

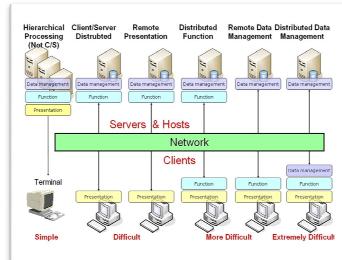
# LINUX DESKTOP ENVIRONMENT

## Objectives

- Understand the GUI and applications in relation to Linux operating system

1 •

## Clients in Client/Server Environment



### Why Windows Workstations?

- Easy to use
- Widely used
- Lots of software



### Why Linux Servers?

- Stable
- Secure
- Low cost

### The X Windows System

- Windows Manager
- Desktop Manager



### Other Interoperability Approaches

- Wine
- Lindows
- VMware

2 •



LINUXDISTRIBUTION.JPG



GLDT76.PNG

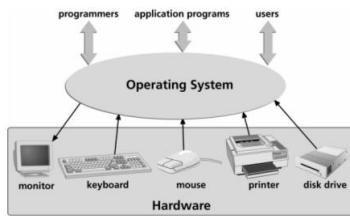


GNULINUXUPDATEDW4.PNG

## LINUX DISTRIBUTIONS

- Just as all the components of a car are bundled together to make a complete car, all components are bundled together in a wide variety of Linux distribution packages, commonly referred to "distros". There are three basic packaging groups:
  - Desktop
    - traditional desktop installation with graphical interface and common applications. Great for home use, on a laptop or as a workstation.
  - Server/Enterprise
    - used in a production environment and business, a server/enterprise edition can still be used on a home server or project.
  - Live CD
    - Computer booted without using the hard drive and without installing into permanent memory, runs directly from the storing device—USB stick, CD, DVD.
- Each group has a different default configuration, including which runlevel is set. The runlevel determines the run condition of the system (config file: /etc/inittab)
  - 1 = Single User Mode
  - 2 = Multiuser, without Network File System
  - 3 = Full Multiuser (command line)
  - 5 = X11 (GUI)
- Before you download and install a linux operating system, read about the various distributions on the internet.
- The images, LinuxDistributions.jpg, gldt76.png, gnulinuxupdated4.png, contain some of the popular distros and the relationship among them.
- Once you decide on which distribution you wish to download, read the document: easiest\_linux\_guide\_ever.pdf.

3 •



## Operating System Components

### What is a kernel?

Kernel is the heart of the operating system where the real work is done. It decides how much RAM space is to be allocated to a program before it is loaded and executed. It ensures that only one file is printed on the printer at a time. It prevents an existing file on the disk from being accidentally overwritten by another file. It guarantees that when execution of a program given to the CPU for processing has been completed, the program relinquishes the CPU so that other programs can be executed. In other words, it manages the resources, which include all hardware components, including keyboard, display, memory, storage, modem, CPU, etc.

### What is a shell?

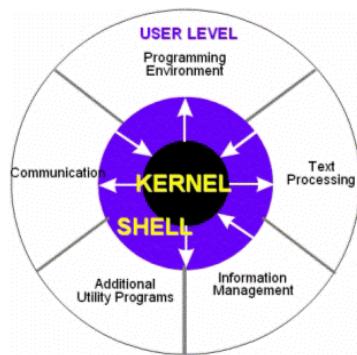
Shell forms the application user's interface that protects the kernel from users. It provides a set of functions that can be used by the user and applications to access the kernel's services. A SHELL IS A LANGUAGE. Most users know the "click-and-point language of the desktop". BUT in that language the user is treated as a baby, asked to pick from what is presented to her. A shell, on the other hand, is an advanced way of communicating with the system, because it allows for conversation and taking initiative. Both partners are equal, so new ideas can be tested. The shell allows the user to handle a system in a flexible way. Don't you want to be treated as an adult?

### Interacting with an operating system

Operating system is a program that allows you to interact with the computer—all of the software and hardware on your computer

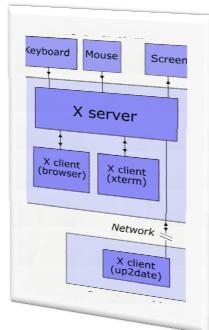
**With a command-line interface (CLI),** you type a text command and the computer responds according to that command.

**With a graphical user interface (GUI),** you interact with the computer through a graphical interface with pictures and buttons by using the mouse and keyboard.



4 •

**GUI is a set of programs, it is NOT part of the operating system, that provides an alternate way of communication between the kernel and users.**

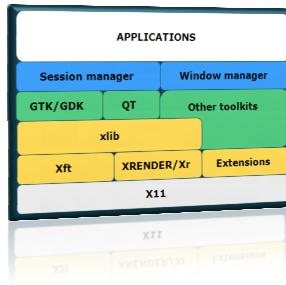


## X Window System

- provides the only **common networked windowing environment** bridging the heterogeneous platforms in today's computing
- one of the most successful **open-source, collaborative** technologies developed to date and is the standard graphical window system for the Linux® and UNIX® operating systems
- Every Unix system supports a Graphical User Interface (GUI).
- Like the ubiquitous Microsoft Windows, every application is
  - displayed in a separate **window**, uses the **mouse** to invoke programs, display **menus, select options**, and handle **cut-copy-paste operations**.
- Because X uses a **bit-mapped display** (*where every pixel on the screen is handled individually*), Web browsers like Netscape, Mozilla, and Konqueror must run under X.
- The inherent independence of X from the operating system, the network and the hardware and its successful interoperability, have made it widely available and deployed with more than 30 million users worldwide
- All major hardware vendors support X Window System

Many third parties provide technologies for integrating X Window System applications into the networked computer or personal computer environments including Microsoft Windows®, UNIX, Linux and Mac OS® X.

5 •



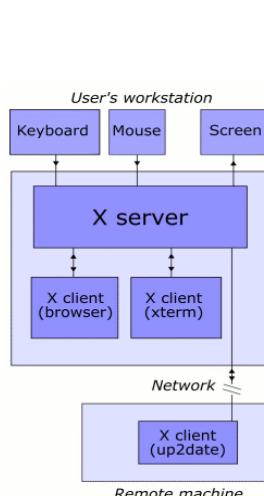
## X Design Principles

- Scheifler and Gettys set out the early principles of X as follows (Scheifler-Gettys 1996):
  1. Do not add new functionality unless an implementer cannot complete a real application without it.
  2. It is as important to decide what a system **is not** as to decide what it **is**. Do not serve all the world's needs; rather, make the system extensible so that additional needs can be met in an upwardly compatible fashion.
  3. The only thing worse than generalizing from one example is generalizing from no examples at all.
  4. If a problem is not completely understood, it is probably best to provide no solution at all.
  5. If you can get 90 percent of the desired effect for 10 percent of the work, use the simpler solution.
  6. Isolate complexity as much as possible.
  7. Provide mechanism rather than policy. In particular, place user interface policy in the clients' hands.

6 •

- X provides the **basic framework** for a **graphical user interface (GUI)** environment:
  - drawing and moving windows on the screen and interacting with a mouse and keyboard.
- X does not mandate the user interface—this is handled by individual programs.
  - As such, the visual styling of X-based environments varies greatly; different programs may present radically different interfaces.
- X features network transparency:
  - the machine where application programs (the client applications) run need not be the user's local machine (the display server).
  - X's usage of the terms "client" and "server" is the reverse of what people often expect—the "server" is the user's local display rather than the remote machine.
- Developed at MIT in 1984 to provide a unified windowing system for high-end scientific workstations (expensive number-crunching machines)
- Current protocol version, X11, was released in September 1987.
- Project led by the X.org Foundation
- Latest official release: X11R6.8.2 is free software under the MIT License and similar permissive licenses.
- Ported by enthusiasts in 1990's to run on inexpensive home PCs, this project became known as XFree86
- XFree86's descendants distributed with Linux today

7 •

**X server**

- Allows graphical output of a program to be displayed on remote machine
- Runs on the user's local machine
- Performs the low-level operations of drawing the graphics on screen
- Talks directly with graphics card, must be configured with appropriate resolution, refresh rate, color depth, and so on.
- Configuration file: XFree86Config
- Linux distributions auto-detect the correct settings, minimizing the need to change config
- Listens for user input via mouse and keyboard and relays the clicks to X client (events).
- Applications may send output to any device running X server

**X client**

- Any program that uses the X window system as a GUI, (Examples: xterm, xcalc, Abiword,...)
- Typically waits for user events sent by the X server and responds by sending redraw messages back to the server

**X protocol**

- X client and X server communicate using X protocol, which enables the client and server to be separated over a network

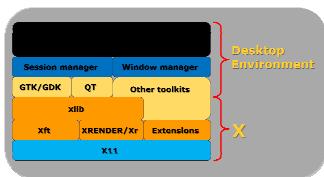
**Xlib**

- The library used indirectly by X client to generate the X Protocol messages.
- Provides a low-level API to allow the client to draw very basic elements on the X server and to respond to the simplest of inputs

8 •

## X and Desktop Environment

- All GUI libraries in Linux sit on top of the underlying windowing system called the X Window System (X11 or just X).



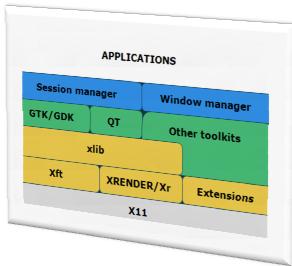
## The Window Manager

- X is no good without a window manager running on top of it. Window managers operate between X and the graphical programs you want to run. They do stuff like draw the borders around your programs, manage their placement, draw title bars, etc. They are responsible for positioning windows on screen. They support separate "workspaces" that divide the desktop area. Also responsible for adding decoration around each window, consisting of a frame, title bar with maximize, minimize and close icons.
- Some WMs are very simple, some are very complex.
- The window manager is a rather special X client. It is a tool which allows you to move, resize, iconify and kill other windows. It provides a title bar for every window, and may make several pop-up menus available.
- A number of window managers are available which one you use is a matter of personal preference.

9 •

## Available Window managers

- twm** — The Tab Window Manager
  - This is the most commonly used window manager at Strathclyde, and is probably the best one to use initially.
- vtwm** — A Virtual twm
  - an extended version of twm which allows for a "virtual desktop". This means that the apparent size of the X servers "screen" can be several times larger than the physical dimensions of the workstation screen. The workstation can be regarded as being a "window" onto the virtual desktop, and this window may be moved over the virtual desktop by using the mouse or the cursor keys. It's a useful facility to have available, but isn't really recommended for the beginner.
- tvtwm** — Another virtual twm, and a rival to vtwm.
  - Again, not for the novice.
- olwm** — Open Look Window Manager
  - provides an interface of a similar style to that available under Sun's OpenWindows environment.
- mwm** — Motif Window Manager;
  - provides a Motif-style interface.



- There are many desktop managers, the most common are:
  - KDE
  - Gnome, Sawfish
  - Fluxbox, Blackbox
  - Enlightenment
  - Window Maker
  - XD640 desktop environment

## The Desktop Manager

### • Fluxbox/Blackbox

- This WM is for more experienced users - those that are annoyed by DEs.
- Fluxbox and the WM it was based off of, Blackbox, are minimalistic to some degree. This means that it only does what is needed. Nothing more. There is the toolbar, which only includes iconified (minimized) programs, the time, and the workspace name. There is also the slit, where you can put dockapps - small graphical programs that monitor time, CPU, Memory, I/O, Network, etc. Most people that use these WMs rely on keybindings to open programs - in addition to a small menu. A major selling point to these window managers is that they're very fast and very stable.
- Fluxbox, unlike Blackbox, also heavily uses tabs - so that you can group windows together.

### • Enlightenment

- is a popular window manager, with an almost cult-like following. It is the inspiration for many current window managers. The window manager however hasn't seen a new release in nearly two years, and is dying out. It is kind of like Blackbox, but more graphics heavy (prettier) with cool effects.

### • Window Maker

- offers a lightweight alternative to GNOME and KDE. It provides a look-and-feel that mimics the NEXTSTEP user interface, but can be themed any way you like. Features such as the Dock and Clip make Window Maker a popular choice among system administrators.

11 •



### • XD640 desktop environment

- XD640 is a simple graphical desktop environment for X-Window. The goal is to create a set of light weight and fast applications running on a slow computer (i486 66MHz 16MB) and on a small screen (640x480) and released under the BSD license.

## KDE/Qt

- Qt is a mature, cross-platform GUI toolkit written in C++ and has native support in Linux, Unix, Windows, MacOS X, and embedded platforms
- KDE or K Desktop Environment is an open source desktop environment based on the Qt GUI library.
- A host of applications and utilities are part of KDE, including a complete office suite, a Web Browser, and a fully featured IDE for programming KDE/Qt applications
- Qt Example:



### Calculator Class Definition

```

class Calculator : public QDialog
{
    Q_OBJECT

public:
    Calculator(QWidget *parent = 0);

private slots:
    void digitClicked();
    void unaryOperatorClicked();
    void additiveOperatorClicked();
    void multiplicativeOperatorClicked();
    void equalClicked();
    void pointClicked();
}
  
```



12 •



## Gnome/GTK+

- GTK+ is a library that greatly simplifies the creation of Graphical User Interfaces by providing a set of ready-made components called widgets that we bolt together with easy-to-use function calls to application logic.
- Gnome is the name given to a project started in 1997 to create a unified desktop for Linux
  - originally stood for GNU Network Object Model Environment, but refers to the complete desktop environment today
  - consists of a panel for launching applications, a suite of programs and utilities, programming libraries, and developer support features.



<http://www.gtk.org/>

more readable and to make changes easier). The data should be of the same data type as the model column.  
Here is an example where we create a store that stores two strings and one integer for each row:

```
enum
{
    COL_FIRST_NAME = 0,
    COL_LAST_NAME,
    COL_YEAR_BORN,
    NUM_COLS
};

GtkListStore *liststore;
GtkTreeIter iter;

liststore = gtk_list_store_new(NUM_COLS, G_TYPE_STRING, G_TYPE_STRING, G_TYPE_UINT);

/* Append an empty row to the list store. Iter will point to the new row */
gtk_list_store_append(liststore, &iter);

/* Fill fields with some data */
gtk_list_store_set (liststore, &iter,
    COL_FIRST_NAME, "Joe",
    COL_LAST_NAME, "Average",
    COL_YEAR_BORN, (quint) 1970,
    -1);
```

You do not need to worry about allocating and freeing memory for the data to store. The model (or more precisely: the GLib/GObject GType and GValue system) will take care of that for you. If you store a string, for example, the

13 •

## Applications

### Running Office Applications

- OpenOffice suite contains a set of office tools similar to MS Office
  - Create documents, spreadsheets, presentations, charts and graphs, and databases
  - File filters that allow it to work with standard MS office documents
- Evolution is an email client application
  - Send and read email messages over an ISP
  - Personal manager that stores contacts, appointments, reminders, and memos

### Media Applications

- Browse the web
  - Mozilla Firefox web browser is open source, free browser available for various platform (developed from original Netscape)
- Message online
  - Pidgin, cross-platform instant messaging client application, allows to log in to multiple IM accounts simultaneously (AIM, ICQ, MSN, IRC, and many more)
- Phone over internet
  - Ekiga Softphone application employs SIP (Session Initiation Protocol) to allow free verbal communication via internet connection

14 •

- Watch videos
  - Movie Player applications usually do not ship with codecs to play proprietary formats—MPG, WMV, etc. offers install-on-demand
- Play music
  - Rhythmbox music player plays music files, podcast aggregator, and internet radio tuner
- Burn disks
  - Sound Juicer saves ripped audio files in open source OGG format, can be played by Rhythmbox
  - Serpentine burns ripped OGG files to a blank CD as audio files creating a compilation CD
- Create graphics
  - GIMP (GNU Image Manipulation Program)
    - Primary image editing application
    - Open-source
    - Includes over 220 plugins in a standard install
    - Many capabilities of Adobe Photoshop, except native support for CMYK colors
    - Highly configurable, powerful scripting support
- Adding more applications
  - Can be installed with an internet connection
  - Linux distros usually provides Add/Remove Applications dialog

15 •

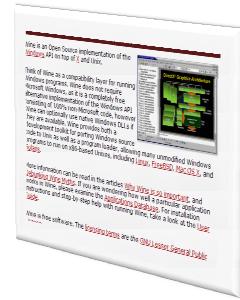
**WINE**

- Implementation of Windows Win32 and Win16 APIs on top of X and Unix.
  - Windows compatibility layer.
  - Provides both
    - a development toolkit (Winelib) for porting Windows sources to Unix
    - a program loader, allowing many unmodified Windows binaries to run under Intel Unix.
  - Works on most popular Intel Unix and Linux systems.
- Why use wine?
- Desire to use Windows applications
    - without a Windows OS
    - without exiting Linux
  - Wine makes it possible to take advantage of all the Unix strong points (stability, flexibility, remote administration) while using Windows applications

Wine is open-source so you can extend it to suit your needs

<http://linux.softpedia.com/get/System/Emulators/Wine-148.shtml>

<http://www.winehq.com>



16 •



<http://www.lindows.com>



### Lindows, Linspire

- Similar to WINE, but a complete OS.
- Based on Debian Linux
- Allows running of Windows software on Linux.
- "Broadband based" Operating System
- Applications are downloaded from the web
  - "Click & Run" technology

### Virtual Machines

- work by creating fully isolated, secure virtual machine that encapsulates an operating system and its applications.
- Maps the physical hardware resources to the virtual machine's resources, each VM has its own CPU, memory, disks, and I/O devices, and is the full equivalent of a standard x86 machine
- Installs onto a host operating system and provides broad hardware support by inheriting device support from the host
- Vmware <http://www.vmware.com>
- Virtual Box <http://www.virtualbox.org>