

# Correct without manual dependencies.

## Fast without incremental rules.

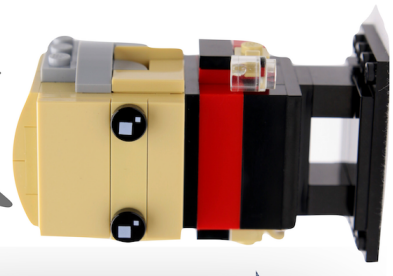
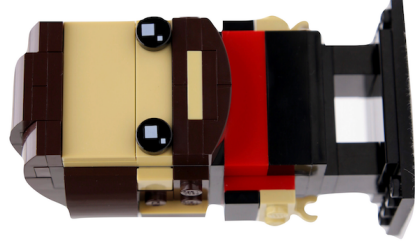
ATC '22  
BEST PAPER

```
● ● ● Makefile
program: main.o x.o y.o
gcc -o program main.o x.o y.o
main.o: main.c x.h y.h
gcc -c -o main.o main.c
x.o: x.c
gcc -c -o x.o x.c
y.o: y.c y.h
gcc -c -o y.o y.c
```

Don't make...

```
● ● ● Rikerfile
#!/bin/sh
gcc -o program *.c
```

...make it so!

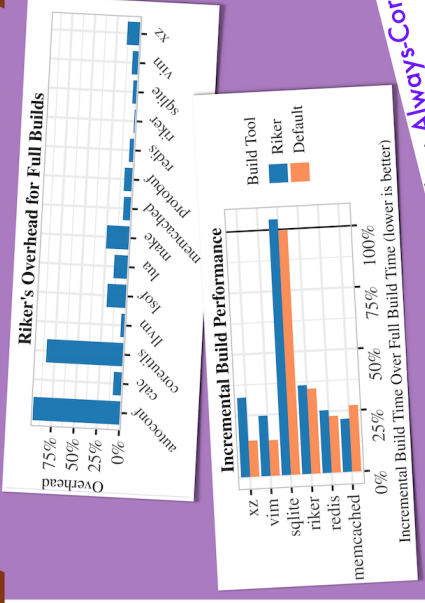


Riker is a novel forward build tool for always-correct, incremental builds using simple scripts in any language.

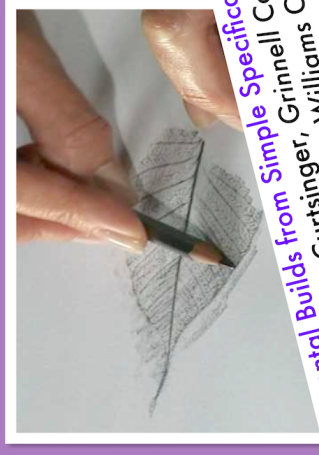
TraceR: linear, binary dependence log

```
cc1_1 = Launch(gcc_0, "cc1 x -o tmp.s", [...])
r71 = PathRef(cc1_1, CWD, "x.c", r--)
ExpectResult(cc1_1, r71, SUCCESS)
MatchMetadata(cc1_1, r71,
[uid=100, gid=100, type=file, perms=rw-r--])
MatchContent(cc1_1, r71,
[mtime=1619457130, hash=3c6ea, cached=false])
r75 = PathRef(cc1_1, r3, "tmp.s",
-r- truncate create (rw-rw-r--))
ExpectResult(cc1_1, r75, SUCCESS)
UpdateContent(cc1_1, r75, [hash=054521])
```

Performance comparable to make



Tracing never misses a dependency



Build Tool  
Riker  
Default

Charlie Curtsinger, Grinnell College  
Daniel W. Barowy, Williams College

<https://rkr.sh>

