The file `fdb_latexmk` written by `latexmk`

Ernst Reißner

January 1, 1970

Contents

1 Introduction

A build tool for LaTeX transforms a source file `xxx.tex` into a file `xxx.pdf` or into other formats. To that end, it invokes a bunch of more basic tools, most notably a LaTeX compiler, and possibly more than once. Among the most widely used build tools is `latexmk`.

The manual [Col23] gives only very few pieces information on that file, because it is intended for internal use only; some results of reverse engineering on that file format is collected in [Rei23]. Also, there is some statement of the maintainer, J. Collins, but of course, it may be subject to change. It is the intention of the maintainer that if the form changes, so does the version number described below, so change is not without notification. Since `latexmk` is quite mature, the author of this description considers the risk as quite low.

To find out whether a compilation process must be repeated, `latexmk` keeps track on whether a relevant file changed. As indicated in [Col23], Section "Description", to that end, `latexmk` uses a file `xxx.fdb_latexmk`: it writes information on dependencies before compilation step and records changes after. This is all information [Col23] contains concerning `fdb_latexmk` files.

The understanding of the author of this document is, that `latexmk` uses the file `xxx.fls`, which is generated by LaTeX compilers to detect dependencies, i.e. files the resulting file depends on. The file `xxx.fdb_latexmk` contains pieces of information on files to detect changes and rules on these files, to decide which rules must be reapplied to recreate outdated files.

This document describes the form of these files. In the long run, it is desirable to base this documentation on an analysis of the source code of `latexmk`, but currently, this is just collection of observations and personal communication.

The `fdb_latexmk` files start with a line

```
# Fdb version 4
```

indicating the file version. Maybe this is the major version of `latexmk` which produced it. Let us call this line the *version line*.

After that follow a sequence of productions. Productions are described in Section 2.

Note that lines starting with `#` or `$`, possibly preceded by blanks, are comment lines and as such ignored. So are lines consisting of blanks only, in particular empty lines. Nevertheless, the

1

version is read by `latexmk` and if it is not as expected, the file is not read. This means it is treated as if not existent.

2   Form of a production in `fdb_latexmk` files

A production describes a step of transformation of source files into target files.

The following listing is an example of a production.

```
["bibtex manualLMP"] 1702855070 "manualLMP.aux" "manualLMP.bbl" "manualLMP" 170285508
  "./lit.bib" 1702001904 30636 95f3702a4882e2bee12e1804952c0464 ""
  "/usr/share/texmf/bibtex/bst/base/alpha.bst" 1292289607 23907 a5f93555796fb564b9243
  "manualC1intro.aux" 1702855072 1784 391ed303762a58203532b3a94c1ab1ad "lualatex"
  "manualC2inst.aux" 1702855073 5712 ea0d574b315618b789351136ad49ea23 "lualatex"
  "manualC3usage.aux" 1702855074 11399 d48d9f1b29494e437814a748ac61c875 "lualatex"
  "manualC4graphics.aux" 1702855076 8936 7cd15bd8a63034e99ee74582f42ad927 "lualatex"
  "manualC5procMain.aux" 1702855077 9519 d8d947944e2816ec0f640c46e6d16bae "lualatex"
  "manualC6paramsSettings.aux" 1702855079 14104 e4dda6be3f2449b6d24bf1671c53e42c "lua
  "manualC7ExcLogging.aux" 1702855079 5557 9265ce2a92076ffb1fb0c5a234bad724 "lualatex
  "manualLMP.aux" 1702855079 4747 64f55a1e1685cf1b96b1184a428c034a "lualatex"
  (generated)
  "manualLMP.bbl"
  "manualLMP.blg"
  (rewritten before read)
```

Each production starts with a *rule line* which represents a production rule. The rule line is the fist line of a production, and it is the sole which is not indented. So each production is started with a non-indented line, and each non-indented line except the version line described in Section 1 starts a production.

After the rule line follow a sequence of *source line*s which reach until the literal line "`(generated)`". This introduces the *target lines* which reach until the literal line "`(rewritten before read)`". This follows a list of files, same form as the target files,

So all we have to do now is to describe the rule line, the source lines and the target lines, all without indent.

The rule line represents a production rule, each source line represents a source file, i.e. a file needed to execute the production rule, and each target line represents a file created by the rule.

The rule line starts with a rule in double quotes in brackets. The rest of the rule line is as follows:

- System time at the start of the last time the rule was run.

- Primary source file, e.g., the document's .tex file for pdflatex.

- Primary destination file (if any), e.g., the output .pdf file for pdflatex.

- The base name for files, e.g., log files and the like.

- The system time when the rule was last checked, which is used to decide whether a rerun was needed.

- A status code for the last run: 0 for success. Currently, other possible values are: 1 if the rule failed to create the intended output file, 2 if there was some other kind of error.

The rule may be user/customer defined, but it may also be internal. User defined rules start with `cusdep`, internal rules start with the name of the command executed, sometimes followed by the filename used in the invocation.

A target line is just the name of the file in double quotes.

In contrast to this, a source line contains the following pieces of information separated by a single blank:

- the name of the file in double quotes, as for target lines,

- its modification time,

- its number of bytes,

- its md5 signature,

- the rule name by which this source file is generated in double quotes, or, if it is not generated, the empty string in double quotes.

By means of the rule name, it refers to another production section. The rule name is just what is given in the rule line of some (other) production.

When non-ASCII filenames are encountered, they are coded as UTF-8.

3 References

[Col23] J. Collins. latexmk - generate latex document. available at `https://ctan.org/pkg/lat exmk/?lang=en`, 4 2023.

[Rei23] E. Reißner. The recorder option for latex processors and its applications. `http://www. simuline.eu/LatexMavenPlugin/recorderOption/recorder.pdf`, 4 2023.