

# **Linux Academy: User Interfaces and Desktops – Course Notes**

## **Video One – X and Startx**

- X
  - underlying system of graphical user interface displays on linux systems
  - the actual desktop GUI is driven by this application
    - can be started on the command line, but keeping it running is generally the responsibility of the display manager (lightdm, gdm, kdm, xdm, etc)
  - `:#` starts the GUI on the indicated display number (so it would be referred to in the DISPLAY environment variable as `DISPLAY=:#`)
  - is started by script 'startx'
- Startx
  - script in `/usr/bin` that starts up the Xwindows GUI interface
  - calls and uses X, xinit and xauth to set up the environment
  - options
    - `-depth`: start the server with the indicated color depth (overrides the `xorg.conf`)
    - `-dpi`: start the server with the indicated DPI
    - `-layout Multihead`: starts the server with multiple displays
- xinit
  - initializes the Xwindow system and often starts the first X client application (xterm is most commonly configured)
  - options
    - `:#` starts the GUI on the indicated display number
  - reads `/home/user/.xinitrc` and `/etc/xinitrc` for applications, variables and options to pass into the initialization routine

## **Video Two – xwininfo**

- provides window information for X applications
  - default run (no options) allows you to select a window to provide the statistics and information on (same as passing option `-stats`)
  - specific window ID or name can be passed at the command line
  - useful for inspecting the various size and event elements of any window on your X window desktop
  - options
    - `-root`: X windows desktop root is the target window
    - `-children`: all (root, parent and child) windows/components appearing on the desktop are displayed with their attributes
    - `-tree`: same as `-children` except that the parent/child relationship of all windows is maintained in a clearer tree view display
    - `-events`: displays just the interaction events that a particular window is listening for (onClick, onEnter, onExit, keyPress, etc)
    - `-size`: sizing hints (sizes in X/Y coordinates) displayed
    - `-wm`: window manager hints (i.e. specific interaction messages between the selected window and the underlying window manager)
    - `-shape`: window and border shape and size displayed
    - `-all`: verbose display with all options for selected window

## **Video Three – xdpinfo**

- display information utility for X
  - in contrast to 'xwininfo', which displayed window information, this utility displays information about the underlying X system
  - options
    - `-display`: the display name to connection to (`:0` or `:0.0` for example)

- -queryExtensions: show all installed X extensions
  - xdpinfo -queryExtensions | grep XINE would indicate whether the xineama extension is installed
- -ext extension-name: if you know a specific extension is installed (the aforementioned xineama for example), passing that name will display only the attributes for it

## Video Four – DISPLAY and DisplayManager

- DISPLAY
  - environment variable that is set and determines the display and screen that X applications are to display on
  - can be viewed:
    - set | grep DISPLAY
      - sample: DISPLAY=:1.0 (meaning localhost, display server #1, screen #0)
  - can also be used to redirect output to another host for X displays when used in conjunction with other utilities in this course
- DisplayManager
  - underlying X display applications that allow graphical system login and are responsible for starting the X window server and keeping it running/restarted
  - examples
    - Gnome – gdm
    - KDE – kdm
    - Generic X – xdm
    - Ubuntu Unity – lightdm
  - killing/stopping/restarting the display manager will affect the entire X system, even after the 'hand off' has taken place between the display manager and the X window system

## Video Five – xhost

- used to add and delete host names/IPs to the list allowed to make connections to the X window system
  - options
    - +[name]: add the name/IP to the allowed list
    - -[name]: remove the name/IP from the allowed list
    - +: allow all incoming client connections even if not on the list
    - -: disallow any incoming connection that is not already on the list
    - no parms: display the current status of connectivity for the server
  - usage:
    - remote server (where the X applications RUN):
      - xhost + (allows any connection)
      - export DISPLAY=[IP to Display To]:[DISPLAY#].[SCREEN#]
      - sample: export DISPLAY=192.168.1.200:0.0
    - any X application run on the server will display on the indicated IP
  - NOTE: this is an older method of remote X application display and is generally known to be insecure. It has largely been replaced by X tunneling over SSH (using the -Y parameter)

## Video Six – xorg.conf

- X configuration file that supplies configuration and run-time parameters for the entire X display system

- used to be required for all X windows setups
- better support for video cards by vendors (open source and proprietary) allow much of the configuration defaults to be obtained dynamically during system start up now
- remains useful in complex or highly customized environments
- Intel, ATI and Nvidia, open source or binary only drivers, all continue to work with the xorg.conf file
- generate a generic configuration file:
  - ATI: aticonfig
  - Nvidia: nvidia-config
  - Generic (and Intel): X -configure (or xorg -configure)
- File layout:
  - Section “Server Layout”
    - Identifier: the internal configuration name for the layout of monitors/desktop that can be used to include or refer to its options in other sections
    - Screen: identifies, by number, the screen that will be referred to in the DISPLAY environment variable, further names the screen with an identifier string that contains the layout options for all monitors as defined in the “Screen” section below
  - Section “Monitor”
    - Identifier: the internal configuration name for the monitor that can be used to include or refer to its options in other sections
    - Option: large number of options can be set (examples – VendorName, ModelName, DPMS, PreferredMode, TargetRefresh, Rotate)
  - Section “Device”
    - Identifier: the internal configuration name for the video device that can be used to include or refer to its options in other sections
    - Driver: the binary video driver name (nvidia, nouveau, radeon, intel, vesa, etc)
    - BusID: contains the bus type and ID of the video card (obtained by using the lspci utility and noting the BusID of the VGA card – example would be “PCI:2:0:0” to indicate PCI/x bus, device 2)
    - Option: large number of options can be set (examples – including the Monitor Identifier(s))
  - Section “Screen”
    - Identifier: the internal configuration name for the screen layout that can be used to include or refer to its options in other sections
    - Device: names the video card driver section as defined in the “Device” section above
    - DefaultDepth: the default color depth of the screen
    - SubSection “Display”
      - Viewport, Virtual Desktop, Depth, etc
- changes can sometimes be applied dynamically (most often with the proprietary binary driver utilities) but sometimes may yield strange results, best method of applying all changes to this file is by logging out, which forces the underlying display manager to rescan and read in the configuration changes

## Video Seven – X Font Server

- xfs
  - font server that contained and served the font configurations (DPI, size, rasterized, etc) to the underlying X window system for display

- applications that called fonts would cause a system 'getFontByName()' and 'getFontParms()' event to return the values needed to display the font in the system window
- font server listens on indicated TCP port for incoming connections and is referred to in the 'xorg.conf' file when used in older distributions
- options
  - -tcp: listen on the port indicated, no value means listen on port 4000
- available for backwards compatibility in Red Hat Enterprise Linux systems until v7, when it will no longer be supported at all
- superseded in modern distributions by the 'fontconfig' local font location and caching system
  - common font directories:
    - /usr/share/fonts
    - /usr/share/X11/fonts
    - /usr/share/cups/fonts
  - scanned periodically for changes, values cached internally
    - fc-cache: utility to scan and report on the font cache
    - options
      - -r: remove all caches
      - -f: rebuild all font caches by scanning default font system directories
    - fc-list: utility to list all fonts as installed on a system
    - options
      - no parms: list all
      - -v: verbose for all
      - familyname: list attributes for matching family
      - :lang=XX list installed fonts in that language (en, hi, ru, tz, etc)

## Video Eight – inittab

- RPM distributions (those that still use the SysV runlevel script directories in /etc/rc.d)
  - /etc/inittab
    - sets the default runlevel that the system boots into
      - sample: id:3:initdefault (which would boot into full multiuser but no GUI)
  - runlevels:
    - 0: Halt/Shutdown
    - 1: Single User/Repair (no services, no network)
    - 2: Multiuser, no networking
    - 3: Full multiuser, network, all services, no GUI
    - 4: Unused
    - 5: Same as runlevel 3 with X11/GUI
    - 6: Reboot
- Debian distributions (those that have moved to the upstart daemon for service and runlevel management)
  - /etc/init/rc-sysinit.conf
    - env DEFAULT\_RUNLEVEL=2 (default runlevel for Debian)
  - /etc/inittab can be created, but will only have the affect of booting to Debian defined runlevels
  - runlevels:
    - 0: Halt/Shutdown
    - 1: Single User/Repair (no services, no network)
    - 2: Full multiuser, networking all services, GUI
    - 3: Full multiuser, networking all services, GUI
    - 4: Full multiuser, networking all services, GUI

- 5: Full multiuser, networking all services, GUI
  - 6: Reboot
- all scripts and start up configurations are kept in the upstart directories (not `/etc/rc.d`) but rather `/etc/init`, `/etc/init.d/` and `/lib/upstart`