Introduction to Open Software Networking Software
: Part I

April 16, 2013

CS Hong

What is Open Software

- Open Source Software (OSS) generally refers to software for which the source code is available and which the licensing scheme permits the user to modify it and redistribute it in modified or unmodified form.

GNU copyleft

- Nobody should be restricted by the software they use. There are four freedoms that every user should have:
  - the freedom to use the software for any purpose,
  - the freedom to change the software to suit your needs,
  - the freedom to share the software with your friends and neighbors, and
  - the freedom to share the changes you make.
- When a program offers users all of these freedoms, we call it free software

GNU

- Goal: to bring a wholly free software operating system into existence, a Unix-like computer operating system developed by the GNU Project

- is a recursive acronym for *"GNU's Not Unix!"*, chosen because GNU's design is Unix-like, but differs from Unix by being free software and containing no Unix code

- Development of GNU was initiated by Richard Stallman in 1983

- GNU Manifesto: free to study the source code of the software they use, free to share the software with other people, free to modify the behavior of the software, and free to publish their modified versions of the software : GNU General Public License(GPL)

Linux Overview
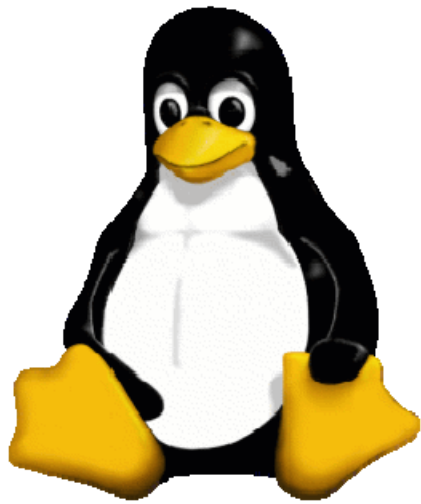
Tux

Official Mascot of Linux Kernel

Brief History of GNU/Linux

- 1985 Free Software Foundation (FSF) founded by Richard Stallman. Along with other programmers creates the tools needed to make a UNIX compatible OS

- 1985 Professor Andy Tannenbaum creates a UNIX like operating system based on System V Unix for the IBM PC & PC/AT computers. It is called Minix.

- 1989 Richard Stallman releases GPL and GNU software but lacks a free kernel.

- 1991 Building on the concepts in Minix, Linus Torvalds (Finnish college student) develops Linux along with help from other users on the web.

What Linux is

- Strictly speaking Linux refers to the kernel
- GNU/Linux more accurately describes the Operating System. Linux Kernel combined with GNU utilities and libraries
- Distribution – GNU/Linux bundled with other applications. Examples Red Hat Linux, Debian, Ubuntu, Suse, Knoppix, etc.
- Distributions can be compiled and maintained by an individual or corporation. Can be small (single floppy disk) or span several CD/DVDs.
- www.distrowatch.com for more information

Linux Distributions

- Red Hat (Enterprise) Linux is probably the largest commercial Linux vendor. Focus on stability and long term support. Red Hat sponsors a community version (Fedora) which undergoes a more rapid development cycle.

- KNOPPIX – The first Live CD version. The OS can be booted from, and run completely off the CD. KNOPPIX is based on Debian Linux. Many vendors now offer Live versions. No installation necessary.

- Ubuntu based on Debian Linux. Ubuntu claims to be most popular desktop version. Many applications and excellent "update mechanism" contribute to its success. Revenue is created by selling technical support.

- Puppy – This distribution is small enough to be loaded entirely in RAM. This frees up the CD/DVD drive for other uses. Configuration can be saved to a file between reboots.

- Recovery Is Possible (RIP) good for data recovery of corrupted media or backup. Tools such as ddrescue allows copying media sector by sector. Can repair NTFS or FAT partitions.

Applications

- **Samba** is a free implementation of the Server Message Block (SMB) protocol used by Microsoft Windows network file system. Allows sharing files, print services, and can even act as a domain controller.

- **Apache** is the most popular HTTP server software on the internet.

- **Rsync** allows synchronizing directories/files locally or across a network. Suited for low bandwidth as only changes get propagated across network.

Other Open Source Software

- The GIMP - Adobe Photoshop clone
- Open Office - Office suite
- Mozilla Firefox - Web browser
- Mozilla - Thunderbird E-mail Client
- VLC - Media player
- Audacity - Digital audio editor
- MySQL – Database
- Ghostscript - Postscript interpreter

Possible Uses

- Various tools permit user to recover data, repair/modify partitions, backup and clone various types of media. "DD type" tools allow copying a storage device by blocks/sectors. This allows duplicating a device without "knowing the filesystem".

- Linux can natively communicate with DECNET protocol. This allows a Linux PC to act as an End Node in a VMS cluster. Other DECNET tools allow additional inter-connectivity between the two OSs.

- With Samba and Ghostscript a Linux PC can be a Windows PDF print server. Giving individuals the ability to create PDF files without buying Adobe Acrobat or installing additional applications on each computer.

Who uses it

- NSA created Security Enhanced Linux (SELinux) to develop better allow the control of dynamic security policies. The architecture they created has been mainstreamed into Linux and ported to other operating systems including Solaris, and Free BSD.
- IBM, HP, and Sun contribute heavily to Linux and OSS
- Even Microsoft's position on OSS has softened and they are now contributing code to various projects.

Miscellaneous Information

- The Linux Documentation Project provide HOWTOs and Guides to accomplish many Linux specific tasks.

- Linux runs on a wide range of processors from embedded devices such as cell phones to supercomputers employing Linux clusters. One estimate from Forbes.com estimates that 60% of the worlds top 500 computers runs Linux.

- The availability of the OS source code allows it to be customized and tailored to users needs.

Web sites for more info and software

- Sourceforge.net
- Freshmeat.net
- Distrowatch.com
- www.linux.org
- Tldp.org The Linux Documentation Project

GNU Free Software

- http://directory.fsf.org/wiki/GNU
- Networking Software :
  - **GNU MAC Changer**: a utility to manipulate a MAC address
  - **GNU SASL**: SASL network authentication library
  - **Gnuradio**: Software to create digital radio signals
  - **GNUtls**: A library implementing TLS 1.0 and SSL 3.0 protocols
  - **Gvpe**: Secure vpn network among multiple nodes over an untrusted network
  - **Lsh**: Free implementation of the **SSH** protocol
  - **Osip**: Library supporting the Session Initiation Protocol

GNU Free Software (2)

- **Radius**: Remote authentication and accounting system
- **SIP Witch**: a pure SIP-based office telephone call server that offers generic business telephone system features
- **Shishi**: Free implementation of the Kerberos 5 network security system
- **WebSocket4J**: a WebSocket protocol implementation in Java
- **Zebra**: Implementation of routing protocols
- Hardware : H-Node Hardware Database A listing of hardware that supports free software : http://h-node.org/

- 1984   BSD 4.2 ships with routed (RIPv1)
- 1986   Fuzz Ball PDP-11 NSFNet Routers
- 1988   Age of dedicated routing machines
  - Cisco, Proteon, Wellfleet, ACC
- 1992   Gated Consortium Formed
- 1996   GNU Zebra
- 2002   Quagga, XORP

Routing Platform : Zebra

- ## by Kunihiro Ishiguro

- ## Zebra makes your computer a router!
  - And a Gateway router too!!
- ## Benefits
  - Test new configurations
    - ✓ New routing policies
    - ✓ New protocols
  - Don't disturb the production network
  - Use cheap PCs instead of expensive routers

Zebra features
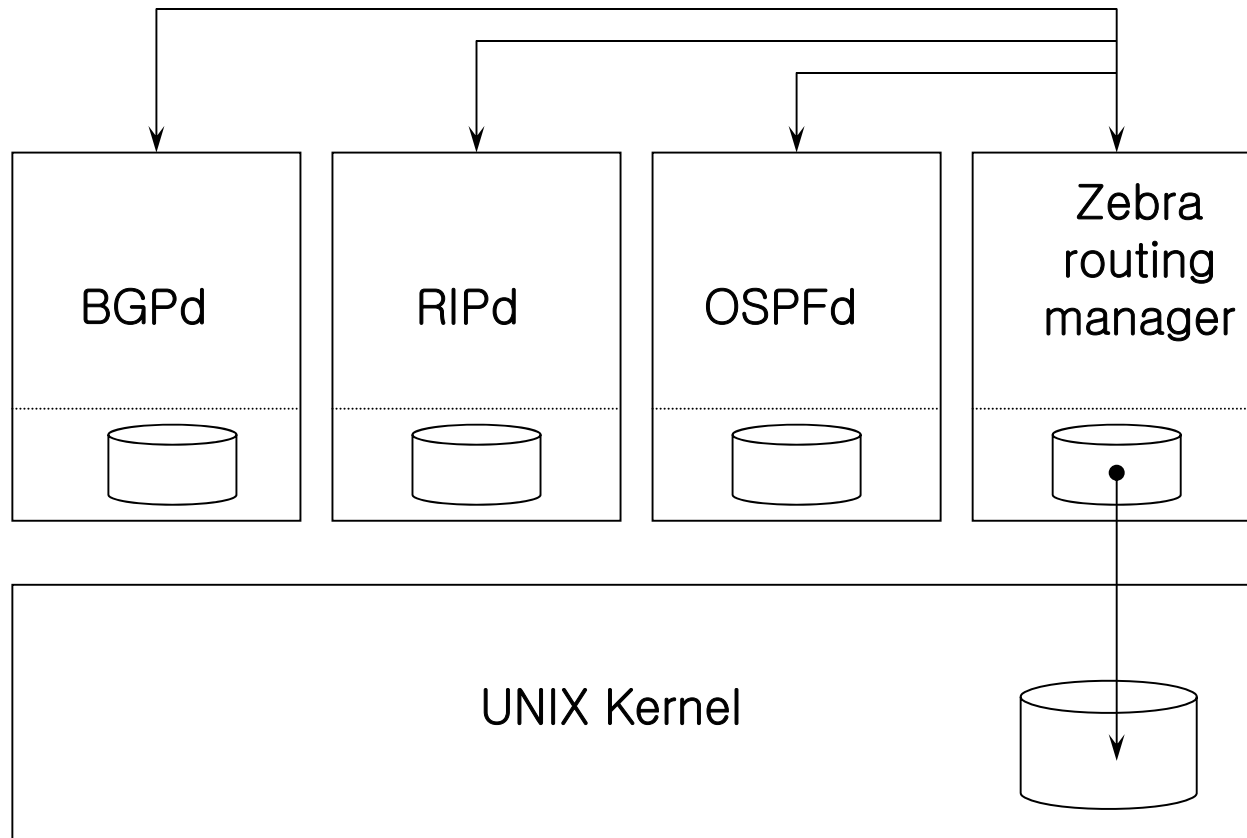
- Features
  - Can handle unlimited # of Interfaces
  - One process per one protocol
  - Route Server & Route Reflector integrated
  - Multithread based technology
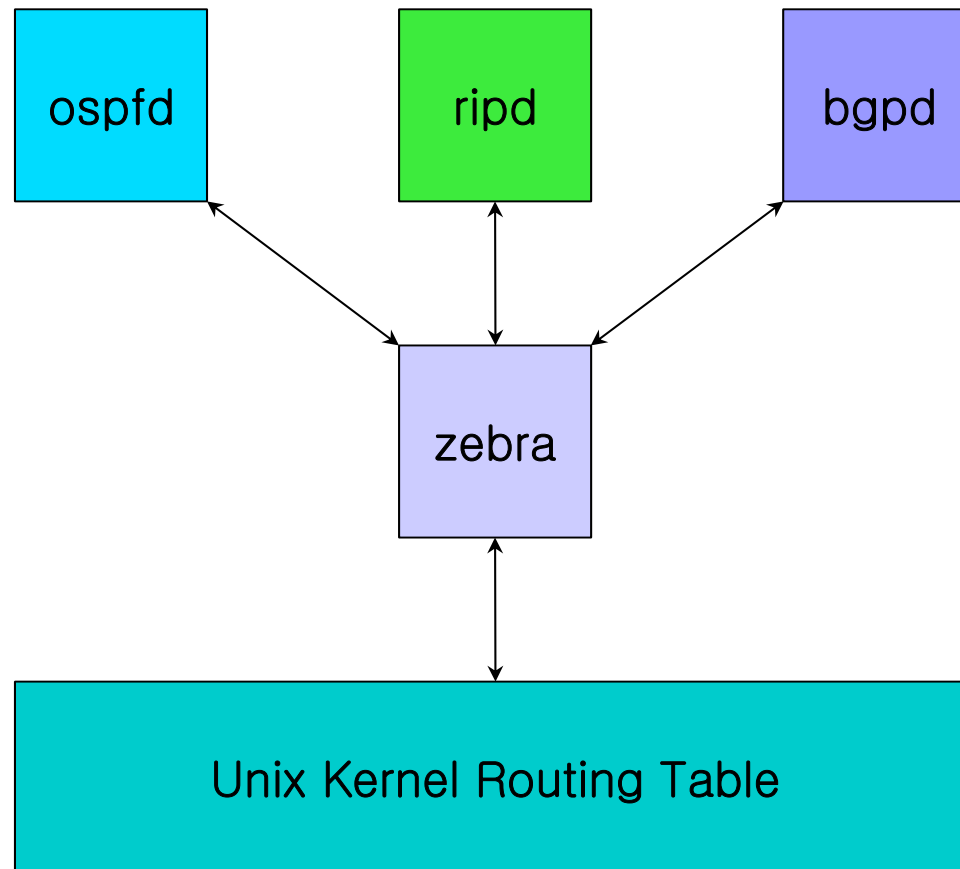 - Policy control and filtering

Routing Platform : Quagga

- Modular Design
- One process per protocol
  - bgpd, ospfd, ripd
- One main controlling process
  - zebra
- Extensible

- Modular Design
- One process per protocol
  - bgpd, ospfd, ripd
- One main controlling process
  - zebra
- Extensible

Quagga Architecture Diagram

- RIPv1, RIPv2, RIPng
- OSPFv2, OSPFv3
- BGP-4, BGP+
- BGP route server and reflector
- IPv6
- Supported RFCs
  - 1058 RIPv1, 2453 RIPv2, 2080 RIPng
  - 2328 OSPFv2, 2740 OSPF for Ipv6
  - 1771 BGPv4, 1965, 1997, 2545 BGPv6, 2796 BGP Route Reflection, 2858 Multiprotocol extensions, 2842 Capabilities Advertisement

Quagga Supported Platforms

- **GNU Linux**
  - Debian, RedHat, SuSE, Slackware
  - Kernels 2.2.x - 2.4.x
- **FreeBSD**
  - versions 4.x and 5.x
- **OpenBSD**
  - version 3.x
- **NetBSD**
  - version 1.4
- **Solaris**
  - 2.6 and version 7

Hardware Requirements

- CPU   Intel 2.0 – 3.0 Ghz
- Memory   512MB
- Disks   18GB
  - RAID-1 (optional)
  - SCSI or IDE
- Ethernet Interfaces
  - 2 x 10/100 Intel, 2 x 10/100/100 Broadcom
- Redundancy
  - hot spare serves as backup to N production units

- *eXtensible Open Router Platform*
- Open source router software suite, *designed from the outset with extensibility in mind.*
  - Main core unicast and multicast routing protocols.
  - Event-driven multi-process architecture.
  - BSD-style license
  - 560,000 lines of C++

XORP Contributions

- Staged design for BGP, RIB.
- Scriptable inter-process communication mechanism.
- Dynamically extensible command-line interface and router management software.
- Extensible policy framework.

First fully extensible, event-driven, open-source routing protocol suite: www.xorp.org.

RIP and RIPng:

- RFC 2453 (RIP version 2)

- RFC 2082 (RIP-2 MD5 Authentication)

- RFC 2080 (RIPng for IPv6)

OSPFv2:

- RFC 2328 (OSPF Version 2)

- RFC 3101 (The OSPF Not-So-Stubby Area (NSSA) Option)

XORP Status: BGP Standards

- **draft-ietf-idr-bgp4-26** (A Border Gateway Protocol 4 (BGP-4))

- **RFC 3392** (Capabilities Advertisement with BGP-4)

- **draft-ietf-idr-rfc2858bis-03** (Multiprotocol Extensions for BGP-4)

- **RFC 2545** (Multiprotocol Extensions for IPv6 Inter-Domain Routing)

- **RFC 3392** (Capabilities Advertisement with BGP-4)

- **RFC 1997** (BGP Communities Attribute)

- **RFC 2796** (BGP Route Reflection - An Alternative to Full Mesh IBGP)

- **RFC 3065** (Autonomous System Confederations for BGP)

- **RFC 2439** (BGP Route Flap Damping)

PIM-SM:

- draft-ietf-pim-sm-v2-new-11 (without SSM).
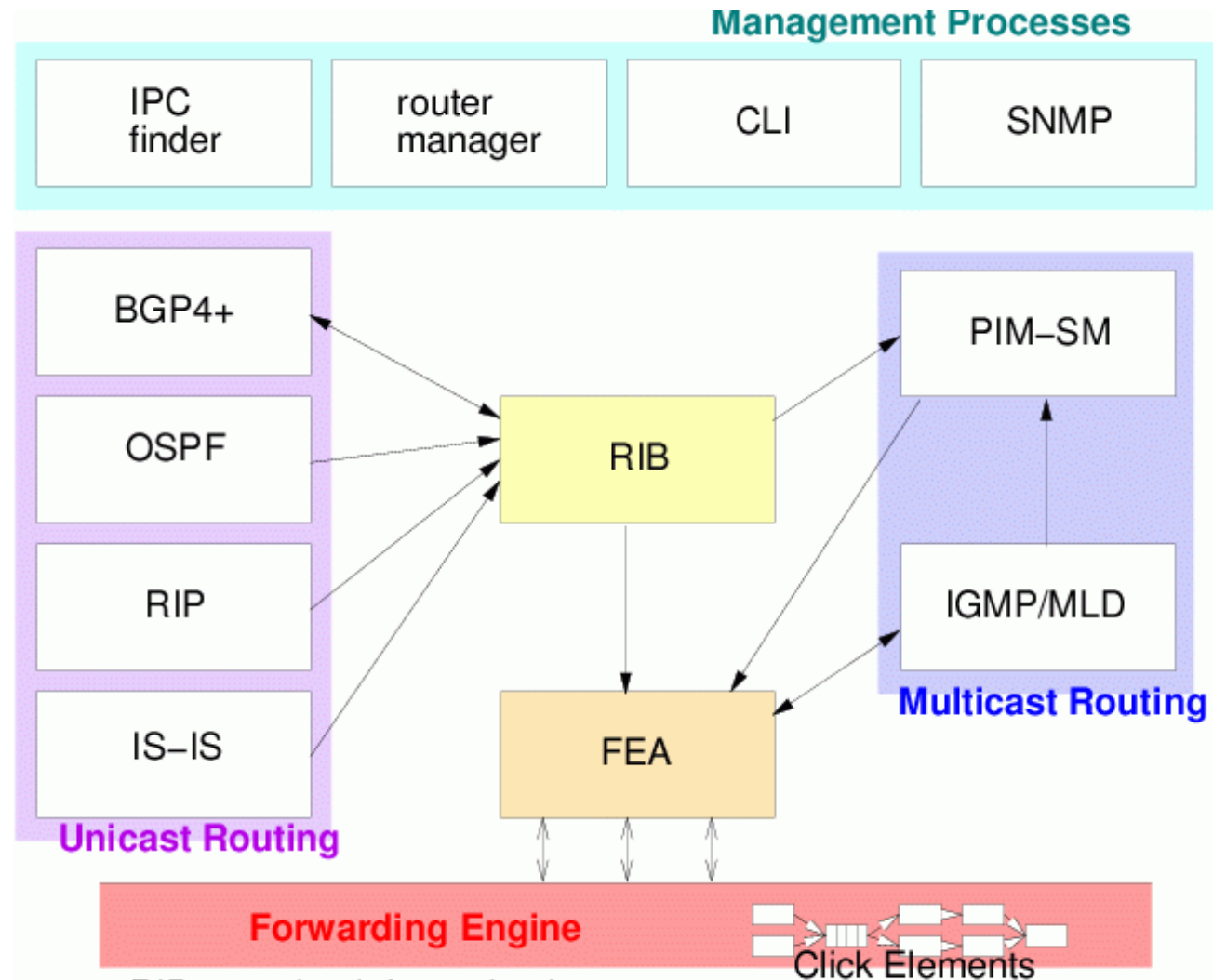- draft-ietf-pim-sm-bsr-03

IGMP v1 and v2:

- RFC 2236

MLD v1:

- RFC 2710

Multi-process architecture, providing isolation boundaries between separate functional elements.

Flexible IPC interface between modules

Management Processes

| IPC finder | router manager | CLI | SNMP |

Unicast Routing

BGP4+

OSPF

RIP

IS–IS

RIB

FEA

Multicast Routing

PIM–SM

IGMP/MLD

Forwarding Engine

Click Elements

RIB = routing information base
FEA = forwarding engine abstraction

- Why GNU Radio?
- Extensive knowledge involved
- What is implemented currently?
- Library
- Architecture
- Development environment
- Development Boards

Why GNU Radio?

- Almost free!
  - All the software are free (Python and C++ source code/linux environment)
  - In most condition, no need expensive RF test machine!
  - No need to purchase development and emulation tools
  - Only a development board needed (Universal Software Radio Peripheral)
- Flexible
  - Software:
    - ✓ reconfigurable for many other modulation methods for both standardize radio or self-defined radio
    - ✓ it is possible to improve the quality of the received signal by utilizing, in software, certain mathematical algorithms
  - Hardware:
    - ✓ Rx and Tx are selectable
    - ✓ Intermediate frequency is controllable
- Best choice for research use and radio amateur

Extensive knowledge involved

- Software and environment:
    - Python/Numeric python library/wxPython
    - C++/boost C++ libraties
    - Linux environment and lots of support packages: FFTW/cppunit/SWIG/SDCC/
    - GNU Radio architecture
- Communications and RF:
    - DSP
    - Digital communications
    - Wireless communications theory
    - FPGA and Assemble language may be used

What is implemented

- Base System
  - Provides the runtime and various signal processing primitives
- Hardware Support
  - Universal Software Radio Peripheral (USRP)
- Audio Device Support
  - ALSA (Advanced Linux Sound Architecture)
  - OSS (Open Sound System)
- Graphics Support
  - wxPython based GUI
  - SDL video library
- General Signal Processing
- Specialty Application Areas

Library

- Communication related implementation
  - AM demodulation
  - Differential BPSK / QPSK
  - GMSK modulation / demodulation
  - Narrow band FM transmitter / receiver
  - Wide band FM transmitter & broadcast FM receiver
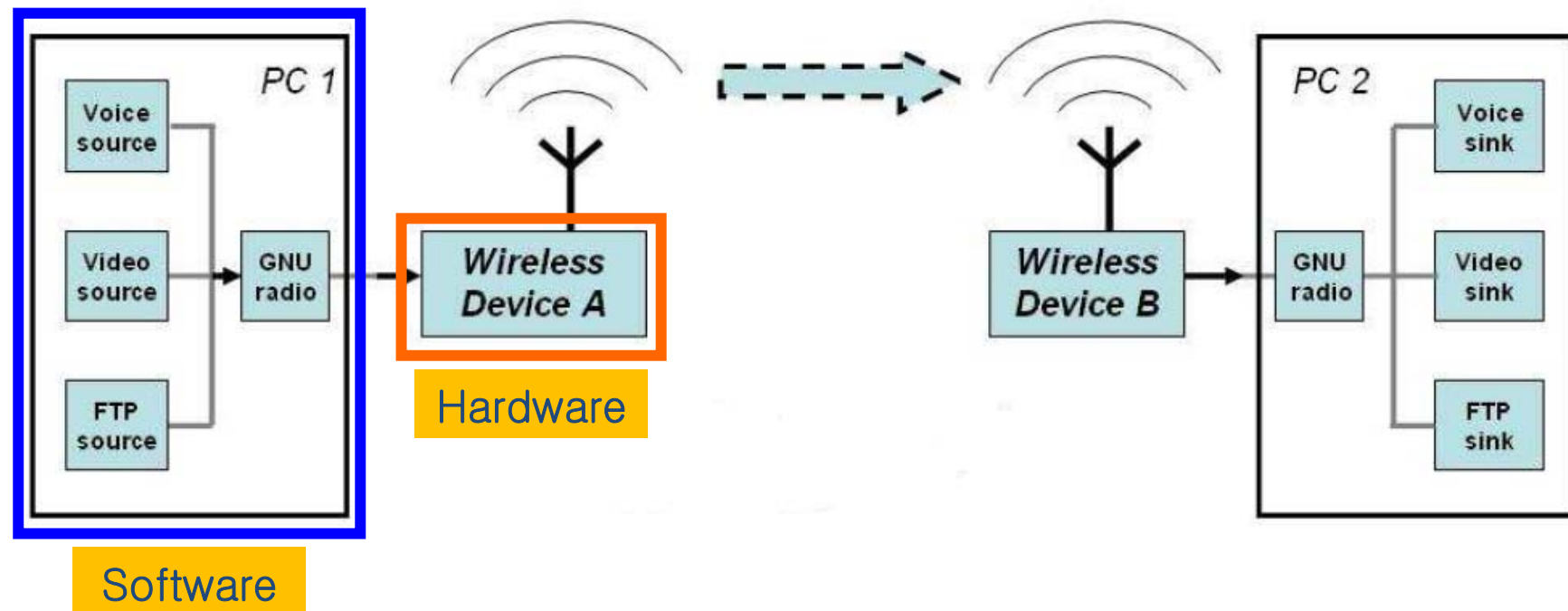
- GNU radio utilities
  - CRC generator
  - Socket setup (TCP / UDP)
  - Compute frequency response of a digital filter
  - Control National IMX2306 & SDR-1000 frequency synthesizer
  - Some utilities
    - ✓ Convert unsigned mask into signed integer
    - ✓ Gcd, Lcm, Log2
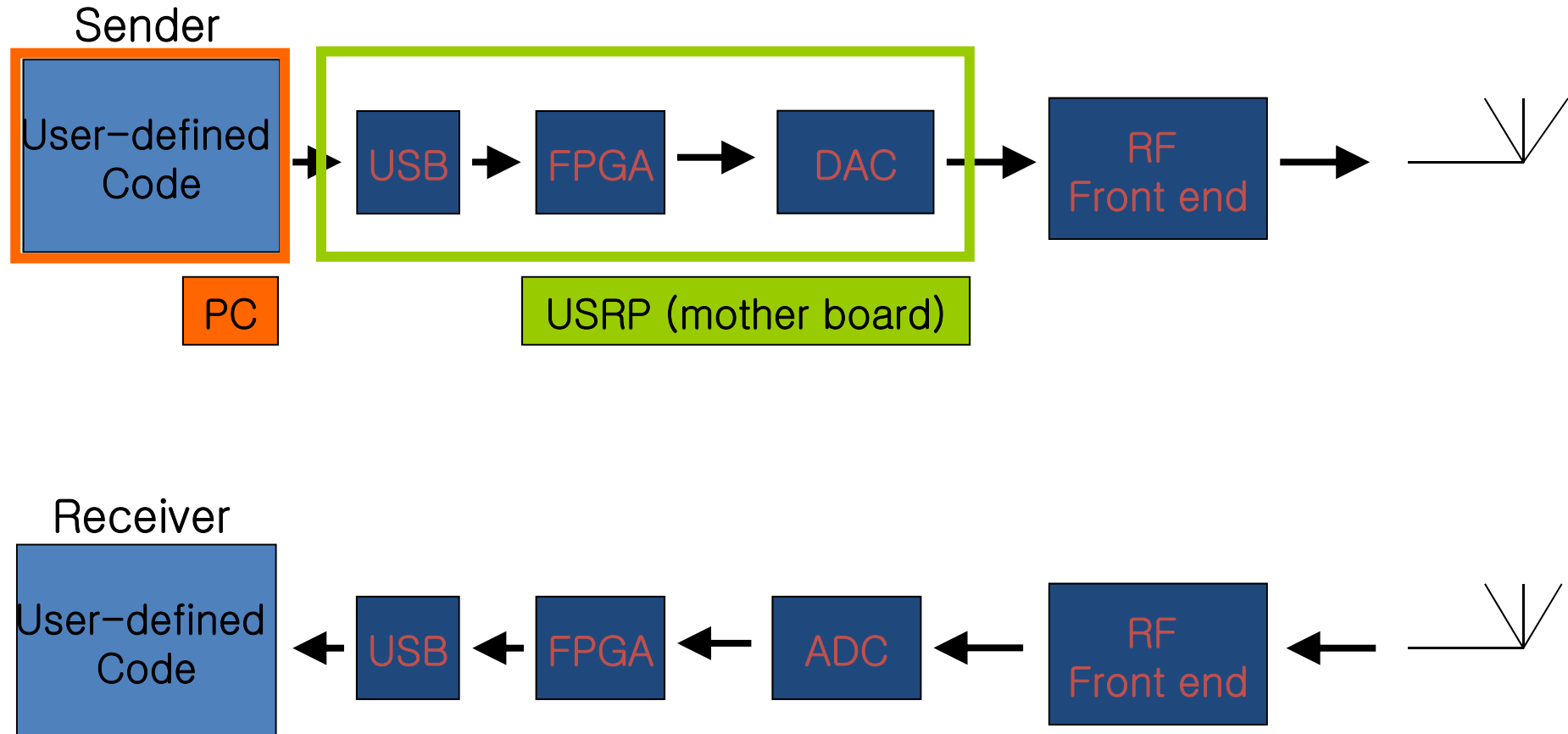   - ✓ Return input 'x' that is reverse order

- GUI examples
  - Provide window application for different usage
  - FFT sink test
  - wxPython EditBox, Slider
  - Drawing
  - Waterfall sink test
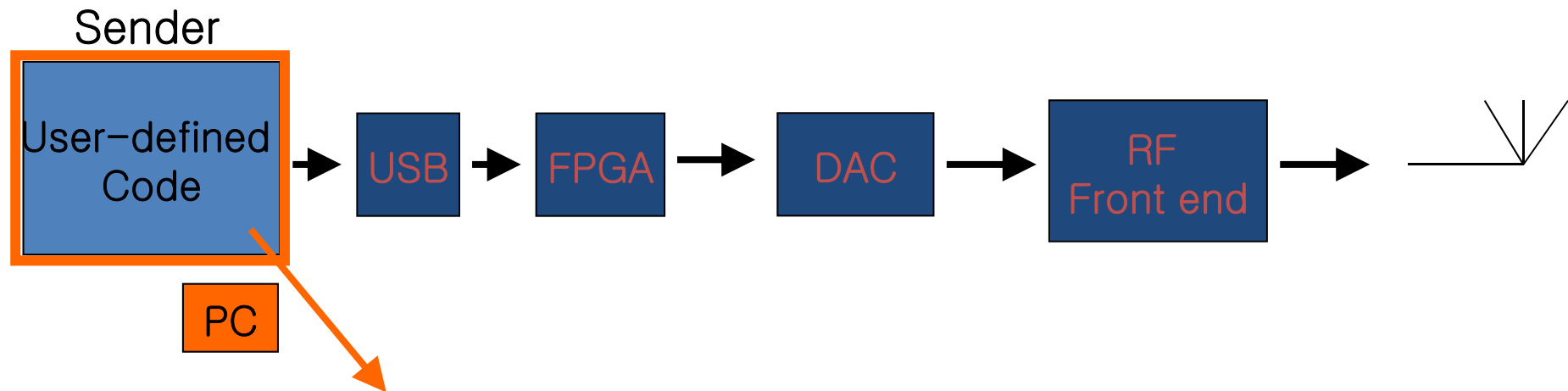 - Oscilloscope Test Application

- pager
  - Create USRP source object supplying complex floats
  - Flex pager protocol demodulation block

Software

Hardware

Sender

User-defined Code

PC

USB → FPGA → DAC

USRP (mother board)

RF Front end

Receiver

User-defined Code

← USB ← FPGA ← ADC ← RF Front end

Sender

```
User-defined
Code
```
→ USB → FPGA → DAC → RF Front end →

PC

- GNU radio has provided some useful APIs
- What we are interested in at this time is how to use the existing modules that has been provided in GNU radio project to communicate between two end systems

- How these modules co-work?
  - **C++**
    - ✓ Performance-critical modules
 - **Python**
    - ✓ <u>Glue</u> to connect modules
    - ✓ Non performance-critical modules

Development Boards

Description	Price
USRP Motherboard	$700.00
BasicTX -- 2 MHz to 200 MHz Transmitter	$75.00
BasicRX -- 2 MHz to 300+ MHz Receiver	$75.00
LFTX -- DC-30 MHz Transmitter	$75.00
LFRX -- DC-30 MHz Receiver	$75.00
TVRX -- 50 MHz to 870 MHz Receiver	$100.00
DBSRX -- 800 MHz to 2.4 GHz Receiver	$150.00
RFX400 -- 400-500 MHz Transceiver	$275.00
RFX900 -- 800-1000MHz Transceiver	$275.00
RFX1200 -- 1150 MHz - 1450 MHz Transceiver	$275.00
RFX1800 -- 1.5-2.1 GHz Transceiver	$275.00
RFX2400 -- 2.3-2.9 GHz Transceiver, 20+mW output	$275.00

- Four 64 MS/s 12-bit analog to digital Converters
- Four 128 MS/s 14-bit digital to analog Converters
- Four digital downconverters with programmable decimation rates
- Two digital upconverters with programmable interpolation rates
- High-speed USB 2.0 interface (480 Mb/s)
- Capable of processing signals up to 16 MHz wide
- Modular architecture supports wide variety of RF daughterboards
- Auxiliary analog and digital I/O support complex radio controls such as RSSI and AGC
- Fully coherent multi-channel systems (MIMO capable)
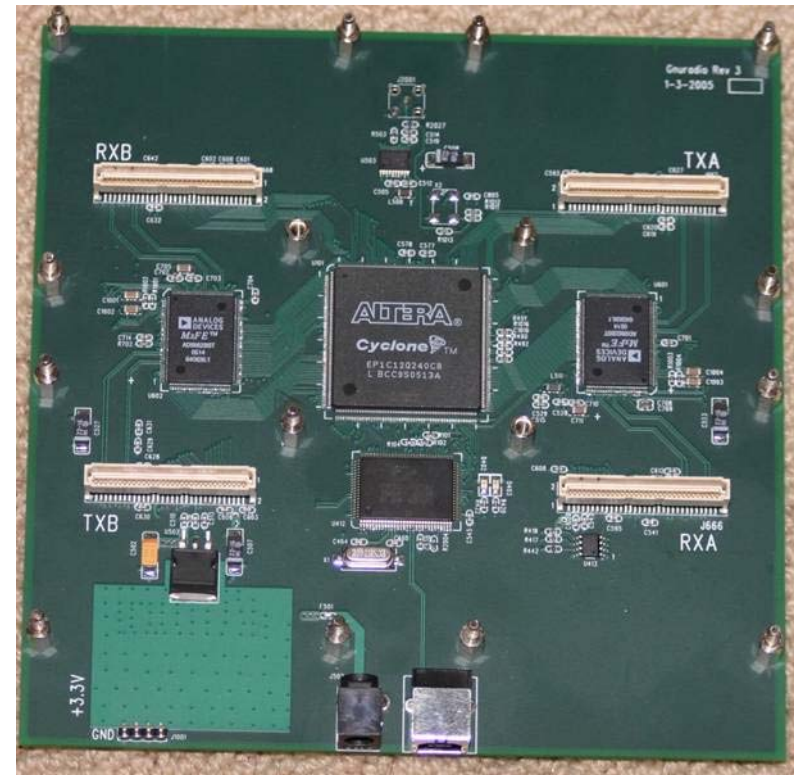
- Ubuntu already added GNU Radio packages to their repositories

  - System > Administration > Synaptic Package Manager

  - Search "gnuradio"

  - Select all related gnuradio libs and then apply to install

- To install from console by using command line:

  - Check: http://gnuradio.org/redmine/projects/gnuradio/wiki/UbuntuInstall

Installation Guide for Fedora

1. **Install all basic required package for building GNU radio**

   $ yum groupinstall "Engineering and Scientific" "Development Tools"

   $ yum install fftw-devel cppunit-devel wxPython-devel libusb-devel guile
      boost-devel alsa-lib-devel numpy

2. **Download and build GNU Radio**

   $ svn co http://gnuradio.org/svn/gnuradio/trunk gnuradio

   $ ./bootstrap # Do NOT perform this step if you are building from a
      tarball. $ ./configure $ make $ make check $ sudo make install

3. **Install small device C compiler for USRP**

   $ yum install sdcc

   $ export PATH=/usr/libexec/sdcc:$PATH

4. **Export python environment parameter**

   $ export PYTHONPATH=/usr/local/lib/python2.4/site-packages

5. **Test an USRP application**

   ./usrp_wfm_rcv.py -f 96.3

http://gnuradio.org/redmine/projects/gnuradio/wiki/FedoraInstall

- Embedded Linux for wireless routers
  - Full command-line environment
  - Appropriate device drivers
  - WWW interface for simple configuration
- Why screw around with flash on my router??
  - Tweak the service configuration
    - ✓ firewall
    - ✓ local DNS
    - ✓ DHCP
  - Get more out of the hardware
    - ✓ signal strength
    - ✓ afterburner
    - ✓ VLAN switch - DMZ

Linksys WRT54GL

- CPU/Filesystem resources
  - Broadcom 5352 200Mhz
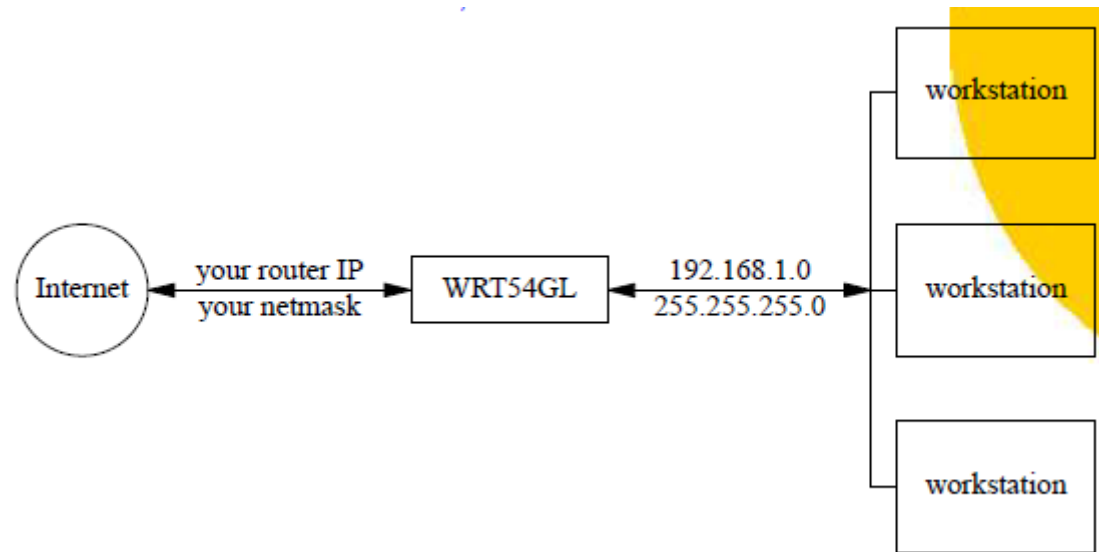 - RAM 16 MB
  - Flash 4 MB
- Networking
  - Wireless interface (including "afterburner")
  - Ethernet bridge (bridges to wireless)
  - VLAN switch
  - Other Hardware:
- Other Hardware:
  http://wiki.openwrt.org/TableOfHardware

- Default OpenWRT setup
  - router does address translation for hosts
  - can forward service requests to servers
  - allocates local IP
  - answers DNS queries

Important Safety Tip

- The WRT54G (not GL) is much less capable:
  - 2 MB RAM 8 MB flash
  - VxWorks OS
  - very brickable
- New from Amazon
  - WRT54G: $49.99
  - WRT54GL: $64.99
  - WRT54GS: $69.99
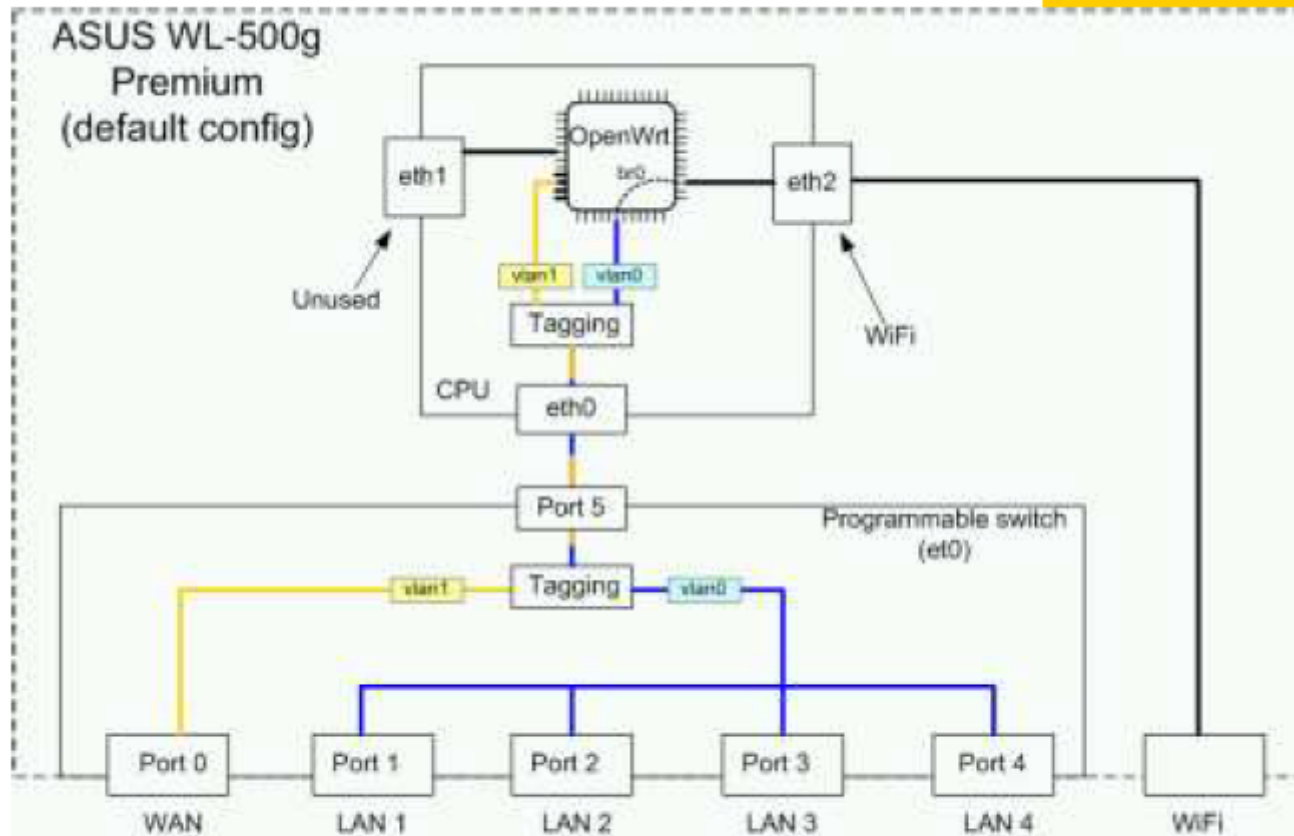 - WRST54GS: $99.99 - USB!

Mmmmm. Block diagram

Image from http://wiki.openwrt.org/OpenWrtDocs/NetworkInterfaces

VLANs and a DMZ

Image from http://wiki.openwrt.org/OpenWrtDocs/NetworkInterfaces

References

- http://www.zebra.org
- A part of PPT by Russell Sutherland, Computing and Networking Services, University of Toronto
- www.xorp.org
- *GNU Radio* - University of Florida
- www.OpenWRT.org