Administrivia
Reminder: "Not accepted past" date for Homeworks 1 through 7 is today at 5pm.
Final exam period is December 15 at 8:30am. We will use this time for project presentations. At 10 minutes each, we won't need the full three hours. Start at 9:30am?
Project-related material (reports and "deliverables" due at exam time. Not accepted late.
Homework 8 due date extended to exam time. Not accepted late.
Information about office hours this week and next coming by e-mail soon.

Course Wrap-Up — What I Hope You Got From This Class (Details)

- More things in your "bag of tricks" (see later slide).
- Practice in reading man pages and otherwise learning more.
- Exposure to traditional tools you might need, or want, to use sometime. You may not remember details, but I hope you will be less intimidated.

Slide 2

Slide 1

Course Wrap-Up — What I Hope You Got From This Class (Big Picture)

• Exposure to a different operating system / user interface paradigm — many small programs that work together, information kept in text files, emphasis on being expert-friendly and scriptable, etc.

Slide 3

• Encouragement to find out how to use all your tools as intelligently as possible.

"A tour of UnixWorld / TextWorld."

"More than one way to do things."

I like the old tools because I know how to make them work together. But it's worth noting that many "modern" tools (GUI-based programs, graphical file managers, etc.) have their own way of working together — common set of keybindings, cut-and-paste metaphor, drag-and-drop, multiple selections, etc. Compare and contrast!

Course Wrap-Up — Topics

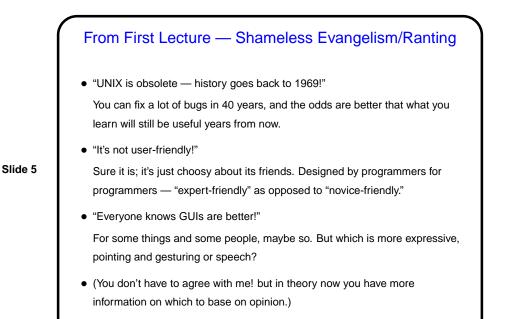
- Shell features command history, redirecting input and output, scripting features (if/then/else and loops).
- Pipes.
- Filter programs (awk, sed, grep, etc.).

Slide 4

● L^AT_FX.

• Text editors.

- make and makefiles.
- Regular expressions (for text editing, grep, etc.).
- Miscellaneous other text-mode tools mail, screen, games, etc., etc.
- A little about X, installing software, and CGI scripting.



From First Lecture — The UNIX Philosophy

• As stated by one of its developers (Doug McIlroy):

"Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface."

Slide 6

• There's more, but the emphasis is on (1) providing a set of lightweight tools that can be put together to do interesting things, and (2) providing choices to users (sometimes almost too many!)

