

Latex con make

Por Paco Aldarias Raya

Impreso: 21 de noviembre de 2004

Email: pacolinux@inicia.punto.es

Web: <http://www.iespana.es/pacolatex>

Con Linux Debian. En Valencia (España)

Este documento es de libre reproducción siempre que se cite su fuente.

Realizado con: 

Índice

Índice	1
1. Introducción	1
2. Varios comandos en una regla del fichero Makefile	1
3. Como make procesa makefile	2
4. Comandos del make	2
5. Ejemplos en latex	3
5.1. Ejemplo con genéricos y varios pasadas en latex	3
5.2. Ejemplo con genéricos	3
6. Ejemplos en c	4
6.1. Ejemplo simple en c	4
6.2. Ejemplo de programa en c	5
6.3. Ejemplo de programa en c con variables	5
7. Make para latex con plantilla	6
7.1. makefile.sh	6
7.2. makefileplantilla	6

8. Mas información	7
9. Bibliografía	8

1. Introducción

Sacado del man make:

”Make es una utilidad de GNU para mantener grupos de programas.”

Es necesario tener el fichero makefile, para que llamar al make desde consola sepa que hacer.

2. Varios comandos en una regla del fichero Makefile

La forma en la que se escriben las reglas es esta:

```
etiqueta : [dependencia1 ...] [;comandos] [# comentarios]
[(tabulación) comandos] [# comentarios]
```

o

```
objetivo: requisito
comando1; comando2; comando3
```

o

```
objetivo: requisito
comando1;
comando2;
comando3
```

o

```
destino ... : prerequisite ..
comando
comando
comando
```

Siendo

destino es el nombre del fichero q es generado por make.

prerequisito es el fichero que es usado como entrada para crear destino

comando es una acción que lleva a cabo.

Nota: Una regla puede tener más de una línea. Debemos poner una tabulación al principio de cada comando. Cada línea se ejecuta en paralelo con las demás.

3. Como make procesa makefile

Por defecto make comienza con el primer destino. Esto es llamado destino por defecto. Puede que los otros destinos o targets se activen si forman parte de los prerequisites.

Si queremos que make solo haga un destino debemos poner `make destino`

4. Comandos del make

1. `make -f mimake`
-f permite especificar el fichero de configuración.

5. Ejemplos en latex

5.1. Ejemplo con genéricos y varios pasadas en latex

```
TEX = latex
%.dvi:%.tex # pattern rule para LaTeX
$(TEX) $< ; $(TEX) $<
```

5.2. Ejemplo con genéricos

```
all: tutorial.ps plantilla-1.ps plantilla-2.ps transparencias.ps \
tutorial-2pag.ps plantilla-2-tex.ps
```

```
clean:
rm -f tutorial.ps tutorial-2pag.ps plantilla-1.ps \
plantilla-2.ps transparencias.ps tutorial-2pag.ps \
plantilla-2-tex.ps *.dvi *.log *.aux *.toc *~
```

```
# Código plantilla a transparencia
plantilla-2-tex.ps: plantilla-2.tex
a2ps -p -8 -1 -B plantilla-2.tex > plantilla-2-tex.ps

# Tutorial a 2 paginas
tutorial-2pag.ps: tutorial.ps
mpage -A2 $< > $@

# Generica para generar postscript
%.ps: %.dvi
dvips -t a4 -o $@ $<

# Generica para generar dvi
%.dvi: %.tex
latex $<
latex $<
latex $<
```

6. Ejemplos en c

6.1. Ejemplo simple en c

```
edit : main.o kbd.o command.o display.o \
      insert.o search.o files.o utils.o
      cc -o edit main.o kbd.o command.o display.o \
          insert.o search.o files.o utils.o

main.o : main.c defs.h
      cc -c main.c
kbd.o : kbd.c defs.h command.h
      cc -c kbd.c
command.o : command.c defs.h command.h
      cc -c command.c
display.o : display.c defs.h buffer.h
      cc -c display.c
insert.o : insert.c defs.h buffer.h
      cc -c insert.c
search.o : search.c defs.h buffer.h
      cc -c search.c
files.o : files.c defs.h buffer.h command.h
```

```
        cc -c files.c
utils.o : utils.c defs.h
        cc -c utils.c
clean :
        rm edit main.o kbd.o command.o display.o \
            insert.o search.o files.o utils.o
```

Con variables:

```
objects = main.o kbd.o command.o display.o \
            insert.o search.o files.o utils.o

edit : $(objects)
        cc -o edit $(objects)
main.o : main.c defs.h
        cc -c main.c
kbd.o : kbd.c defs.h command.h
        cc -c kbd.c
command.o : command.c defs.h command.h
        cc -c command.c
display.o : display.c defs.h buffer.h
        cc -c display.c
insert.o : insert.c defs.h buffer.h
        cc -c insert.c
search.o : search.c defs.h buffer.h
        cc -c search.c
files.o : files.c defs.h buffer.h command.h
        cc -c files.c
utils.o : utils.c defs.h
        cc -c utils.c
clean :
        rm edit $(objects)
```

6.2. Ejemplo de programa en c

```
lista : lista.o main.c lista.h
gcc lista.o creditos.o main.c -o lista

lista.o : lista.c
```

```
gcc -c lista.c
```

```
creditos.o : creditos.c
gcc -c creditos.c
```

6.3. Ejemplo de programa en c con variables

```
FILES= flectura.c fagnadir.c agnadir.c ini.c \
estad.c varios.c mensaje.c flujo.c\
describe.c juntar.c selec.c ordenar.c
```

```
OFILES=flectura.o fagnadir.o agnadir.o ini.o \
estad.o varios.o mensaje.o flujo.o\
describe.o juntar.o selec.o ordenar.o
PFILE= main.c
LIBS= -lm
NOMBRE= pp
```

```
all: $(OFILES)
    gcc -g -o $(NOMBRE) $(OFILES) $(PFILE) $(LIBS)
```

```
comp: $(FILES)
    gcc -g -c $(FILES)
```

```
save:
    tar cvfz /home/iolalla/pp.tgz /home/iolalla/pepe
```

7. Make para latex con plantilla

7.1. makefile.sh

```
echo "Uso: ./makefile.sh fichero [pdf|html|..]"
```

```
cp makefileplantilla $1_make
sed -e "s/fichero/${1}/g" $1_make > makefiletemp
mv makefiletemp $1_make
```

```
echo "make -f" $1_make $1.$2
make -f $1_make $1.$2
```

7.2. makefileplantilla

```
TEX=latex
f=fichero.tex fichero_datos.tex

fichero.all: fichero.dvi fichero.pdf fichero2.pdf \
fichero.html fichero.zip

fichero.mio: fichero.dvi fichero.pdf fichero.html

fichero.latex: $(f)
    $(TEX) fichero.tex;

fichero.dvi: $(f)
    $(TEX) fichero.tex;
    makeindex fichero.idx;
    $(TEX) fichero.tex;

fichero.bib: $(f) fichero.bib
    $(TEX) fichero.tex;
    bibtex fichero;
    makeindex fichero.idx;
    $(TEX) fichero.tex;

fichero.pdf: $(f)
    pdflatex fichero.tex;
    xpdf fichero.pdf;

fichero.html: $(f)
    latex2html fichero -no_subdir -split 0 \
-local_icons 1 -show_section_numbers -no_auto_link

fichero2.pdf: fichero.pdf
    pdftops fichero.pdf;
    pstops "2:0L@.7(21cm,0)+1L@.7(21cm,14.85cm)" \
fichero.ps fichero2.ps;
    ps2pdf fichero2.ps;
```

```
rm fichero2.ps fichero.ps;

fichero.zip: fichero.tex
    zip -u $(f)

fichero.clean:
    rm -f fichero.log fichero.ps \
fichero.dvi fichero.out fichero.aux

fichero.txt: fichero.pdf
pdftotext fichero.pdf
```

8. Mas información

```
gunzip -c /usr/share/info/make.info-1.gz | less more
gunzip -c /usr/share/info/make.info-2.gz | less more
gunzip -c /usr/share/info/make.info-3.gz | less more
...
```

9. Bibliografía

1. Introducción al programa make
<http://www.unav.es/cti/curso-c/make.html>