Open Source Takes Shape

Since all early software was assumed to be open source—open to improvement and reuse—and the technology of the time largely required that software be distributed in a form that allowed those uses, there was little need or drive to formalize the concept.[1]

Eventually, however, as software matured, the opportunity to charge for it emerged.

Unix was born in AT&T's Bell Labs in 1969, but it was raised at UC Berkeley, where students and faculty added many of the features we now recognize as essential to the operating system, including virtual memory and almost every aspect of modern networking.[2] Berkeley's enhancements were distributed widely as a package called BSD (Berkeley software distribution), and over time they became as important to a Unix system as the licensed and purchased product from AT&T.[3]

But for all their work, Unix was still owned by AT&T, not Berkeley, and the complexities (and legal ambiguities) of that relationship highlighted the difficulty of working with and depending on software that that was not "free." Not free in the economic sense, but in the legal sense.[4]

At about the same time, Bill Gates (yes, that Bill Gates) wrote a letter published in the January 31, 1976, issue of the Homebrew Computer Club newsletter describing how important it was to the future of computing that hobbyists support the development of a commercial software industry:

> To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?[5]

Much of the letter, however, attacks the community of sharing that was prevalent at the time:

> As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?[6]

Commercialization of software might be seen as a natural evolutionary landmark, but it also reflected a profound philosophical shift: software that had been easy to copy, learn from, fix, and share was being restricted to fit the economics of scarcity.

The commercialization of software was happening, not in a graceful leap, but in lurched motions, and Richard M. Stallman was among the first to recognize it.

Stallman was a software developer in MIT's Artificial Intelligence Lab when he was faced with a dilemma: the systems and servers their software was built on had been eclipsed, and new systems depended on commercial software. Stallman and others in the lab would be required to sign nondisclosure agreements that would prevent them from sharing not only any improvements they made, but also anything they learned about the systems.[7]

Stallman identified that, as software became more important in our lives, a user's freedom to choose how to use, modify, and improve it became more important.[8]

Rather than join the world of proprietary software, Stallman chose instead to resign from MIT to form the Free Software Foundation (FSF) to develop an operating system and related software and promote the following four essential software freedoms.[9]

- The freedom to run the program, for any purpose (freedom 0).

- The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor (freedom 2).
- The freedom to improve the program, and release your improvements to the public, so that the whole community benefits (freedom 3). Access to the source code is a precondition for this.[10]

Among the FSF's successes are the development of a collection of applications that helped make Linux a complete operating system, and most significant, the development of the GNU General Public License (GPL). The GPL legally codified Stallman's four essential freedoms and allowed those who applied the license to their software to trust that their work would be contributing to the community.

As open-source software evolved and matured, the economic benefits became clearer. VA Linux founder Larry Augustin believes that open-source software became a real option just as the proprietary software market was reaching a crisis where the cost of market exceeded the cost of development.

> My favorite is Salesforce.com. In 1995 they spent under $10 million in R&D and over $100 million in sales and marketing. That doesn't work.
>
> Open source enables people to reach all those customers. It's a distribution model. The people who create great software can now reach the rest of the world.[11]

And what was the essential component? What made it all happen? Augustin pointed to Richard Stallman and the GNU General Public License.

It's hard to overestimate Stallman's influence on computing. Speaking on how history will view him, biographer Sam Williams remarked:

> Nobody but him could have had the patience, and the stubbornness, and the will to build something this big. There are other people writing free software, but he's the one that made it an issue. He's the one that provided the initial gravitation that everybody else could gather around.[12]

Notes

1. Richard Stallman, "The GNU Project" (originally published in the book *Open Sources*), on the GNU Operating System Web site, www.gnu.org/gnu/thegnuproject.html (accessed Mar. 19, 2007).
2. Dennis M. Ritchie, "The Evolution of the Unix Time-sharing System," on the archival Web site for Bell Labs computing sciences research, http://cm.bell-labs.com/cm/cs/who/dmr/hist.html (accessed Mar. 19, 2007; paper first presented at Language Design and Programming Methodology conference at Sydney, Australia, Sept. 1979; conference proceedings published in *Computer Science #79: Language Design and Programming Methodology,* Springer-Verlag, 1980; this rendition based on version reprinted in *AT&T Bell Laboratories Technical Journal* 63, no. 6, part 2 [Oct. 1984]: 1577–93).
3. David Pescovitz, "Berkeley Unix and the Birth of Open-Source Software," in *Lab Notes* 1, no. 2 (Oct. 2001), on the Web site of the College of Engineering, University of California, Berkeley, www.coe.berkeley.edu/labnotes/history_unix.html (accessed Mar. 19, 2007); "Berkeley Software Distribution," Wikipedia, http://en.wikipedia.org/wiki/Berkeley_Software_Distribution (accessed Mar. 19, 2007).
4. Andrew Leonard, "BSD Unix: Power to the People, From the Code," *The Free Software Project,* chapter 2, part 1, on the Salon.com Web site, http://dir.salon.com/story/tech/fsp/2000/05/16/chapter_2_part_one/index.html (accessed Mar. 19, 2007); Thor Olavsrud, "Novell Challenges SCO Over Unix, Linux," May 28, 2003, in "DevX News," on the DevX Web site, www.devxnews.com/article.php/2213031 (accessed Mar. 19, 2007).
5. William Henry Gates III, "An Open Letter to Hobbyists," *Homebrew Computer Club Newsletter* 2, no. 1 (Jan. 31, 1976), on the DigiBarn Computer Museum Web site, www.digibarn.com/collections/newsletters/homebrew/V2_01/gatesletter.html (accessed Mar. 19, 2007).
6. Ibid.
7. Stallman, "The GNU Project."
8. Richard Stallman, "The Free Software Definition," Free Software Foundation Web site, www.fsf.org/licensing/essays/free-sw.html (accessed Mar. 19, 2007).
9. Stallman, "The GNU Project."
10. Stallman, "The Free Software Definition."
11. Larry Augustin, quoted in Dana Blankenhorn, "Augustin Still Believes in Open Source Values," June 9, 2006, ZD Net Web site, http://blogs.zdnet.com/open-source/?p=675 (accessed Mar. 19, 2007).
12. Sam Williams, quoted in Bruce Stewart, "How Will History View Richard Stallman? An Interview wth Sam Williams," Feb. 28, 2002, ONLamp.com Web site, www.onlamp.com/pub/a/onlamp/2002/02/28/williams.html (accessed Mar. 19, 2007).