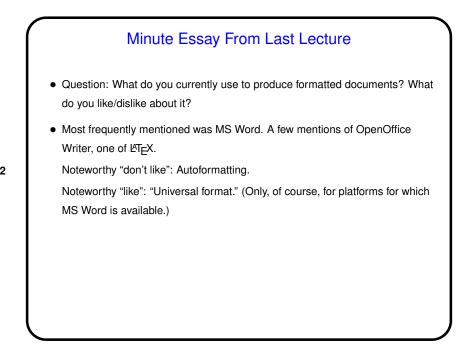


Slide 1

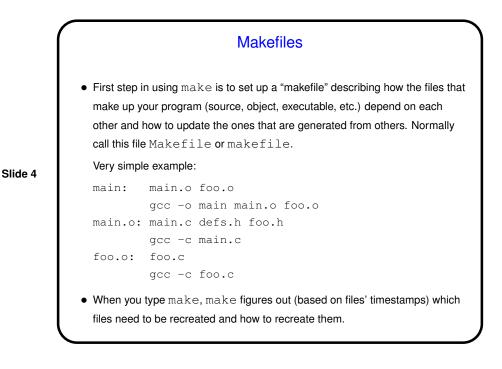


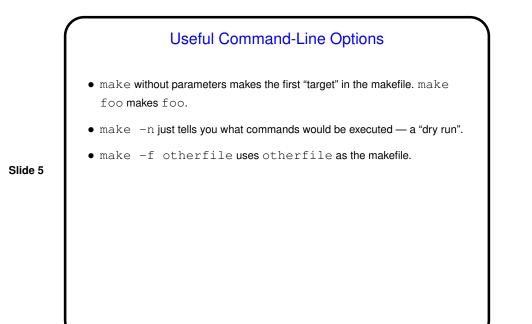


 Motivation: Most programming languages allow you to compile programs in pieces ("separate compilation"). This makes sense when working on a large program — when you change something, just recompile the parts that are affected.

Slide 3

• Idea behind make — have the computer figure out what needs to be recompiled and issue the right commands to recompile it.





Defining Rules
• You define dependencies for a rule by giving, for each "target", a list of files it depends on.
• You also give the list of commands to be used to recreate the target. NOTE!: Lines containing commands must start with a tab character. Alleged paraphrase from an article by Brian Kernighan on the origins of Unix:
The tab in makefile was one of my worst decisions, but I just wanted to do something quickly. By the time I wanted to change it, twelve (12) people were already using it, and I didn't want to disrupt so many people.

Phony Targets
• Normally targets are files to create (e.g., executables), but they don't have to
be. So you can package up other things to do ...
• Example — many makefiles contain code to clean up, e.g.:
clean:
 -rm *.o main
To use — make clean.

