

Self-replicating macros¹

Victor Eijkhout

Center for Supercomputing Research and Development
 University of Illinois
 305 Talbot Laboratory
 104 South Wright Street
 Urbana, Illinois 61801-2932, USA
 eijkhout@csrd.uiuc.edu

Ron Sommeling

Mathematisch Instituut
 Katholieke Universiteit Nijmegen
 Toernooiveld 1
 6525 ED Nijmegen
 The Netherlands
 sommel@sci.kun.nl

The problem of writing a program that gives its source as its output is one of the oldest conundrums of computer science. (An extended discussion can for instance be found in [1]) The basic idea of any solution is probably to write (in meta-language):

```
Initial_operations;
Print_Twice(Initial_operations;
Close_off;);
Close_off;
```

Of course there is the problem that the procedure ‘Print_twice’ has to be defined, and its call printed, but that’s a minor point . . .

Here are two solutions to this problem in plain T_EX, first one that print itself, in typewriter type, and on an otherwise blank page.

```
\output{}\def\do#1{\catcode\#112}
\def\t{\dospecials\obeylines\tt~}
\def~#1^^:{#1#1^^:\end}\t
\output{}\def\do#1{\catcode\#112}
\def\t{\dospecials\obeylines\tt~}
\def~#1^^:{#1#1^^:\end}\tz
```

The following solution is a variation on the original theme: it gives the source as message on the screen.

```
\catcode 13=12 \newlinechar 13
\def \a #1{\let \#\relax \let \a \relax
\newlinechar 13\immediate \write 16{#1
\catcode \#=12 \a {#1}}\end }
\catcode \#=12 \a {\catcode 13=12
\newlinechar 13
\def \a #1{\let \#\relax \let \a \relax
\newlinechar 13\immediate \write 16{#1
\catcode \#=12 \a {#1}}\end }}
```

The reader may enjoy coming up with more variations, for instance a L^AT_EX document that produces itself, or a plain T_EX document that produces its L^AT_EX source, or . . .

References

- [1] Douglas Hofstadter, Gödel, Escher, Bach, an eternal golden braid. New York 1979.

¹To be published in TUGboat, © 1991, T_EX Users Group.