

Operating Systems Design
 24. Windowing
 Paul Krzyzanowski
 pxk@cs.rutgers.edu

User Interfaces: 1st Generation

Historically, the **command-line interface**


- Still great for scripting, systems management, remote access, and customized operations

```
cat *.txt | tr -cs "[alpha:]" "\n" |
tr A-Z a-z | sed '/^$/d' |
sort | uniq -c | sort -nr
```

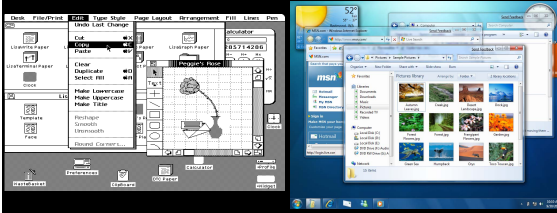
User Interfaces: 2nd Generation

Most users prefer a graphical UI

- Dominant interface:
 - desktop metaphor
 - **WIMP** (Window, Icon, Menu, Pointer) design paradigm
 - 1964-1968: Douglas Englebart
 - 1968 demo: mouse, windows, hypermedia links, video conferencing
 - 1973: Xerox Alto – PC with GUI, folders, mouse, keyboard



WIMP



User Interfaces: 3rd Generation

- Touch (& multitouch) interactive
 - No windows, mouse, pointer
 - Jeff Han, NYU: Multitouch sensing, 2006
 - Huge mindshare due to the popularity of the iPhone & iPad



Hardware for graphics

- Fundamental interface
 - **Framebuffer**
 - Memory buffer containing a video frame
 - Memory mapped into system's memory space
- **Graphics accelerator (GPU)**
 - Send drawing commands to the GPU, which rasterizes the results onto a framebuffer
 - Abstraction libraries: *OpenGL*, *DirectX/Direct3D*
 - Provide a uniform interface for hardware graphics
 - Translate commands into GPU-specific commands
 - GPUs are multithreaded; driver may control thread scheduling
 - GPU's results are sent to the framebuffer

Windowing System

- Interfaces with mice, keyboards, cursor, & graphics HW
- Provides virtual interfaces to processes
 - Virtual screen (framebuffer)
 - Virtual keyboard
 - Virtual mouse

The diagram shows three boxes labeled 'Process' at the top. Below them is a larger box labeled 'Window System'. Bidirectional arrows connect each 'Process' box to the 'Window System' box. Below the 'Window System' box are three labels: 'displays', 'keyboard', and 'mouse'. Bidirectional arrows connect the 'Window System' box to each of these hardware labels.

Window Manager

- Handles interactions between windows, applications, and the underlying windowing system
- Does not interact with the hardware
- **Stacking (floating) window manager**
 - Draws windows in a specific order (sorted by z-order)
 - Allow overlapping windows by drawing background windows first
 - Contents have to be redrawn when window new parts exposed
 - Limited ability to accelerate with a graphics card
 - Limited ability to accelerate with a graphics card
 - Used in X Windows & Windows XP
- **Compositing window manager**
 - Windows drawn separately. Graphics HW places them in a 2D or 3D environment
 - OS X, Vista and Windows 7 use this
- **Hybrid**: treat foreground window differently: have graphics card render it

Kernel Interface: Windows ≥ Vista

Windows Display Driver Model (WDDM)

The diagram is divided into 'User Mode' and 'Kernel Mode'. In User Mode, there are 'Windows Presentation Framework (WPF)' and 'Desktop Window Manager (DWM)'. WPF is associated with 'Programming interface: rendering graphics, typography, media'. DWM is associated with 'Desktop: Aero GUI & theme'. Below these are 'Media Foundation' and 'Other components'. 'Media Foundation' is associated with 'Media services & rendering'. Below these are 'DirectX Video Acceleration', 'Direct3D 9', 'Direct3D 9Ex', 'Direct3D 10', and 'OpenGL'. All these components connect to a 'Device Driver Interface (DDI)'. The DDI connects to 'OpenGL driver', 'DirectX driver', and 'Kernel Driver'. The 'Kernel Driver' is associated with 'supplied by hardware vendor'. In Kernel Mode, there is 'Win32 Kernel' and 'DirectX Graphics'. The 'Kernel Driver' connects to 'DirectX Graphics', which in turn connects to 'Win32 Kernel'.

Windows Display Driver Model

- **Virtual video memory (memory protection)**
- **GPU thread scheduling**
- **Lots of rendering APIs**
 - Legacy: DirectDraw, Direct3D (3..8)
 - Mainline: GDI, Direct3D 9/9Ex, OpenGL
 - New: Direct3D 10, Windows Presentation Foundation
- **Separate rendering from device management**
 - Direct3D 10 manages graphics
 - DXGI component manages
 - Adapters, display modes, output, gamma/color, monitor controls
- **Desktop Window Manager**
 - Compositing desktop

Virtual desktop

- Large virtual desktop (64K × 64K)
- Portions are mapped to monitors through views

The diagram shows a large blue rectangle labeled 'Virtual desktop'. To its left is a smaller monitor icon. An arrow points from the monitor icon to a small rectangular area on the virtual desktop, which shows a view of the desktop content (windows, taskbar, etc.).

X Window System (X11)

- Window system
 - User-level interface to hardware
 - Manages graphics card, keyboard, and mouse
 - I/O multiplexing
 - Client-server API
 - Create/destroy windows
 - Basic drawing (text, lines, fills) commands into windows

The diagram shows a central box labeled 'X Server'. Below it are three labels: 'displays', 'keyboard', and 'mouse'. Bidirectional arrows connect the 'X Server' box to each of these hardware labels. Above the 'X Server' box are four boxes: 'Window manager', 'xterm', 'opera', and 'xterm'. Bidirectional arrows connect the 'X Server' box to each of these client boxes. A box labeled 'Optional component: renders desktop, responsible for window frames, icons, task bars, etc.' has an arrow pointing to the 'Window manager' box. A box labeled 'remote system' has an arrow pointing to the 'xterm' box. A box labeled 'Some events are redirected to the window manager (e.g., create/move windows)' has an arrow pointing to the 'Window manager' box. A box labeled 'TCP/IP' has an arrow pointing to the 'xterm' box.

X Windows

- **X Server**
 - Provides *mechanism*, not *policy*
 - Provide windows, drawing primitives, cut buffers, text rendering
- **Window manager**
 - Application that runs on X
 - Controls the placement & appearance of windows, icons, ...
 - fvwm, 3dwm, afterstep, Window Maker, Enlightenment, ...
- **Widget Libraries** (Toolkits, APIs)
 - Common UI components: scrollbars, sliders, dialog boxes, ...
 - Gtk, At, LessTif
- **Desktop environments**
 - Window manager + applications to provide consistent UI (program launchers, ...)
 - GNOME, KDE Software Compilation, CDE, ...

13

The End

14