



## Open Season

### **Why an industry of cutthroat competition is suddenly deciding good karma is great business.**

*By Andrew Leonard*

The standing-room-only crowd at the Stanford University Faculty Club is getting antsy. For the second or third time, the moderator has skipped over Linus Torvalds, preferring to address his questions about open-source business models to other members of the panel. The audience is interested in what the other participants have to say - they are, after all, the up-and-coming stars of open source - but not at the expense of a chance to hear Linus, the man himself, creator of the kernel of the Linux operating system, declaim on the future.

The audience is split between scruffy fanboys eager to cheer every anti-Microsoft riposte and nattily dressed Stanford students itching to break out of academia and into a high-flying start-up. It is not the kind of audience that will sit idly by as its desires are ignored. But before the grumbling can erupt into a full-scale riot, Torvalds interrupts the moderator. "I've got a business model too," he says with an impish grin. "After this talk, I'm going to pass my hat around."

Wild laughter ensues. Linus Torvalds is at that stage of fame and popularity at which every joke he tells is considered hysterical and every pronouncement he delivers is treated as profound. But there's a nugget of real humor to be mined from his quip. The concept of a business model for open-source software is a contradictory feat of legerdemain. The term *open source*, after all, refers to a model of software development in which the underlying code to a program - the source code, aka the "crown jewels" - is by definition made freely available to the general public for modification, alteration, and endless redistribution. No wonder the venture capitalists are confused - where's the profit in *free* software?

The answer lies somewhere at the intersection between bold new businesses delivering startlingly good software to market and the evolving economy of code. At the Faculty Club, it's the titillating possibilities of the former - in addition to Torvalds' presence - that has packed the hall. The panel is stocked with articulate representatives of the still fledgling, but quite vibrant, open-source start-up community. A programmer from Red Hat Software, the North Carolina-based company that is the leading packager of commercial Linux distributions, sits next to the CEO of Sendmail Inc., a start-up aiming to sell commercial "enhancements" of a venerable open-source program that still, after 18 years, moves the bulk of the Internet's email.

VA Research, which sells computers preloaded with Linux - and has received a dose of venture capital from Sequoia Capital, the same company that steered Yahoo! to its current \$36 billion market valuation - is present, as is Cygnus Software, an open-source services and support company that reports raking in revenue of \$20 million in 1998. Piddling by Microsoft's standards,

but for open-source business fans, a sure sign of many billions yet to come. These companies and many more are ferreting out niche markets that exploit the robust, high-quality software built collectively by the faithful. The code might be free, but such side benefits as tech support, packaging, installation software, regular upgrades, Linux-based office applications, and Linux-friendly hardware are not. There's money to be made in open-source software - you just need to know where to look.

The Faculty Club panel even includes a token venture capitalist, who makes gruff remarks about not wanting to see business plans with the words *Marx* or *manifesto* in them - proof that despite the steadily improving profile of open-source software in the business community, the movement still hasn't shrugged off all the messy connotations of that most dangerous of words: *free*.

By spring of 1999, open-source mania has transfixed the computing industry. Gold rush fever, never hard to unleash in today's overheated high tech economy, is on the rise again. IBM has endorsed open-source Web-server software called Apache and has committed to providing support for Linux. Intel is investing in Red Hat, Corel has a Linux version of WordPerfect, and scores of other tech stalwarts are announcing some level of support for open-source systems. Hardly a day goes by without another company declaring that it is now "going open source."

Never mind that some of these open-source-come-latelies may be trying to cover up for some misbegotten product that would never have had a prayer in the marketplace, or that they may well be aiming to exploit open-source resources without giving anything back in return. Those two little words - *open source* - have become a magical incantation, like *portal* in 1998 or *push* in 1997. Just whisper them and all will be yours: media attention, consumer interest, and, of course, venture capital.

But is there really a jackpot waiting at the end of the open-source rainbow? Or is that even the correct question to ask? As the high tech competitive landscape becomes ever more brutal - as product cycles speed up, consumers become more discerning, and international pressure intensifies - the tactic of going open source may best be envisioned as a necessity, not a business plan. A necessity because you must go open source to get access to the benefits of the open-source development community - the near-instantaneous bug fixes, the distributed intellectual human resources of the Net, the increasingly large open-source code base. As open-source momentum continues to grow, the code itself will become a commodity - available to all, for next-to-nothing prices, like water or wheat.

In this world to come, open-source software may be better thought of as a tool for stripping down company operations to lean, mean levels of efficiency than as a revenue generator in and of itself. Certainly, start-ups like Red Hat and VA Research may be dreaming of (and planning for) their open-source IPOs, but on a grander scale, larger, more established businesses will focus instead on discovering what their enterprises can best outsource to open source. What can the open-source development community take care of better than my company can do on its own? And what do I do best?

There's never been a more cost-effective way to contract out services than to use this type of free labor (minus one's own contributions to the general welfare). And it never could have happened without the fertile conjunction of the Internet with the particular form of cooperative behavior exemplified by open-source software.

This should come as no surprise. Hackers working in the tradition of free software built the Internet. The Net, in turn, encourages and facilitates collaboration on ever more complex projects.

It's a bootstrapping feedback loop that shows no signs of weakening. If anything, it is gaining speed, fueled by both the pragmatic benefits of the system and a growing cultural consciousness that sharing information works.

Imagine the open-source development environment as a vast, open-to-all-comers laboratory stocked with every imaginable tool, including an infinitely large library and seamless communication conduits to talented colleagues all over the world. There is no entrance fee to this laboratory, no need to pay exorbitant license charges simply to use a particular tool, and no limit to the possibilities. All you need to do is find a likely niche and take advantage of the resources of the laboratory to leverage your creativity and talent - or to rev up the throttle on your already established business.

Take, for instance, what might happen if a particular tool doesn't exist and no single individual or small company has the time or resources to create it. In a dynamic open-source environment, alliances of enterprises and individuals with similar needs can easily coalesce around open-source project nodes to collectively address any given problem.

The example offered by the Apache project's work on Web servers underscores the validity of this model. The Apache Group includes several programmers who work for companies - Covalent Technologies, the Internet Movie Database, Organic Online, and IBM - that for one reason or another find it useful to have access to a cheap and powerful program that transforms ordinary computers into workstations capable of handling immensely stressful loads of Web traffic. These companies are not in the business of actually developing or selling Web servers - they are in the business of using them. Management is enlightened enough to encourage employees to devote time and energy to the Apache project - for free - because they understand the long-range benefits that will accrue.

Apache works. It's the Web-server program of choice for more than half of all publicly accessible Web servers - a statistic that is all the more amazing because Web servers are a market niche laser-targeted by both Microsoft and Netscape. The Apache project has proven that a rock-solid, commercial-quality software program can be created collectively by taking advantage of the flexibility of the open-source development model.

The software industry built on these principles may well become a dizzying blur of kaleidoscopic frenzy, populated by thousands of hustling enterprises endlessly reforming and recombining into new Apache Group-like networks and fresh alliances. And all the while, the common body of available open-source code will grow steadily, which in turn will encourage more businesses to take advantage of it.

Of course, potential investors have every right to be skeptical of open-source-oriented business plans. And those who echo Linus Torvalds' joking "world domination" call to arms may do themselves more harm than good through their unbridled optimism. Open source is no panacea, and it has some very obvious shortcomings. Most open-source software, for instance, is written by hackers aching to "scratch their itch" - to solve some particular problem unique to their own circumstances. It is far from clear that itch-scratching will lead to user-friendly interfaces that can match the sophistication of Microsoft Windows or the Macintosh OS. Will an open-source project produce an interactive help system useful to computer illiterates? No one really knows.

Nor does anyone have a good answer for how a small open-source company with limited resources will defend itself against such treacherous pitfalls as patent and product liability suits. Or what happens if, for example, Linus Torvalds wearies of overseeing the Linux development process. Who

will make the ultimate decision on what software patch is incorporated into the all-important Linux kernel? And how exactly will the open-source "community" enforce the "give back what you create" open-source licensing terms?

**There's never been a more cost-effective way to contract out services than to use this type of free labor.**

Moreover, there are plenty of situations in which releasing source code plainly doesn't make good economic sense. In the case of extremely high-value-added programming projects, such as advanced digital-imaging software employed in life-or-death medical procedures, keeping control of one's intellectual property can be enormously lucrative.

But questions such as whether Linux can wrest control of the consumer desktop from Microsoft, or whether Joe User has any interest in mucking about with source code, miss the deeper, essential revelation delivered by the open-source phenomenon - the decentralization of talent and of economic power. Code as commodity, widespread access to tools and infrastructure, institutionalized collaboration via the Net - all these factors allow the individual or enterprise to shed unnecessary baggage that drags against entrepreneurial acceleration.

The combination of good software and messianic faith - the Internet's fabled "gift economy" - that spawned the technocultural phenomenon of free software in the '80s has today morphed into both a business strategy and a way of life. In the mid-'90s, the explosion of commerce and mainstream attention that overwhelmed the Net obscured the workings of the gift economy; some observers even pronounced it dead - a nice experiment while it lasted, but not the kind of robust business model that could scale up to take into account the needs of hundreds of millions of users. But open source is now giving the skeptics pause. Not only are open-source programs still the bedrock of the Internet's infrastructure, they are also integral to the operation of huge Web sites like Yahoo! and Amazon, and are increasingly popular for deployment in mission-critical corporate computing tasks.

The emergence of the open-source Apache Web server program as the market leader in April 1996, Netscape's decision in early 1998 to release the source code to the Navigator Web browser, and, all along, the steadily rising profile of Linux have sent shock waves rippling through the commercial-software marketplace. The gift economy suddenly appears to be stronger than ever.

If Linus Torvalds had, after all, passed around his hat at the Stanford Faculty Club that evening, it would no doubt have returned filled with cash - or, even more likely, with multiple job offers. He needs neither, of course - that's one reason the joke was funny. But he would have been rewarded anyway, because what he has contributed for free has such immense value. He, and all the rest of the open-source hackers, have proven that software entrepreneurs don't have to spend their lives guarding the crown jewels against the depredations of interlopers. They can give their treasures away, and succeed.

---

*Andrew Leonard ([aleonard@salonmagazine.com](mailto:aleonard@salonmagazine.com)) is the senior technology correspondent at Salon Magazine. He is working on a book about free software.*

[Copyright](#) © 1993-2004 The Condé Nast Publications Inc. All rights reserved.

[Copyright](#) © 1994-2003 Wired Digital, Inc. All rights reserved.