GNU Make

history

syntax

usage

GNU Make

Richard Stallman

Roland McGrath

syntax

target

dependency

up to date

makefile

What a Rule Looks Like

A simple makefile consists of "rules" with the following shape:

target ... : prerequisites ...
command
...
...

sample makefile

```
# a sample makefile

# targets begin at the left margin and followed with ":"

# shell command lines begin with a tab, often overlooked

stimulate:

# one or more commands to create

        stimulate gcc -o stimulate -O stimulate.c inputs.c outputs.c

stimulate.db:
        gcc -DDEBUG -g -o stimulate.db stimulate.c inputs.c outputs.c

inputs.do:
        inputs.c headerfile.c gcc -o inputs.do -c -DDEBUG -g inputs.c

clean:
        rm *.o *.do stimulate stimulate.db
```

another with macros

```
# another make file
# with macros

DEPENDS= inputs.o outputs.o stimulate.o
DBDEPENDS= inputs.do outputs.do stimulate.do

stimulate: $(DEPENDS)

   gcc -o $@ $(DEPENDS)

stimulate.db: $(DBDEPENDS)
  gcc -o $@  $(DBDEPENDS)
```

invoking make

checks to see if files on which target depends are up to date
creates a new version of any file that is out of date
checks if target is up to date on files which it depends on
creates a new version of the target if out of date

```
$ make

$ make clean
```