

X Window System

Copyright © 1998 P. Tobin Maginnis

This document is free; you can redistribute it and/or modify it under the terms of the [GNU General Public License](#) as published by the Free Software Foundation.

1. **Describe the evolution of the X-Window system and how explain how this relates to Linux.**
 - i. Project Athena 1985 - MIT and DEC - Public Domain
 - ii. X/Open Company 1984
 - iii. Open Software Foundation 1988
 - iv. The Open Group 1996 [X-Window](#), [Return to Public Domain](#) and [Single Unix Specification](#)
 - v. The XFree86 Project, Inc 1995 [X-Servers](#)
 - vi. Desktop managers 1998 [The KDE desktop](#)
2. **Explain why a typical MS-Windows user is "unaware" of the graphic adapter details while a typical Linux user usually knows the details of the graphic adapter.**

The PC system architecture includes firmware on the motherboard (BIOS) and on various adapters that hide the details and differences among graphic adapters. At system boot up, the BIOS performs initialization and testing of the motherboard and system memory. The BIOS routines also search the "I/O space" for additional firmware initialization routines. Thus, the very first message displayed at the top of the monitor screen after power up comes from the on-board graphic adapter firmware (BIOS) initialization routines. MS-DOS and MS-Windows access the graphic adapter through these on-board BIOS (firmware) routines.

Graphic adapter manufactures also provide custom device drivers that can access special adapter acceleration features such as a "block move" operation that quickly moves a sprite (small bit-map) around video memory. The custom device drivers are specific to the type of system bus (ISA, EISA, MCA, VLB, and PCI) and type of OS (MS-Windows, OS/2, and Novell) used by the adapter. (Manufactures used to also provide drivers for MS-DOS graphic applications such as AutoCAD, Lotus 1-2-3, and WordPerfect.) Users simply click "install" and the correct video device drivers are copied to the system. Although convenient, drivers usually allow just the standard modes described below.

Graphic adapter manufactures typically do not write video device drivers for Unix or Linux. It is therefore up to the XFree86 Project to develop drivers for each new graphic adapter as it appears in the market. Taking their job seriously, X server (device driver) developers provide many more options for a given adapter than the "standard drivers" provided by the manufactures. Instead of being limited to the few standard video modes, the Linux user is

able to select a graphic adapter and monitor and configure them for optimal speed, resolution, and "visual comfort."

3. Describe the role of an "X server" and its relationship to other system components.

In Linux (Unix) the GUI executes as a user-level process outside of the kernel proper. The X server is the lowest-level module in the GUI. It is hardware (video adapter) specific and its job is to directly establish a graphic mode, wait for keyboard and mouse events, and update the display.

4. Describe the general GUI algorithm and define "focus."

- a. Load a full-screen graphic, set a color, or wall-paper a small graphic as the desktop image.
- b. Render boxes and buttons and/or load bit-mapped graphic icons.
- c. Load or convert to character bit-maps from font library for text on buttons, icons, and title bars.
- d. Render the mouse cursor and wait for events.
- e. As the mouse moves, calculate new cursor x,y position, save the desktop background, move the cursor to its new position, and restore old position background.
- f. If the cursor is over a hot spot or button, load or render the "highlighted" version of the icon or button. Variations include no highlight or highlight after mouse click.
- g. If a mouse click occurs while in the hot spot, activate the associated program. Variations include showing a 3D effect such as button depression, window movement, or icon transformation before program activation.

The term focus refers to which program on the desktop should receive keystrokes. The currently active program has its window frame highlighted to indicate it has the focus. Focus can be switched among programs with cursor movement to a new window, cursor movement to a new window plus a mouse click, or with a control key combination such as <Alt-Tab> where <Alt> is the depression and holding of the "Alternate" key and <Tab> is the depression of the "Tab" key. But note, if the key combination is held too long, the focus will quickly cycle among all active desktop programs.

5. Describe the general evolution of PC graphic adapters.

EGA 570x480x16, VGA 640x480x16, SVGA 640x480x256, EVGA 1024x768x256, VESA (see below)
ET, S3, Trident, Oak, ATI, #9 and countless others.

6. Contrast the basic elements of video adapter card.

- . Shared memory
- I. Video controller
- II. LUT
- III. DAC
- IV. Clock dot generator

7. **Contrast bit planes *versus* pixel depth.**
8. **Define "mode," explain how it interacts with BIOS or VESA "graphic modes," give some common examples, and explain how the clock dot generator interacts with the graphic mode.**
9. Mode Resolution Horizontal Vertical
10. 0x12 - 640x480x16; 31.47 KHz 60 Hz
11. 0x101 - 640x480x256;
12. 0x103 - 800x600x256; 37.88 KHz 60.31 Hz
13. 0x105 - 1024x768x256; 78.9 KHz 74 Hz
14. 0x110 - 640x480x32K
15. 0x111 - 640x480x16M; 37.86 KHz 72 Hz
16. 0x113 - 800x600x32K; 57.9 KHz 72 Hz
17. 0x115 - 800x600x16M; 56.5 KHz 70 Hz
18. 0x116 - 1024x768x32K; 81.2 KHz 76 Hz
19. 0x118 - 1024x768x16M;
20. **Describe the scan circuitry of a monitor, explain the term "multi-sync," explain why the monitor may chirp or click, and explain why the picture tube may have to adjusted several times. Explain how one can break a monitor.**
21. Monitor Dots Display area Addressability Horizontal Vertical Combined
22. Samsung 14" .39mm 640x480 31.5 KHz 60-70 Hz
23. Samsung 14" .28mm 1024x768 31-35 KHz 60-70 Hz
24. Seiko 14" .25mm 240x180 mm 1024x768 31-50 KHz 50-90 Hz
25. old Magi 17" .28mm 300x225 mm 1024x768 24-64 KHz 55-90 Hz 100 Mhz
26. new Magi 17" .28mm 322.8x242 mm 1280x1024 30-69 KHz 50-120 Hz 100 MHz
27. Seiko 20" .31mm 360x270 mm 1280x1024 30-77 KHz 50-90 Hz
28. Sony 20" .30mm 350x280 mm 1280x1024 28-85 KHz 50-160 Hz
29. **Contrast addressability *versus* resolution and monitors *versus* LCD displays.**

Addressability refers to the programmers digital view of the video adapter. The display area is abstracted as a matrix of x (row) and y (column) coordinates with each matrix "square" containing a pixel color value.

A monitor has an "unknown" number of red, blue, and green phosphor "dots" that are grouped into trios of phosphor "units." These units may or may not represent a pixel. Thus, resolution refers to the analog nature of the monitor where a given number of "square" digital pixels are spread across or squeezed and dropped among the available "round" phosphor units.

The disadvantage of this analog process is that the monitor will accept large addresses and automatically drop unusable pixels creating a "blurry" image. The advantage of this analog process is that any addressability is automatically "scaled" to a full screen image.

Liquid Crystal Displays (LCDs), on the other hand, have "known" number of display "units" that are directly addressable. As a result, there are no longer blurry images, but automatic image scaling to full screen is generally not done since it is not built into the current generation of video drivers.

30. **Contrast the X Window "virtual desktop" *versus* "viewport."**

The X server exploits "unused" video RAM by filling it with an "offscreen" desktop. The user can pan through the virtual desktop fvwm by clicking on the virtual desktop map area, or by pushing the cursor offscreen in afterstep.

The viewport is simply the starting point within the virtual desktop. Usually the desktop is divided into quadrants and "ViewPort 0 0" is the upper left quadrant.

31. Since the X server is hardware specific, explain why the XF86Config file is required and describe its six sections.

A given X server does know the hardware, but it does not know how the user wants the hardware configured. Moreover, even though some "ideal" parameters can be obtained from the hardware at run-time, parameters in the configuration file are given precedence since they may have to override a broken feature or they may have critical timing values that are not subject to system load induced changes.

Each section of the XF86Config file is delimited the key words Section "name" and EndSection, and the six sections are:

0. Section "Files" - Resource file names of color values and fonts.
1. Section "Keyboard" - X server run-time flags such as "do not allow keyboard termination."
2. Section "Pointer" - List the mouse protocol, Baud, and button emulation.
3. Section "Monitor" - List the monitor horizontal and vertical frequency ranges and the wave shape for a given dot clock frequency.
4. Section "Device" - List the possible dot clock frequencies for graphic modes and adapter-specific hardware options.
5. Section "Screen" - List possible graphic modes, virtual desktop size and initial viewport.

32. Describe the command `X > /tmp/log 2>&1` and explain its role.

X is a symbolic link to a hardware specific driver that can access the video graphics array (VGA) adapter. The X server reads the file XF86Config for settings and probes the adapter for additional features. The command runs the X server in "stand alone" mode and redirects its diagnostic, probe, and configuration messages to the file /tmp/log.

There are two formats in bash for redirecting both standard output and standard error `&> /tmp/log` and `>& /tmp/log`. Of the two forms, the first is preferred and both forms are equivalent to the above form `> /tmp/log 2>&1` where standard out was redirected to /tmp/log and standard error was subsequently redirected to standard out.

The role of this command is to test the X server configuration. The log file contains a description what the server discovered when probing the adapter and the subsequent setting of the adapter. If a valid graphic mode can be found, a gray chain-link desktop wall paper appears with an outlined "X"

cursor centered on the screen. Moving the mouse should move the cursor. The resulting configuration is saved in the log file.

The Ctl-Alt-Backspace key combination will terminate the X server and return the screen to "text" mode. The results of the adapter probe can be seen by viewing the /tmp/log file.

33. Describe the relationship among virtual terminals and the GUI and describe how to control the GUI modes.

Generally, six virtual terminals are defined (F1 through F6). The GUI can be started from any of the six with any of the following commands:

0. init 5 - Switch to run-level five and automatically start xdm.
1. xdm - Manually startup the login graphic.
2. startx - Manually begin an X-Window desktop.
3. xinit - Begin an X-Window desktop with specific arguments.

Once the X-Window GUI begins, control switches to the first logical display after the defined displays (generally F7). While in this logical display, keystrokes are only directed to an active window that currently has the desktop focus. Put another way, you may or may not have keyboard control over the display contents. Use of the mouse or Alt-Tab key can move the desktop focus to the correct window so that keystrokes may be entered.

By convention, all X servers (the video drivers) accept keyboard control commands regardless of the desktop focus. The sequence Ctl-Alt-Backspace will terminate the X Server. Ctl-Alt-NumericPlus will step forward through the supported graphic modes while Ctl-Alt-NumericMinus steps backwards through the graphic modes. If only one graphic mode is defined for the display, the Plus or Minus control commands have no effect. This convention can be modified by editing the XF86Config file.

34. Contrast xdm, startx, and xinit.

The X display manager (xdm) presents a login graphic and the complete domain name of the machine. After a user logs into the machine, xdm launches startx to setup the user's GUI.

Startx is an easy-to-use shell script front-end that executes the older xinit program. Startx also reads a desktop preference file to see if a newer desktop manager such as afterstep or KDE is desired.

The xinit program reads the user's local shell script ".xinitrc" file or the system-wide default shell script file /usr/X11R6/lib/X11/xinit/xinitrc and starts the X-Window applications listed in the xinitrc file such as xclock, xload, and xterm. If a desktop has not already been launched, xinit also launches one of the two standby desktops Tom's window manager (twm) or Fred's virtual window manager (fvwm).

35. Describe the following /usr/X11R6/lib/X11/xinit/xinitrc file entries:

36. `#!/bin/sh`

37. `xterm -geometry 80x40+10+100 -fg black -bg white -name "virtual terminal" &`

38. `xterm -fn 7x13bold -geometry -20+10 -fg darkslategray -bg white -name "general"`

39. `&`

40. `xterm -geometry -20-30 -fn 9x15bold -fg white -bg black -name "mail" &`

41. `oclock -geometry 70x70+7+7 &`

42. `xsetroot -solid midnightblue &`

43. `exec fwm`

The shell script starts three xterm sessions (windows). The `-geometry` switch takes the form of "**columnsxrows+xpixels+ypixels** " where **columnsxrows** represents the number of character columns and rows. Pixels describe the distance from the display edge to the window edge beginning from the upper left corner of the screen. The "**xpixels**" are the absolute number of pixels from left to right and "**ypixels**" are the absolute number of pixels from top to bottom. If the x or y pixel delimiter is a "-" instead of a "+", it means start from the right hand side or start from the bottom of the display. Note that physical distance depends on the DPI and selected resolution of the monitor.

Thus, the first xterm window above has a size of 80 characters wide by 40 characters tall and it begins 10 pixels to the right of the display edge and 100 pixels down from the top. The second xterm has a default size of 80 characters wide by 25 characters tall and it begins 20 pixels from the right hand side and 10 pixels from the top of the display. The third xterm has the same default size and begins 20 pixels from the right hand side and 30 pixels from the bottom of the display. Two of the three xterms specify a unique font. Both are fixed-width and thick (bold), one is 7 pixels wide and 13 pixels high, while the other is 9 pixels wide by 15 pixels high. Finally, the three xterms each have different foreground and background colors specified as well as unique names in their title bars.

Next, a clock is specified for the desktop. It is 70 pixels by 70 pixels and placed seven pixels in from the upper left hand corner of the display, above the left hand side xterm.

Next, the root window (desktop) is switched from the default gray chain-link wallpaper to a color.

Last, but not least, the window manager is launched.

44. Explain the role of window managers.

Employing the general GUI algorithm described above, window managers present a visual abstraction of programs, activate and terminate programs, as well as switch focus among active programs. Programs can be activated from a desktop menu, a task bar, icons, or controls built into the program frame.

45. Describe the general design of X resources, application classes, and resource classes.

X resources provide abstractions that allow the manipulation of groups of desktop elements with single configuration statements. To create this second level of abstraction, another configuration file (`.Xdefaults`) specifies these classes and the `.xinitrc` shell scripts run the X resource database (`xrdb`) manager which loads the classes into the X server for subsequent use by the window manager.

Applications may be grouped into common classes such as virtual terminals, clocks, Web browsers, editors, etc. These classes are fixed by the programmer and specified in the manual page for that X client.

Resources are visual attributes such as foreground color or bitmap, background color or bitmap, geometry, font face, font kerning, etc. There will also be class-specific resources such as the log file associated with an `xterm` client.

Finally, to complicate things a bit more, resources themselves are arranged in hierarchical classes. This "grouping" feature allows one to specify many resource "instances" to be members of the same class.

46. Describe the following generic form of resource invoking:

47. (ApplicationClass|applicationName)*(ResourceClass|resourceName) : value

Consistent with many Unix applications, the vertical bar means logical OR. Thus, a "value" can be associated with a particular class or individual application. The asterisk (*) associates one or more visual attributes with the single program or class of programs.

48. Contrast the `xrdb` directives `xterm*background:purple` versus `Xterm*Background:green`.

The initial capital letter for class and resource specify whether the value (purple or green) is to apply to a single instance or the whole class. In the first example, just one `xterm` window background is made purple while the background for all `xterms` are made green in the second example.