LPIC-1 TRAINING COURSE

Topic 106: User interfaces and desktops

Contents

Objectives

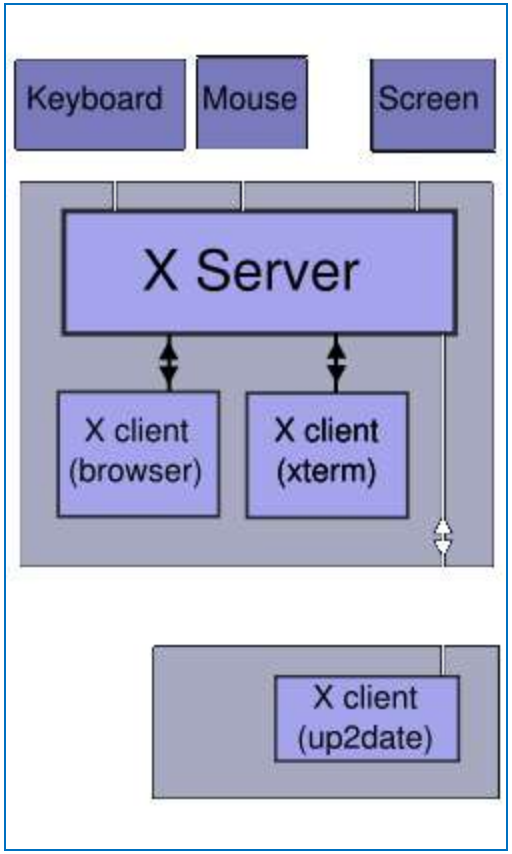
- Install and configure X11
- Setup and customize a display manager (XDM, GDM, KDM)
- Have a knowledge and awareness of accessibility technology

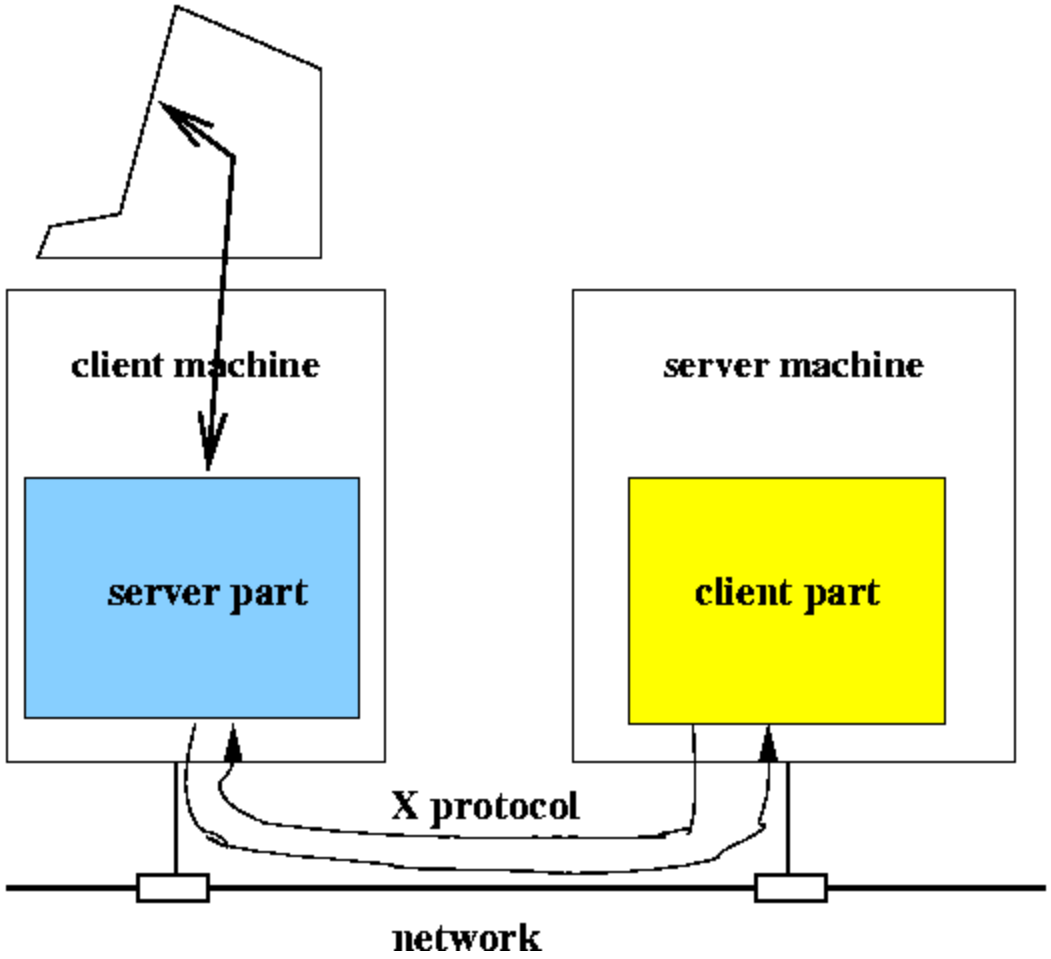
1. Overview of X Window System

What is X Window System

- A window system for graphical display
- Core component of Unix/Linux GUI
- Based on client-server model
  - ***X Server***: runs on a computer with a graphical display and keyboard
  - ***X Client***: sends requests for graphical output on *X server* and receives user input (keyboard, mouse) from *X server*
  - ***X Client*** and ***X Server*** may run on same machine or on different ones
    - ***X Client:*** **`DISPLAY`** variable contains server information
    - ***X Server:*** **`xhost`** controlling client connection
- X does not specific a user interface
  - ***Window Manager*** provides the look and feel

Components of X Window System





History of X Window System

- **1984**: *X1* (*X* version 1) originated at MIT
- **1987**: First release of *X* version 11 (*X11*)
- **1991**: XFree86 project develop *XFree86 1.1* based on *X11R4*
- **1992-2004**: *XFree86* is the dominant *X* window system in Linux
- **2004**: *XFree86 4.4* released under more restrictive and GPL-incompatible license
- **4/2004**: X.Org foundation release *X.Org-X11R6.7* based on *XFree86 4.4RC2*
- **Since 2004**: most Linux distros have adopted *X.Org-X11* in place of *XFree86*

Video hardware support

- Both **XFree86** and **X.Org** support a wide range of modern video cards
- Some manufacturers do not release open source drivers for all function
  - Often the case for accelerated 3D drivers
  - You may need to integrate a driver from the manuafacturer into your X system
- Live CD distros (Knoppix, Ubuntu…) can be used to test your hardware with X

X installation

- Most distros include a version of **XFree86** or **X.Org** already packaged for the system
- **XFree86** installation:
  - Download prebuilt packages or source files from www.XFree86.org
  - Should backup your ***/usr/X11R6***, ***/etc/X11*** and ***/etc/fonts*** directories before installation
- **X.Org** installation:
  - Download prebuilt packages or source files from www.X.org

XFree86 configuration tools

- **`XFree86 -configure`**
  automatically query hardware and produce a configuration file ***/root/XF86Config.new*** for later modification
- **`xf86cfg`**
  help you tweak settings once X is at least partially running
- **`redhat-config-xfree86`**
  available on some RedHat system using XFree86

X.Org configuration tools

- **`X -configure`** or **`Xorg -configure`** automatically query hardware and produce a configuration file ***/root/xorg.conf.new*** for later modification
- **`xorgcfg`**
  similar to **`xf86cfg`**
- **`system-config-display`**
  included in Red Hat, CentOS and Fedora

X configuration file

- All X configuration tools gather the same type of information needed to manually configure X
- You should understand the underlying configuration file to understand these tools
- Location of X configuration file

X.Org	XFree86
/etc/xorg.conf or ***/etc/X11/xorg.conf*** or ***/usr/X11R6/etc/xorg.conf***	***/etc/XF86Config***(*) or ***/etc/X11/XF86Config***(*) or ***/usr/X11R6/etc/XF86Config***(*)

(*) can be ***XF86Config-4*** for XFree86 version 4

Layout of the X configuration file

- Configuration file are broken down into multi-line section
- Sections:

Section	Description
Files	Location of fonts
ServerFlags	Server flags
Module	Dynamic module loading
InputDevice	Input device description
Device	Graphic device description
VideoAdapter	Xvideo adapter description
Monitor	Monitor description
Modes	Video modes description
Screen	Screen configuration
ServerLayout	Overall layout

Example of X configuration file

```
Section “Monitor”
    Identifier “Monitor0”
    ModelName  “VisionMaster Pro 450”
    HorizSync  27.0-115.0
    VertRefresh 50.0-160.0
    Modeline “1360x1024” 197.8 \
                  1360 1370 1480 1752 \
                  1024 1031 1046 1072 -HSync -VSync
EndSection

Section “Device”
    Identifier  “Videocard0”
    Driver      “nv”
    VendorName  “nVidia”
    BoardName   “GeForce 6100”
    VideoRam    131072
EndSection
```

```
Section “Screen”
    Identifier “Screen0”
    Device     “Videocard0”
    Monitor    “Monitor0”
    DefaultDepth  24
    SubSection “Display”
       Depth     24
       Modes    “1024x768” “1024x600” “800x600”
     EndSubSection
    SubSection “Display”
       Depth     8
       Modes    “1024x768” “800x600” “640x480”
    EndSubSection
EndSection
```

The X Configure-and-Test Cycle

- Shutdown the X session or kill the X server
  - (Red Hat, Fedora) **`telinit 3`**
  - (Debian, Ubuntu, Gentoo) **`/etc/init.d/xdm stop`** (or **`gdm`**, **`kdm`**)
  - **`Ctrl + Alt + Backspace`**
- Log in using a text-mode and tweak your X settings
  - Using text-based X configuration program
  - Modifying X configuration file
- Start the X server to see the result
  - **`startx`**
- If you get the desired results, restart the X session
  - (Red Hat, Fedora) **`telinit 5`**
  - (Debian, Ubuntu, Gentoo) **`/etc/init.d/xdm start`** (or **`gdm`**, **`kdm`**)

Obtaining and Tuning X Display

- **`xdpyinfo`**
  display detailed technical information about the current display
- **`xwininfo`**
  display detailed technical information about a specific window
- **`xvidtune`**
  tune the size and position of your X display, you can test, apply and view the settings information to modify X configuration file

Display Managers

- Present a graphical login screen, handle authentication, start a graphical session
- Implemented as a network login protocol (XDMCP – X Display Manager Control Protocol)
- *X Display Manager* (***XDM***), *KDE Display Manager* (***KDM***) and *GNOME Display Manager* (***GDM***) are common on Linux

Display Manager (cont')

- On Red Hat & SUSE, X is usually started in runlevel 5
  - Determiniation of default runlevel is made in ***/etc/inittab***.

```
# The default runlevel is defined here
id:5:initdefault:
```

  - Display Manager is started by an entry in ***/etc/inittab*** or by runlevel startup script
- Stopping a Display Manager
  - Exampe: **`/usr/sbin/gdm-stop`**
- Starting a Display Manager
  - Example: **`/usr/sbin/gdm start`**

Configuring XDM

- XDM is included in the XFree86 and X.Org packages
- Main configuration file: ***/etc/X11/xdm/xdm-config***
- Access control file: ***/etc/X11/xdm/Xaccess***
- X-based program control file: ***/etc/X11/xdm/Xresources***
- You can customize the way XDM works by updating the scripts in ***/etc/X11/xdm***
  - Eg: ***Xsetup*** script lets you customize the greeting
- XDM uses the default resolution from ***XF86Config*** or ***xorg.conf*** file

Configuring KDM

- ***KDM*** is the Desktop Manager for the KDE
- ***KDM*** is based partly on ***XDM*** and so shares many of its configuration options
- Location of the KDE configuration files is unpredictable
  - ***$KDEDIR/share/config/kdm*** or ***/etc/X11/kdm*** or ***/etc/kde/kdm***
- Main configuration file is ***kdmrc***
  - **`[Xdmcp]`** section provides options relating to network operation. **`Enable=true`** will enable network logins
- You can also configure KDM by using KDE control center (**`kcontrol`**)

Configuring GDM

- ***GDM*** is the Desktop Manager for GNOME
- ***GDM*** was not based on XDM but written from scratch
- Configuration file (***gdm.conf***) normally located in ***/etc/X11/gdm***
  - Setting **`enable=yes`** in **`[xdmcp]`** section will enable remote login
- You can also configure GDM by using **`gdmsetup`**

X Core Fonts and Fonts server

- **X core fonts** are fonts that are handled directly by X
- Fonts are listed in ***XF86Config*** (or ***xorg.conf***) by the keyword **`FontPath`** in the **`Files`** section
  - **`FontPath "/usr/X11R6/lib/X11/fonts/TrueType"`**
  - **`FontPath "/usr/X11R6/lib/fonts/75dpi:unscaled"`**
- Font servers can also be listed but MUST be first in the list
  - **`FontPath "unix/:7100“`**
  - **`FontPath "tcp/myserver.fd.com:7100“`**

Setting-up X Core Font directory

1. Create a new font directory
   - Eg: **`mkdir/opt/fonts`**
2. Copy your new fonts to this directory
3. Run **`mkfontscale`** and **`mkfontdir`** to create summary files ***fonts.scale*** and ***fonts.dir*** that describes the fonts
   - Eg: **`mkfontscale && mkfontdir`**
4. Adding fonts to X's FontPath:
   - Temporarily add with **`xset`**:
     - Eg:
       ```
       xset +fp /opt/fonts
       xset fp rehash
       ```
   - Permanently add: Add **`FontPath`** to ***XF86Config*** (or ***xorg.conf***) file

Setting-up X Core Font server (xfs)

- ***xfs*** is the standard Font Server which listens on port 7100
- ***xfs*** configuration file: ***/etc/X11/fs/config*** or ***/etc/X11/xfs.conf***
  - Example of ***xfs*** configuration file
- Steps to setup a X Font server:
  1. Install the fonts on the system (see previous slide)
  2. Add new font directory to the **`catalogue`** section in the xfs configuration file
  3. Restart the font server: **`/etc/init.d/xfs restart`**
  4. Restart X or run **`xfs fp rehash`** on client

X core fonts vs. Xft fonts

- **X core fonts** have several drawbacks
  - Server based (not local to applications)
  - Limited support for advanced typographic features
  - Don’t support font smoothing (anti-aliasing)
- New **Xft fonts** system is client-based
  - Application access font files on the computer they’re running
  - Support font smoothing and other advanced font features
- **X core fonts** and **Xft fonts** can share the same font directory
- Most modern GUI Linux programs are Xft-enabled

Configuring Xft Fonts

- Simply copying your font files to Xft default
  - ***/usr/X11R6/lib/X11/lib/fonts/***
  - ***~/.fonts/***
  - ***/usr/local/share/fonts/***
- Can edit Xft configuration file (***/etc/fonts/local.conf***) to add new font directory or setting advanced features.
  - run **`fc-cache`** as root for ***Xft*** to reload fonts

Exercise 1

In this exercise, you will get familiar with X server from your local system

1. Change your linux system to runlevel 3
   - Hint: **`init 3`**
2. Switch to the **tty1** (Ctlr+Alt+F1 or **`chvt 1`**) and login with **root** account
3. View your X11 configuration file
   - Hint: **`less /etc/X11/xorg.conf`**
4. Start X server
   - Hint: **`X &`**
5. Switch to the **tty2** (Ctlr+Alt+F2 or **`chvt 2`**), login and reate the environment variable: **`DISPLAY=localhost:0`**
   - Hint: **`export DISPLAY=localhost:0`**
6. Run **`xterm`** then switch to **tty7** (Ctlr+Alt+F7) where X server is running to see **xterm**
   - Hint: **`xterm &`**
7. Run **`twm`** (Tab Window Manager) from the **xterm** shell and see what happens
   - Hint: **`twm &`**

Exercise 1 (cont')

8. Switch to **tty1** (Ctlr+Alt+F1) and start a new X server in display 1
   - Hint: **`X :1 &`**
9. Switch to **tty3** (Ctlr+Alt+F3 or **`chvt 3`**), login and and create a environment variable: **`DISPLAY=:1`**
   - Hint: **`export DISPLAY=:1`**
10. Run **`xterm`** then switch to **tty8** (Ctlr+Alt+F8) where the second X server is running to see **xterm**
    - Hint: **`xterm &`**
11. Run **`gnome-session`** from this **xterm** shell and see what happens
    - Hint: **`gnome-session`**

Exercise 2

In this exercise, you will install **cygwin** *or other* **X Window Server** *software on your MS Windows computer, then run programs from a Linux computer but displays and accepts input from the MS Windows computer. You also need* **putty** *to remotely connect to Linux computer via ssh.*

1. Download or install **cygwin** or other **X Window Server** software on your MS Windows system, then start X Server
   - Example with **cygwin**: **`startx`**
2. Disable X Server access control on your Windows system
   - Example with **cygwin**: **`xhost +`**
3. Open a **ssh** or **telnet** session to your Linux system with **putty**.
4. Whithin the ssh session, export the DISPLAY variable with the appropriate value
   - Example: **`export DISPLAY=<Windows_system_IPAddress>:0`**
5. Whithin the ssh session, run a GUI application (Eg: **firefox**, **ooffice**, **xterm**, **gnome-session**…) and see the result.

2. Accessibility

Keyboard Accessibility Settings

- On GNOME: Access these settings by seleting ***System > Preferences > Hardware > Mouse and System > Preferences > Hardware > Keyboard***
  - Keyboard repeat rate
  - Sticky key
  - Slow keys
  - Bounce/debounce key
  - Mouse tracking and click options
  - Simulate mouse clicks
  - Mouse emulation/mouse navigation
  - Mouse gesture
- GNOME Onscreen keyboard: ***Access via Applications > Accessibility > On-Screen Keyboard***

Screen Display Settings

- Including font options, contrast settings and screen magnification tools
  - Adjusting default fonts and contrast: ***System > Preference > Appearance***
  - Magnifier Tools: Install ***KMag*** (part of KDE suite)

Additional Assistive Technologies

- Configuring Linux to Speak:
  - Using ***Orca*** (*http://live.gnome.org/Orca*)
  - Using ***Emacspeak*** (*http://emacspeak.sourceforge.net*)
- Configuring Braille displays: Using ***BRLTTY*** utilities (*http://www.mielke.cc/brltty/*)
  - The 2.6.26 Linux kernel adds some direct support for Braille displays

Thank You !

BACKUP SLIDES

Example of *xfs* configuration file

```
# allow a max of 10 clients to connect to this font server
client-limit = 10

# when a font server reaches its limit, start up a new one
clone-self = on

# alternate font servers for clients to use
#alternate-servers = foo:7101,bar:7102

# where to look for fonts
#
catalogue = /usr/X11R6/lib/X11/fonts/misc:unscaled,
          /usr/X11R6/lib/X11/fonts/75dpi:unscaled,
          /usr/X11R6/lib/X11/fonts/100dpi:unscaled,
          /usr/X11R6/lib/X11/fonts/misc,
          /usr/X11R6/lib/X11/fonts/Type1,
          /usr/X11R6/lib/X11/fonts/Speedo,
          /usr/X11R6/lib/X11/fonts/cyrillic,
          /usr/X11R6/lib/X11/fonts/TTF,
          /usr/share/fonts/default/Type1

# in 12 points, decipoints
default-point-size = 120

# 100 x 100 and 75 x 75
default-resolutions = 75,75,100,100

# how to log errors
use-syslog = on

# don't listen to TCP ports by default for security reasons
no-listen = tcp
```