

Introduction to Linux

- ▶ Why Linux?
 - ▶ Nothing is hidden (opensource is good for learning)
 - ▶ It runs anywhere, on almost anything
Android, OS X, Raspberry Pi, Old PCs
 - ▶ Low resource requirements
 - ▶ You are likely to use it in your job; i.e.; good on resume
 - ▶ Free: as in beer and in freedom
 - ▶ Software development toolchain is free and high quality
 - ▶ Lots of software available
 - ▶ It conforms to your needs
- ▶ Why not?
 - ▶ Occasional hardware support issues
 - ▶ Won't run most Microsoft programs, many games
 - ▶ If you're not paying attention, you can really screw up

Introduction to Linux

- ▶ Linux is in more places than you think...
 - ▶ U.S. Army is the largest install base for Red Hat Linux.
 - ▶ The US Navy nuclear submarine fleet runs Linux.
 - ▶ In 2006, the FAA migrated to Red Hat Linux to manage air traffic.
 - ▶ Over 90% of all supercomputers and IBM's Watson run Linux.
 - ▶ Google's search clusters and other apps run on Linux.
 - ▶ Almost everything that happens in Amazon's nine worldwide distribution centers is driven by Linux.
 - ▶ Wikipedia uses Linux to serve up its web pages.
 - ▶ The New York Stock Exchange uses Linux for its trading platform.
 - ▶ Linux powers the \$10 billion Large Hadron Collider. CERN also runs Linux on its 20,000 internal servers.
 - ▶ The Apache HTTP Server is the dominant web server today.

Source: <http://www.comparebusinessproducts.com>

Introduction to Linux

Unix, Linux, What's the difference?

▶ UNIX

- ▶ Ken Thompson and Dennis Ritchie, AT&T Bell Labs 1971
- ▶ Dennis Ritchie developed "C", to build Unix
- ▶ Designed to be shared and portable
- ▶ Became quickly popular in the academic community
- ▶ Unix philosophy: "The power of a system comes more from the relationships among programs than from the programs themselves." -Brian Kernighan and Rob Pike

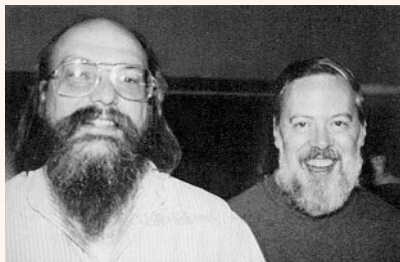


Figure: Ken Thompson and Dennis Ritchie - The original neckbeards!

Introduction to Linux

Unix, Linux, What's the difference?

- ▶ Linux

- ▶ Richard Stallman's goal was a free UNIX-like OS circa 1983
- ▶ Written in "C" as a free alternative to UNIX
- ▶ Kernel written originally/maintained by Linus Torvalds 1991
- ▶ Linux is more accurately called GNU/Linux



(a) Linus Torvalds



(b) Richard Stallman

Bottom Line: Users can hardly tell the difference

Introduction to Linux

What can I do with it?

- ▶ Do program development or uC's, Android, cross platform apps
- ▶ Start a business with nearly no software tool cost
- ▶ Bring old computers back to a useful life
- ▶ Customize your work environment the way you want it
- ▶ Customize the OS the way you want it

What can I avoid doing with it?

- ▶ Buying software
- ▶ Updating software packages separately
- ▶ Hunting for drivers
- ▶ Waiting for a big company to fix bugs
- ▶ Being on hold for an hour to get help

Introduction to Linux

The Linux Shell

- ▶ The shell is your direct portal into the operating system
- ▶ Shell is a command interpreter
- ▶ Linux + Shell programming = Software Lego Blocks
- ▶ Traditional user interface is the command line
- ▶ Shell has built-in command language
- ▶ Typically we use the "Bash" shell

Upside/Downside

- ▶ High level of control
- ▶ Considerable complexity
- ▶ High level of productivity possible
- ▶ GUI easier but shell is far more efficient (with time)

Introduction to Linux

Linux General Philosophy

- ▶ *Unix was not designed to stop its users from doing stupid things, as that would also stop them from doing clever things.* - Doug Gwyn
- ▶ *With great power, comes great responsibility* - Spider-Man, Winston Churchill...
- ▶ Linux allows you to do almost anything you tell it to, good or bad.
- ▶ Win/Mac: Keep users away from everything that might get them into trouble

Introduction to Linux

Linux General Philosophy

- ▶ The user should know better.....so he must specify how things work
- ▶ Provide mechanisms, not policy
 - ▶ Mechanism: long life time
 - ▶ Policy: short life time
- ▶ Its not friendly, but its efficient
 - ▶ Don't confuse ease of use with efficiency
 - ▶ Efficiency will pay dividends in time
 - ▶ Pedestrian OS'es achieve glossiness by locking users into one interface policy. Its narrow, rigid and works well for a fixed set of jobs. But, unanticipated tasks are often impossible of very painful.
- ▶ Easy things are easy, hard things are possible

Introduction to Linux

Linux General Philosophy

- ▶ Linux provides a large set of simple, *sharp* tools connected with with well specified interfaces which are usually data streams of plain text.
- ▶ No one big tool is smart enough to handle all cases or optimized for everything or can anticipate all the uses to which it may be put.
- ▶ Linux is like a big tool box, or a lumberyard full of lumber
- ▶ We get a big say in what gets built and how its structured
- ▶ Its a plus for us. We're engineers. We build stuff.

Introduction to Linux

GUI:

- ▶ What you see is what you get
- ▶ What you see is all you get
- ▶ Easy but not necessarily efficient

CLI:

- ▶ Not always the best way: CAD Tools, etc.
- ▶ For power users, not novices

A CLI allows you to do things for which a program does not exist.

Introduction to Linux

Accessing a Linux Machine at OSU

- ▶ Windows
 - ▶ The program PuTTY provides a window with a shell interface
 - ▶ Fill in the boxes, point and click
- ▶ Mac
 - ▶ Using built-in Unix shell, execute "ssh" command
 - ▶ `$ ssh -l username access.engr.orst.edu`
- ▶ Virtualize
 - ▶ VMWare - virtualization software hosts a guest operating system
 - ▶ VMWare is free from COE computing website
- ▶ Which distribution (distro)
 - ▶ Ubuntu - popular, but has Windows 8-like interface
 - ▶ Mint - built on Ubuntu but with better windowing interface
 - ▶ Fedora - beta version of Red Hat
 - ▶ Red Hat Enterprise Version - This is what COE hosts

Introduction to Linux

Linux General Philosophy

- ▶ Button pushers are easily replaced, craft your own tools
- ▶ We are being groomed to be consumers
- ▶ Resist and be creators of new things

