Why window systems?

- Increased usability due to
  - Visibility
    - Graphical representation of programs
    - See multiple environments at once
  - Direct manipulation
- Enables powerful graphics programs
Window systems and UNIX

- Most UNIX users can be considered experts, and are fiercely protective of the command line (for good reason)
- However, all current UNIX systems have a built-in window system - the advantages are inescapable
X Windows

- Practically all UNIX window systems are based on X Windows
- The standard version is called X11
- Very complex system with many parts
- Basically, X11:
  - Manages the screen space
  - Draws simple graphics
  - Assigns rectangular regions to various programs
X’s client-server architecture

- X is actually meant to work over the network
- X server: software that runs on the machine where the program’s output will be displayed
- X client: program running on the same or another machine
- Client sends drawing and other X commands to the server, which displays the results
Historical use of X

- Users sat at “X terminals” - graphical terminals that only knew how to run an X server
- They logged in to other UNIX machines remotely and ran X clients there
- This gave users the benefits of a window system without the need for a full-featured computer on every desk
Features of X

- Transparent remote execution
- Gives programs their own virtual screen
- Includes important windowing concepts:
  - Window damage
  - Window reveal events
  - Backing store
- X11 programs are highly portable
Window managers

- Not part of X11 itself
- Run on top of X11
- Place borders, decorations on windows
- Handle input from users
- There are many, many choices with different “look & feel”
Desktop environments

- Yet another layer, running on top of window managers
- Complete the desktop metaphor with:
  - Iconic access to files and directories
  - Overall system menus / toolbars
  - Trash can
  - etc.
- GNOME is one example