

From PostScript to PDF

Herbert Voss

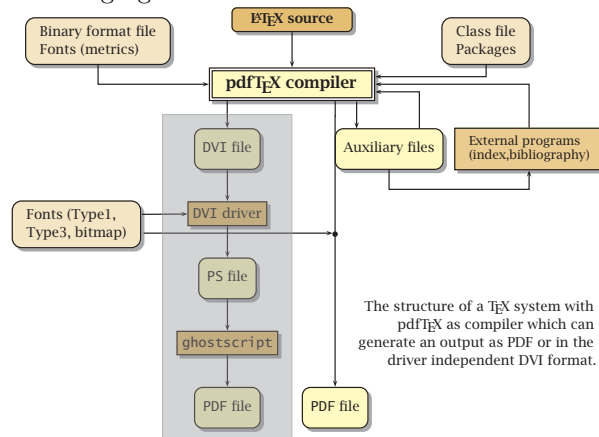
◇ Herbert Voss
Wasgenstrasse 21
Berlin, 14129
Germany
hvoss@tug.org

Abstract

There are still several reasons to use the “traditional” way of creating PDF output, namely the sequence `latex` → `dvips` → `ps2pdf`. Using `pdfLATEX` is only possible when the PostScript related code is handled before the `pdfLATEX` run. Thus, several packages and/or scripts have been developed which supports EPS images, or general PostScript-related code, in a document which is compiled at least one time with `pdfLATEX`: `pst-pdf`, `auto-pst-pdf`, `pdftricks`, `epstopdf`, `pst2pdf`, `pstool`, ... All have the same general goal, but each works in a different way.

1 Introduction

The traditional way for running `TEX` is shown in the following figure.



2 Demonstrations

We will demonstrate with several examples how `eps` images or any other PostScript related code can be used within a `pdflatex` run::

- Using the program `ps2pdf` to generate a `pdf` document
- Using `eps` images with `pdflatex` with Package `epstopdf`
- Using zipped `eps` images with `pdflatex`
- Using other image types with `pdflatex`
- Using `gif` and `png` images with `latex`
- Using `eps` images with `pdflatex` with Package `pstool`
- Using `PSTricks` with `pdflatex` and package `pdftricks`
- Using `PSTricks` with `pdflatex` and package `pst-pdf`
- Using `PSTricks` with `pdflatex` and package `auto-pst-pdf`
- Using `PSTricks` with `pdflatex` and the program `pst2pdf`