

# Opening That Which is Closed

Bruce Perens K6BP <[bruce@perens.com](mailto:bruce@perens.com)>

**Have Bruce Perens keynote your conference! See <http://perens.com/have-bruce-perens-keynote-your-conference/>  
Seeking my next academic fellowship. Part-time only. Write to me.**

Slides and Performance Copyright © 2017 Equipment Unit LLC.  
**Because this is a work of opinion, and I do not want my opinion to be distorted by others, it is not licensed as Open Source.** You may copy and redistribute it unmodified. You may make translations of the complete work only, without deliberate distortion, which remain my property and must be licensed under these terms. Usage generally considered to be fair use in courts of your jurisdiction is permitted. For additional rights, write to the author.

# Please Interrupt!

- I would rather have a feisty and argumentative audience than a sleepy and sullen one.
- I'm getting hard of hearing, so if you raise your hand I will try to get close to you before you speak so that I can hear you. But some of you are so loud that doesn't matter.

# About Bruce Perens

- Co-founder of the Open Source movement in software and creator of the *Open Source Definition*, the legal rules for Open Source licensing. Of course in this I am standing on the shoulders of Richard Stallman and *Free Software*.
- Founder of *No-Code International*, successfully eliminated Morse Code *testing* as a prerequisite for Amateur Licensing worldwide except for Russia. And there is more Morse Code *use* now than before!

# About Bruce Perens (2)

- Creator of *Busybox*, *Electric Fence*, and other interesting Open Source programs.
- Second *Debian* project leader and the person to break the *base system*, the core of Debian, out to multiple package developers who often weren't even on the same continent with each other, before anyone understood that you could do that and have the result actually work.

# About Bruce Perens (3)

- Participated in the creation of the 3D Computer Graphic Feature Film industry, 19 years in film and 12 at Pixar. Credited on two Pixar films, *Toy Story II* and *A Bug's Life*, worked on many others.
- Series editor, *Bruce Perens' Open Source Series*, 24 books with Prentice Hall PTR publishers. First book series under an Open Publication license, before Creative Commons existed.

# Bruce Perens Today

- **LegalEngineering.com**: Assist law firms and their customers with Open Source and other intellectual property and technical issues. This supports me while leaving time for other projects.
- **Algoram**: So far *not* successful in making SDR mobile transceivers and eventually HTs, first design was too noisy.
- Undisclosed start-up: Doing the VC thing again this year.

# Latest Academic Paper

- Publishing a paper on *Open Cars* in the next issue of the *Berkeley Technology Law Journal*, presented at Boalt Hall (Berkeley Law). Which is really cool because I'm not a lawyer, although my co-author is one.
  - Will you continue to have the right to open up the hood?
  - John Deere issue, Tesla battery capacity grant.

# GNU Radio, the Beltway Bandit Version

- Before GNU Radio, everybody would have expected this to come from TRW and its ilk, and indeed a system *did*.
- Cost USD\$60 Billion.
- Never actually worked to the customer's satisfaction.
- No longer maintainable.
- Nobody would ever have *believed* that we could do this in Open Source.
- This was Opening That Which is Closed.



# Open That Which is Closed

- Making things Open Source that are only available from proprietary vendors today, and where people still don't believe that Open Source can act successfully in the field.
- Historically, we've actually been pretty successful at this.
- Several operating systems and all of their supporting software.
- Steve Jobs actually told me that Open Source could not produce it's own GUI, two years before Apple based Safari on an Open Source GUI.
- Software galore.
- Rocket Science like Codec2, Open Source Line Echo Canceller, Opus, GNU Radio.

# How Can You Achieve Really Cool Things?

- Opening that which is closed is really hard!
- *You aren't qualified* to do what you dream of, and you may never be!
- Sometimes brains count, but much more often, it's *chutzpah*. Ignoring that you're not at all qualified and don't have a chance in hell of doing something.

# I Am Not Smarter Than You

- What I am talking about today does not take some sort of genius. I am not one.
- I am innumerate by the standards of this crowd (that means I'm bad at Math).
- I have no college degree.
- I am a survivor of a motor speech and coordination handicap, a decade of remedial speech education, and sometimes I still walk on my toes.
- Logically, I should not belong on the same program with Phil Karn, Michelle Thompson, Bob Mcgwyer, and you.
- This doesn't mean that the talk's over and it's time for lunch.

# It Starts With Anger

- I have really strong opinions about a lot of things. So strong that it can keep me up at night. A lot of them wouldn't matter to most people, at least at the start.
- I didn't like that there was still a Morse Code examination for radio amateurs.
- Steve Jobs told me, to my face, that *Windows* was going to take over the computing world, and he made his peace with Bill Gates and put a Windows laptop on his desk at Pixar (obviously, before he went back to Apple).

## It Starts With Anger (2)

- A fundamental part of ham radio *was* that you could build and understand every part of it. So, I didn't like that Amateur digital voice was being taken over by commercial products with locked-down codecs.

# For Me, Writing Was How I Dealt With Anger

- If I wrote about something I didn't like, and published the article, I would feel more in control of the issue and could go to sleep.
- This often meant that the writing would happen between midnight and three A.M.
- It generally did turn out that a lot of people felt the same way I did.
- People would write to me: "I wanted to explain this to a friend, but didn't have the words. You gave me the words."

# It Starts With Writing

- Once I wrote about something, I would be able to influence people's opinions, and sometimes helped people to get together on an issue where they didn't realize that anyone else felt the way they did.
- Not everyone can do persuasive writing well or be a public speaker. Some of us are better at making hardware or software, or technical documentation. That's just as useful.

# Some “Impossible” Things

- These are some things I worked on. In every case, there were lots of people working on them with me. I don't mean to take all of the credit. What I did was to influence people and get a bunch of them to work on the same thing.



# Creation of Open Source

- Obviously, Free Software came first.
- There were problems with the perception of “Free”.
- Richard’s presentation depended on an a priori acceptance of the value of software freedom.
- Rules for Open Source started as Debian rules and manifesto.
- Re-published as *Open Source Definition*.
- The world was strangely ready for this.

# No-Code

- The Morse code requirement existed for all Ham Radio licenses, for decades. For all but the lowest grades of Amateur Radio license in the U.S., it was specifically set at a speed *just high enough* that you could not decode the dots and dashes in your head, but would have to learn all of the letters *as a sound*. To get a ham license, you had to pass a test in receiving them by ear and writing down what you heard.
- I felt that this deterred people who would be good hams, and were interested in the present and future rather than ditting and dotting like an 1850's telegrapher.

# Morse Code, and the Purpose of Ham Radio

- You don't get all of those frequencies to operate a private club. Amateur Radio *must* operate in the public interest. Its purpose is spelled out in U.S. Federal law.
- I felt that Ham Radio should have the primary purpose of **being a tool for learning and experimentation.** There are things that you can learn on ham radio that nothing else allows you, because you can own the entire physical layer, rather than be a client of someone else's network.

# Amateur Radio Fails at Most of It's Own Missions

- The emergency services mission of Amateur Radio is still important, but is declining.
- The military mission of having a pool of pre-trained radio operators is dead, dead, dead.
- Amateur Radio has not been a driver of radio innovation for half a century, we're trying to change that now.
- International friendship is still a good mission.
- But **education must be the main mission of ham radio in the future**, and today it is not directly mentioned in the FCC's current definition of Amateur Radio's mission.

# The Big Shut-Out

- The *publicly expressed purpose* of a 13 word-per-minute test on receiving Morse Code and writing down what was sent, when it was implemented, was **to limit the total number of radio amateurs.**
- There were less than 100,000 licensed hams in the U.S. at the time, and *they* were afraid of what would happen if many more people were allowed to become hams.

# Legal Requirements

- FCC did not impose a Morse code requirement on hams. Hams asked for it, and got FCC to make it law, and persuaded the International Telecommunications Union, a U.N. organization, to add a Morse Code requirement to the international telecommunications treaty.
- So, here I had to both change U.S. law, and change an *international treaty* to get rid of the Morse code requirement.

# At The Start

- Being a ham who was for ending the code requirement was *very* unpopular with other hams when I started.
- I did many presentations at Amateur Radio conferences. I expected to be punched out or spat upon, but was only cursed and shouted at.
- But it turned out that there were a lot of people “in the closet” who shared my opinion, and would “come out” when they could point at a credible speaker for the cause.

# Internationally

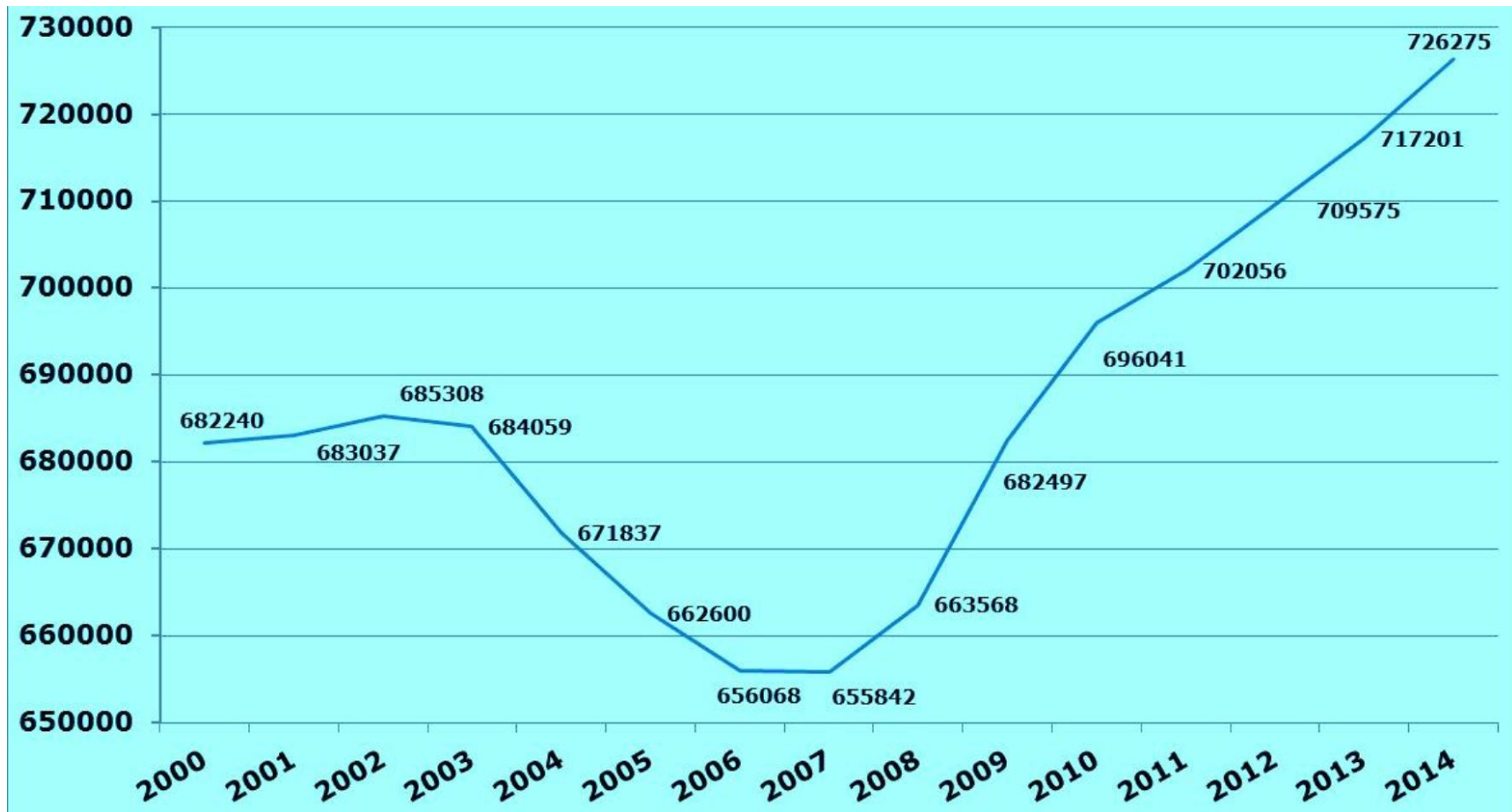
- I presented internationally, including in Iceland, where half of the hams in the country were in the room with me when I spoke. Iceland had a vote in the international ham association that was equal to the vote of the entire U.S.
- Others of the board and membership presented to FCC, local groups, and in their own nations.
- The American Radio Relay League, the organization of U.S. hams, fought to the last to preserve the code requirement.



# We Won

- Eventually, international ham organizations voted to remove the treaty requirement, and asked ITU to do so. Through continued campaigning, we were eventually able to get FCC to follow, and most other nations did so as well.

# Change in Number of Hams After Code Test Ended in 2007



# Did We Save Ham Radio?

- The trend, before the Morse code test was ended in 2007, indicated that Amateur Radio would have ended during our lifetime.
- This Year, there are more U.S. hams than at any time in history. The trend may be leveling off at about  $\frac{3}{4}$  Million.

# Codec2 and Amateur Digital Voice

- The Japanese Amateur Radio League designed the first practical VHF/UHF digital voice system for Amateur Radio, with funding from the Japanese Government, releasing it in 2001. They did it a lot differently than it would be done today, because they didn't believe in the capabilities Open Source if they had even heard of it.

# D-STAR

- Dependent on the AMBE codec, a proprietary product of DVSI. 4800 bits per second. Patented.
- Protocol (not codec) published as an Open Standard.
- While originally closed, DVSI published a specification of AMBE in order to be accepted as a component of APCO Project 25. The specification is deliberately obfuscated, for example it's implemented as a large discrete FFT (which nobody would really do). It's a paper spec with no reference implementation.

# Status of AMBE Patents in D-STAR

- We believe that the implementation of AMBE originally specified for D-STAR, AMBE 1000, is out of patent and/or the patents can no longer be enforced due to previous disclosures.

# Creating an Open System

- I wanted to replace AMBE in D-STAR with a fully open codec which hams could tweak. Speex existed at the time, but was too high bandwidth.
- For some things, you really do need a rocket scientist. Fortunately, you can find one.
- Jean-Marc Valin, author of Speex, introduced me to David Rowe VK5DGR. He had previously written the Open Source Line Echo Canceller and embedded hardware for the Asterisk PBX.
- David has a Ph.D. in digital voice coding. I don't know if the world has more than one of those. Rowe had published a codec as part of his 1997 thesis, and this became the basis of Codec2. he's done a lot of new development since then.

# Don't Let This Happen To Your Thesis!

- Having designed a digital voice codec that was probably revolutionary in 1997, but might have been too slow for the embedded computers of the time, Rowe's thesis sat on the shelf for 15 years.
- His work was cited in patents by the makers of proprietary codecs. But a thesis with good work and existing software sat in public view for 15 years without a user other than those proprietary companies. This is what we do to our best and brightest.
- Moral: Issue Open Source software with your thesis.



# Codec2 and Modem Progress

- Today, Codec2 provides a range of Open Source codecs from 3K bits per second down to 700 bits per second.
- Brady O'Brien KC9TPA and Rowe have developed a family of software modems to work with it.
- Besides working on VHF/UHF, one goal driving development is to exceed the capabilities of SSB for HF DX communication. We are seeing the first cases of that.

# Commercial Ham and Two-Way Radio Digital Voice Implementations All Stink

- These are D-STAR, DMR, Yaesu System Fusion, etc.
- All are built to work on top of analog FM modulators and demodulators, and analog radio platforms that have linearity and equalization issues.
- *System Fusion* uses 4FSK (“C4FM”) with tones that aren’t orthogonal.
- An optimal FSK modem implemented using an SDR transceiver can achieve *at least* 8 dB improvement over these implementations.
- In power terms, a 1 Watt radio would perform as well as a 6.3 Watt one.

# The Next Step

- Now that we have all of this software, we need mobile, base, and HT radios to run it.
- They must be programmable SDR transceivers.
- They should have the capability to run apps, either on the radio platform, or on a connected phone.
- They must be built with the assumption that they will not run the same codecs, modems, and apps over their entire useful lifetime.

# The Next Step

- Now that we have all of this software, we need mobile, base, and HT radios to run it.
- They must be programmable SDR transceivers.
- They should have the capability to run apps, either on the radio platform, or on a connected phone.
- They must be built with the assumption that they will not run the same codecs, modems, and apps over their entire useful lifetime.

# Next Step, First Try

- Chris Testa and I tried to build a power-efficient SDR HT with a radio based on the CMX991 and a computer with built-in FLASH-based gate-array based on Microsemi *SmartFusion*.
- Chris and I spent a lot of time making the computer. By the time we were done, we could buy better, faster computers, already built, for less. We won't make computers again.
- I bought a lot of test equipment at surplus, so we each have a pretty good lab.

# Biggest Mistake

- Chris and I got the computer working before we entirely debugged the radio. In retrospect this was backwards, and we should have built a radio that we could debug without building a computer at all.
- To make up for the computer's low speed, we took too long working on gate-array code.
- The radio design turned out to be too noisy, and that killed the design. By the time we got to that point, there were a lot better platforms than CMX991.

# Why Not Use Raspberry Pi 3?

- Many small, powerful, and really cheap computers, like Raspberry Pi 3, are too I/O limited to do high-bandwidth SDR. In the case of Pi 3, its USB 2 is too slow, and it has serial channels dedicated to a camera and display that might have worked, except that they aren't fully documented and depend upon undocumented coprocessors.
- But there are somewhat more expensive boards with USB3, etc.

# Next Steps

- Chris and I took a two-year break to work on other things after this design failure. In that time, nobody has approached creating the radio we wanted. So, it is probably time to work on the next version.
- This would use an existing computer, existing SDR board, and only require the production of hardware for filters, amplifiers, and glue.
- I have the development hardware on hand.



# How Can I Make Money By Doing Open Source?

- If you must answer how you can make money by doing an Open Source project, you're probably the wrong person to lead the project.
- Most successful Open Source projects are operated by people who need the software to do their work, but make money in some other way.

# Much Open Source Income is Serendipitous

- Example: Eric Blossom never expected for Bob Mcgwyer to come along with money from the U.S. government.
- I never expected that Busybox would get me involved in a compliance business.
- The key seems to be to develop an Open Source project and a personal capability, and then people with money who want that come along.

# Marketing Has No Crystal Ball

- Marketing departments can't really chart the future of a company. If marketers could do this, they'd be at home investing in stocks.
- Customers can't tell you what they'll need in the future. They don't know what that is. Thus, customer-driven companies can't lead.
- The strategic marketing paradigm of Open Source is a massively-parallel drunkard's walk filtered by a Darwinistic process. It works as well as strategic marketing at big corporations.

# Open Source Will Continue To Be Important

- This is the only way we know to make big things that does not take a huge investment to start.
- The development power available to Open Source is very much larger than that available to any start-up, and most developed companies.
- There are a lot of desirable things that just can't be justified financially at the start. Like the web.

# Lesson of the Web

- Ted Nelson created the Xanadu project to build the web, and had a very well designed paradigm for everyone to make money from web texts. Working from 1960 to 1998, with a 10 Million dollar investment from Autodesk in the '80's, he failed to build the web.
- Tim Berners-Lee just made basic web software as Open Source, and let people figure out by themselves how to make money from it.

# Lawsuit

- You may have heard that I am being sued by “Grsecurity Inc.” (Brad Spengler) for publishing an opinion about their GPL compliance and that of their customers.
- Without discussing the specifics of my own case, I am concerned about *your* freedom to publish and discuss opinions about the GPL and any entity’s compliance with the GPL – especially when it concerns your own software or that of a group you support. Silencing such speech would deter understanding of the GPL as well as enforcement.
- Although it is not here yet, the day could come that I ask for help.