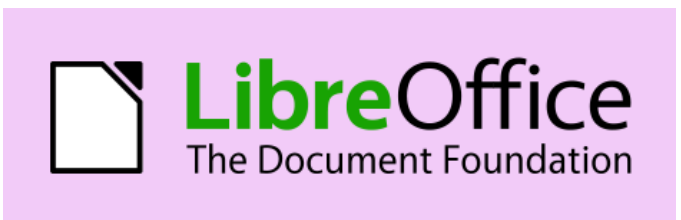
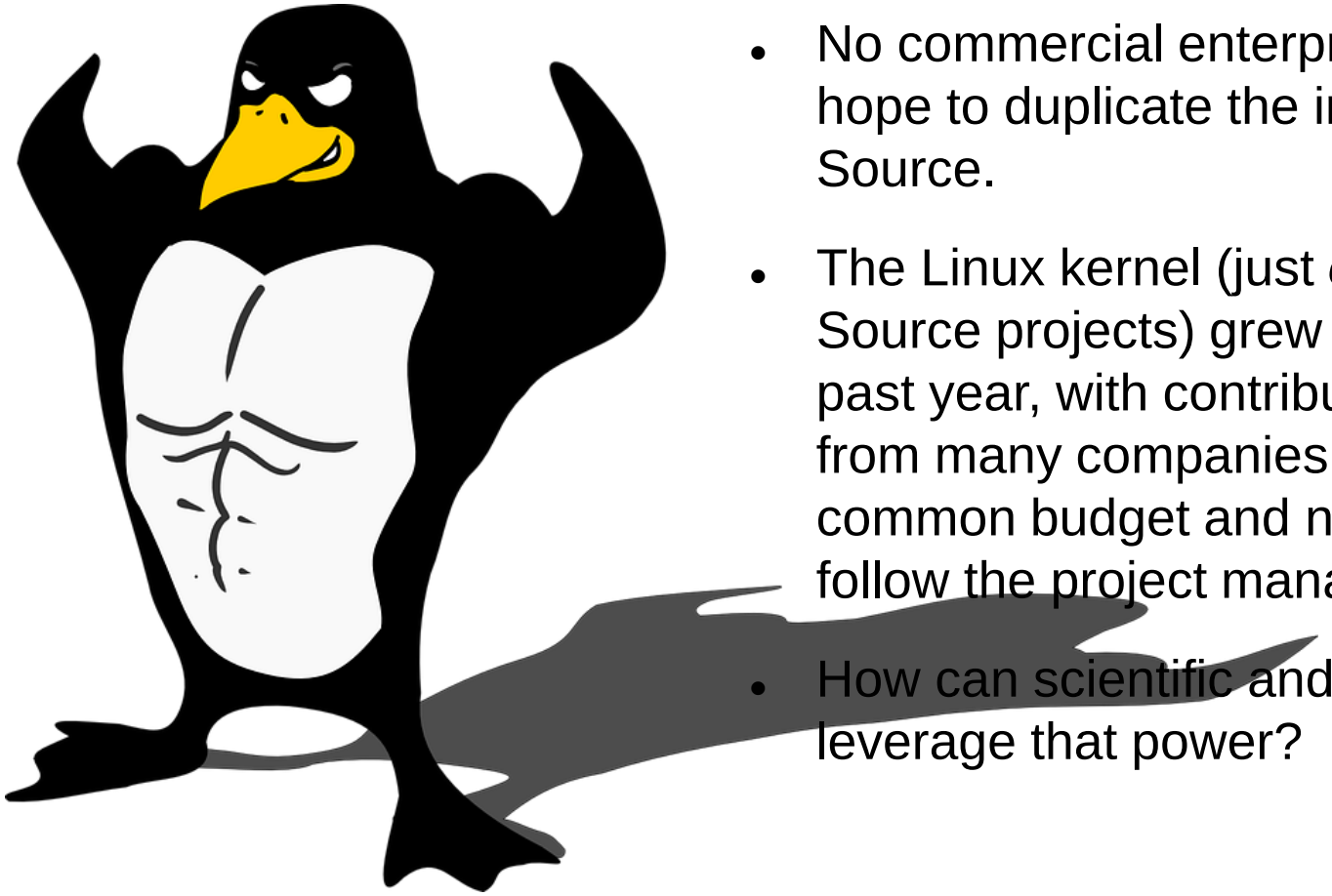
Open Source Isn’t Rocket Science, But Rocket Science Is Open Source!

Bruce Perens K6BP <bruce@perens.com>
President, Open Research Institute

This presentation was produced using only Open Source Software.



Why Open Source Now?



- No commercial enterprise, no matter how large, can hope to duplicate the innovative power of Open Source.
- The Linux kernel (just *one* of *thousands* of Open Source projects) grew by 225000 lines over the past year, with contribution from 3300 developers from many companies and institutions, with no common budget and no compulsion for anyone to follow the project management.
- How can scientific and research organizations leverage that power?

Nobody Would Have Believed This

- In 1985, when Richard Stallman released the GNU Manifesto.
- And in 1998, When I Announced “Open Source” To The World.
- Nobody really believed that we could do much of anything.

Imagine Telling Your Friends This In 1998

- My friends, who I have never met, from the Internet, and I will write a free encyclopedia that has a Million articles and puts the commercial ones out of business.
- We’re going to make a free operating system that becomes the basis of everything you do – the Internet, air traffic control, rockets in space, you name it.
- They’d wonder if you were using TCP/IP, or LSD.

Economics

- Open Source turned out to be the most effective way to operate software development among large, very loosely organized teams.
- But *understanding its economics can be difficult.* Obviously, there *are* economics behind the vast adoption of Open Source.
- To understand the economics, we must first understand what is business-differentiating and non-differentiating software.

Non-Business-Differentiating

- Business-differentiating software makes a business more desirable than its competitors *to the customer.*
- It must be something that the customer sees and interacts with directly.
- Infrastructure, middleware, operating systems, cellular radio protocol stacks, etc. are all *non-business-differentiating.* In general, the customer *only sees their effect when they fail.*

Non-Business-Differentiating (2)

- Businesses reduce cost and increase profit if they **distribute the cost and risk of non-business-differentiating software development across multiple companies** through Open Source.
- They thus have more money to spend on their business-differentiators, *the most critical things they can develop.*

For Science: Open Source Is Often The Best Means of Technology Transfer

- But the question for science is not so much *how will I make money?*, as it is *what is the best way to distribute the fruits of my research to society?*
- As proven by organizations like CERN, Open Source that can be used and improved by everyone is often a better technology transfer policy than monopoly patents that mainly benefit one company.
- This is especially true for software, and other kinds of works which **do not need a huge front-end investment before they can be brought to customers, which would be returned through patent revenue.**

So, *How Will I Make Money?* Is The Wrong Question

- Most businesses *save* money by using Open Source, and they transfer these saved development dollars to development of their business differntiators, or something else important to their business.
- Thus, the effectiveness of the business, and its profit, is increased.

Old vs. New Technology Transfer

- The monopoly incentive provided by patents is *only* important for works that need a very large front-end investment to fund their commercialization. That's not software, or many other kinds of modern work.
- For most work, **monopoly incentives actually harm adoption of technology,** simply because it's easy for one business to fail, leaving the patent mired in bankruptcy, held by a bank that doesn't understand it, or sold to a *patent troll* which only *impedes* entities that produce technology.
- When many businesses have access to the invention through Open Source, at least one is likely to succeed in bringing it to the public.
- This is simply using the market in a *capitalistic* way. Monopolies are anti-capitalist.

Open Source Development *Has Evolved*

- ***Early Open Source Man*** could only develop for ICT *(Information and Communications Technology),* using primitive tools: a laptop, GNU C Compiler.

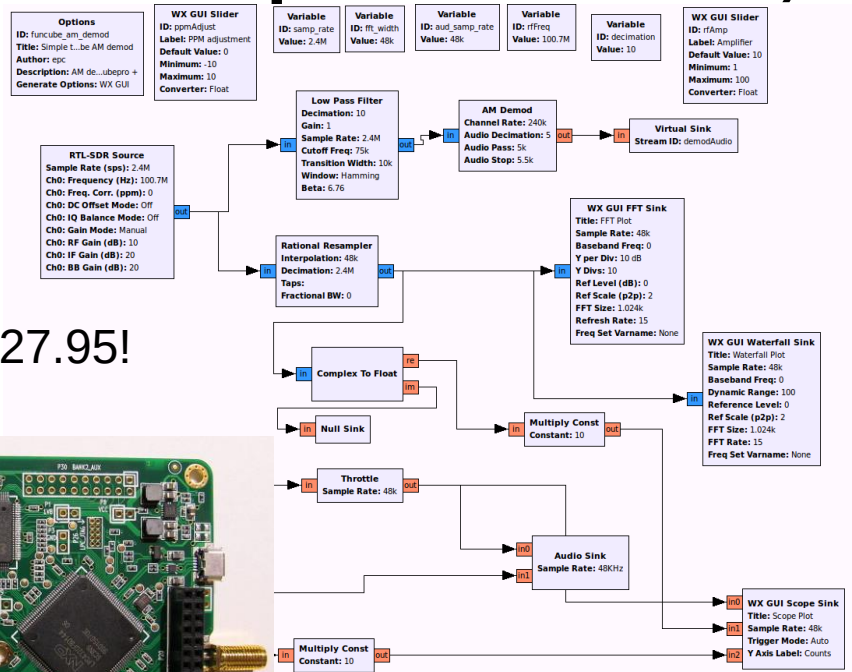
RF and ASIC Could Only Be Developed With A Multi-Million or Billion-Dollar Lab

What Changed?

- The advent of **Software-Defined Radio** (SDR) and **inexpensive gate-array development platforms** put laboratory capabilities formerly accessible only to large corporations within the reach of the *individual!*
- There has also been a rise of other *empowering tools,* such as 3D printing, small CNC mills, etc.

Modern Open Source Person
Develops For RF, Gate Array

RTL-SDR USD$27.95!



GNU Radio Sketch:
Weather
Radio with GUI Display of FFT and Waterfall Data

HackRF Open Hardware SDR

ORI Senior Scientist
Michelle Thompson W5NYV
(formerly senior engineer, Qualcomm Globalstar and Handset divisions). Develops space comms.

Open Source in Space

- Debian first flew on the Space Shuttle in 1997, as part of
- The Microgravity Science Lab.

- SpaceX is heavily Open-Source-based.

- Dragon and Falcon 9 vehicle run Linux.

- 6 USRPs (Universal Software Radio Peripheral) on each
- Falcon 9. Each saves $200,000 over a purpose-built
- space radio transceiver.

- SpaceX Ground Facilities Run LabView on Linux.

Open Source Cubesats

- Entirely Open-Source and Open Hardware Cubesats are now the state of the art, and are being pursued by multiple organizations.
- But what do we do about national munitions export laws like the United States *ITAR* (International Trafficking in Arms Regulations) and *EAR* (Export Administration Regulations)?

Open Source Is Often The *Only* Viable Strategy for International Collaboration

- Munitions laws are intended to keep advanced weapons and their technology from falling into the hands of nations that might use them against us.
- Munitions laws like ITAR and EAR are designed to protect the ***proprietary*** technology of companies from being revealed to other nations.

What If It’s Not Proprietary?

- Munitions laws like ITAR and EAR have **legal carve-outs** for basic science, scientific conferences, scientific publications, and public libraries.
- If you publish your work as Open Source *as you make it,* and keep your project discussions on public mailing lists, *It is not subject to ITAR or EAR.*
- Physical objects, rather than computer data, are still subject to full ITAR and EAR regulations.

Open Research Institute

- Michelle Thompson and I spun *Open Research Institute* (ORI) off of AMSAT in order to pursue public development of technology, *based in the US* (unlike LibreSpace, which is based in Greece) that would otherwise be restricted by our own country's munitions export laws.
- ORI literally has no secrets. It's operated so that all work becomes public as it is made.
- We're just starting up.

Other Troublesome Laws

- US licenses space photography (private remote sensing) through the National Oceanic and Atmospheric Administration (NOAA).
- Must not exceed resolution of current publicly available sources.
- Imaging of Israel and some other battle zones is restricted by US law.

Other Troublesome Laws (2)

- FCC licenses satellite communication, but uses the license to enforce other satellite parameters such as size, de-orbit.
- Declines to license satellites smaller than 1U, because NORAD only has the technical capability to effectively image a *certain number* of objects with very small radar-reflective profiles.

ORI Digital Communication Project

- DVB-S2X Open Source implementation, without television as the main payload.
- Both Satellite and Ground Station.
- Facilitates consolidation of thousands of slow-to-medium-speed uplinks into a single high-speed downlink stream.
- Thus, a satellite can illuminate an entire continent, and all stations illuminated can hear each other.
- Original concept was to faciliate continent-wide emergency communications via Amateur Radio.

ORI Cubesat Project

- Just starting, and looking for technical leadership.
- Digital Communications for Radio Amateurs.
- Members want high-orbit, are bored with low-orbit analog Amateur satellites.
- Thus, it must be radiation-tolerant.

Radiation

- Radiation can cause soft errors, but the *real* problem is **physical damage to components.** Components eventually fail due to radiation dose if they remain in orbit long enough.
- **Radiation-induced latch-up** between a transistor detail on the IC and the substrate can physically destroy a gate.
- One solution is to use a non-conductive substrate, thus silicon-on-insulator or silicon-on-sapphire.

Inexpensive Strategy

- Simply resetting a latched-up chip is insufficient if the latch-up current damages the device.
- An inexpensive strategy is to use a component that is inherently radiation tolerant due to its fabrication details. Some gate-arrays meet this requirement for radiation doses of around 100 kRad.
- CERN is creating a Risc-V CPU on Microsemi IGLOO with triplicated registers (and possibly triplicated logic).

Expensive Strategy

- Vorago has fabricated an ARM Cortex M0 on silicon-on-insulator that is useful up to about a 400 kRad dose, and extreme temperatures.
- Cost is about USD$1000 for what might otherwise be a USD$5 chip.
- Crowdfunded fabrication of CERN design on silicon-on-insulator might be more affordable.

The Future

- Open Source will assume a dominant role in academic and non-profit small satellites, and will provide the basis of many commercial small satellites.
- ORI plans to organize an online library of Open Source space technology in one place.
- We plan to commoditize manufacture of Open Source designs for use by others, at lower cost than current cubesat stores. Why make one when you can make and sell 100?

Contact The Author

Bruce Perens K6BP <bruce@perens.com>
President, Open Research Institute

This presentation was produced using only Open Source Software.