does not look ahead very far. The look ahead is nearly impossible if CWEB sections are used for the parameter declaration or the function body. We introduced a new (i.e. in addition to WEB) control sequence @h that marks the start of a function heading in a declaration.

The grammar had to be rewritten completely. We tried to overcome some of the problems that WEAVE has with Pascal formatting if there is not a bunch of explicit formatting commands. The

[*] now at Universität des Saarlandes, Saarbrücken