

```

.DEFAULT_GOAL := pdf

MD = (wildcardsource/*.md)PDF = output/(notdir (CURDIR)).pdfTEX =
output/(notdir (CURDIR)).texDOCX = output/(notdir (CURDIR)).docxHTML5 =
output/(notdir (CURDIR)).htmlEPUB = output/(notdir (CURDIR)).epubBEAMER =
output/(notdir $(CURDIR))-presentation.pdf

FILFILES = $(wildcard style/*.py) FILTERS := $(foreach FILFILES, $(FIL-
FILES), -filter $(FILFILES)) TEXFLAGS = -F pandoc-crossref -F pandoc-
citeproc -latex-engine=xelatex

ifneq ("(wildcardstyle/template.tex)", "")TEXTTEMPLATE := " -
-template = style/template.tex"endififneq("(wildcard style/reference.docx)", "")
DOCXTEMPLATE := "-reference-docx=style/reference.docx" endif ifneq
("$(wildcard style/style.css)", "") CSS := "-include-in-header=style/style.css"
endif

ifneq ("$(wildcard style/letterhead.md)", "") LH := "style/letterhead.md" endif

help: @echo " @echo 'Makefile for automated typography using pandoc.' @echo
'Version 1.1' @echo " @echo 'Usage:' @echo 'make prepare first time use, setting
the directories' @echo 'make html generate a web version' @echo 'make pdf
generate a PDF file' @echo 'make docx generate a Docx file' @echo 'make tex
generate a Latex file' @echo 'make beamer generate a beamer presentation'
@echo 'make all generate all files' @echo 'make update update the makefile to
last version' @echo 'make will fallback to PDF' @echo " @echo 'It implies some
directories in the filesystem: source, output and style' @echo 'It also implies
that the bibliography will be defined via the yaml' @echo " @echo 'Depends on
pandoc-citeproc and pandoc-crossref' @echo 'Get local templates with: pandoc
-D latex/html/etc' @echo "

all : tex docx html5 epub pdf

pdf: $(LH) $(PDF) $(PDF): $(MD) pandoc -o $$@ source/*.md $(TEXTTEM-
PLATE) (TEXFLAGS)2 > output/pdf.logif[["OSTYPE" == "darwin" ]]; then
open $$@; else xdg-open $$@;fi

$(LH): pandoc -s -o style/letterhead.pdf $(LH) $(TEXFLAGS)

tex: $(TEX) $(TEX): $(MD) pandoc -o $$@ source/*.md (TEXFLAGS)2 >
output/tex.logif[["OSTYPE" == "darwin" ]]; then open $$@; else xdg-open $$@;fi

docx: $(DOCX) $(DOCX): $(MD) pandoc -o $$@ source/*.md $(TEXFLAGS)
(DOCXTEMPLATE) --toc2 > output/docx.logif[["OSTYPE" == "darwin"
]]; then open $$@; else xdg-open $$@;fi

html5: $(HTML5) $(HTML5): $(MD) pandoc -o $$@ source/*.md $(CSS)
(TEXFLAGS) --toc2 > output/html5.logif[["OSTYPE" == "darwin" ]];
then open $$@; else xdg-open $$@;fi

epub: $(EPUB) $(EPUB): $(MD) pandoc -o $$@ source/*.md (TEXFLAGS) --
toc2 > output/epub.logif[["OSTYPE" == "darwin" ]]; then open $$@; else

```

```

xdg-open $@;fi

beamer: $(BEAMER) $(BEAMER): $(MD) pandoc -o $@ source/*.md
(TEXFLAGS) - -toc -tbeamer2 > output/beamer.logif[["OSTYPE" ==
"darwin" ]]; then open $@; else xdg-open $@;fi

prepare: command -v pandoc >/dev/null 2>&1 || { echo "I require pandoc
but it's not installed. Aborting." >&2; exit 1; } command -v pandoc-crossref
>/dev/null 2>&1 || { echo "I require pandoc-crossref but it's not installed.
Aborting." >&2; exit 1; } command -v pandoc-citeproc >/dev/null 2>&1 || {
echo "I require pandoc-citeproc but it's not installed. Aborting." >&2; exit 1; }
mkdir "output" mkdir "source" mkdir "style"

update: wget http://tiny.cc/mighty_make -O Makefile

clean: rm -f "output/" .md .html .pdf .tex .docx .epub

.PHONY: help prepare update beamer clean

```