The Microsoft law suit, software development and the capitalist market

Mike Ingram
2 May 2000

The US government proposal to break up Microsoft, dividing its operating system (Windows, NT etc.) from the Office suite (Word, Excel etc.), raises fundamental questions concerning the development of computer software and its relationship to the capitalist market.

Microsoft presently controls over 80 percent of the operating system market and 90 percent of the market in business applications. The company denies having acted anti-competitively, arguing that Microsoft's dominance of the software market has led to both standardisation and relatively low-cost software for the consumer.

Microsoft's competitors such as Netscape, Sun Microsystems and others claim that in blocking competition, Microsoft effectively stifles innovation. Both sides assert that the market is the source of technical innovation. How does this latter claim measure up when viewed against the current situation in the computer software market?

At the very least it can be said that Microsoft's unrivalled dominance of the market leads to the release of software that is less than perfect. A recent example is the company's new flagship product, Windows 2000. It is reported that the new operating system contains over 60,000 bugs. This averages out to 12 bugs for each of the 5,000 programmers who worked on the package.

Microsoft Windows, in all its flavours (3.1, 95, 98, 2000 and NT), is not so much the operating system (OS) of choice. Rather, it is perceived to be the only game in town. Few who have used an Apple Macintosh computer, for instance, would argue that the Microsoft OS is superior. The problem is that Apple Mac users make up fewer than 10 percent of computer users, and these are to be found mainly in the fields of education and the graphic arts. The small user base leads to a dearth of available applications and expensive hardware.

From the standpoint of the capitalist market, it can be argued that Microsoft boss Bill Gates has done nothing other than behave as a good capitalist should. He has used the dominant market share established by the Windows operating system to make advances into Internet technology, integrating the web browser into the operating system and signing deals with hardware manufacturers to bundle Microsoft products rather than those of competitors. There are no indications that Microsoft's competitors would behave any differently if the situation were reversed.

The problem with much of the commentary surrounding the Microsoft case is that it takes the present forms of software development and distribution as givens. In much the same way as the capitalist market is worshipped as the only possible vehicle for the organisation of economic life, so too the development of proprietary software[1] has been presented as the only possible variant.

Even more significant than any action finally decided on by the courts is what the Microsoft case says about the relationship between the market and technical innovation. From this standpoint, an interesting footnote to the Microsoft case has been the increasing attention being paid to various open source software[2] developments, and the Linux operating system in particular.

Most commercial software is distributed only in machine code—ones and zeroes understood only by the computer hardware. By contrast, open-source programs always include the source code, written in a programming language such as C++ or Java, which any programmer can understand.

Programmers are free to change the code in order to fix bugs, or even modify it and incorporate it into new programs. Different licenses give varying amounts of freedom as to what can and cannot be done with open source software. While some licenses allow projects to be incorporated into proprietary systems that are not then released as open source, the GNU[3] Project prohibits this, insisting that any software that incorporates parts of open source applications must be released under the same license with full source code made available.

In the wake of the spectacular fall in the share values of so-called dot.com companies, articles have begun to appear in the financial and technology press drawing attention to the existence of open-source software development projects.

For example, the *New York Times* of April 20 ran an article in its business section which said, "When technology stocks took their sharp tumble last week, many companies appeared to lose one of their most important assets—the ability to lure talented employees with options. To attract and hold the best, you have to offer the chance to strike it rich."

The *Times* then asks, "Or do you? What are we to think when the best of the best—the elite programmers that industry wisdom deems 100 times more productive than the typical competent coder—donate their precious time to develop software anyone can use without charge?"

Unable to comprehend anyone doing anything without personal financial gain, the *Times* article argues that the motivation of the programmers involved is name recognition. "Contributing to open-source projects can enhance a programmer's career prospects," the paper says.

The *Times* continues: "The most striking example of this is Mr. Torvalds, who was a graduate student in Finland when he released Linux and is now a highly paid executive at a Silicon Valley start-up."

Linux is a freely distributed operating system that has been around since 1991. It now receives regular press coverage as a result of commercial distribution companies such as Red Hat being floated on the stock markets. With the development of "windows like" graphical front ends such as KDE and Gnome, Linux has grown in popularity among business and home users alike.

International Data Corporation estimates that Linux has anywhere from 7 to 21 million users world-wide, and a 200 percent annual growth rate. Observers believe it represents a leading potential challenger to Microsoft Windows in the personal computer operating systems market. Some commentators believe that part of the government's reasoning for breaking up Microsoft is that a new company responsible for the Office applications would develop a version for Linux, thus increasing competition in the sphere of the operating system itself.

Press coverage of Linux and open-source software projects in general is

invariably tainted with an air of astonishment, reporting this as a new and inexplicable development. An examination of the history of the development of computer technology reveals, however, that far from open-source development being a digression from the norm, it is the way in which most of the more significant technological advances were made.

In the late 1960s and the 1970s it was common practice for programmers to share the products of their labour with no restrictions. At that time companies and individuals were more interested in the development of the technology as a whole than safeguarding "trade secrets".

Not only academic institutions such as the Berkeley campus of the University of California and the Massachusetts Institute of Technology (MIT), but also commercial research centres such as Bell Labs and Xerox's Palo Alto Research Center (PARC) operated an open policy in which computer source code was freely exchanged between organisations.

Many of the original ideas for the desktop computer popularised by Apple and Microsoft were first developed by scientists working at the PARC labs. These included the first desktop computer, the first commercial mouse, Ethernet networking technology and many other innovations.

The early efforts in the 1970s focused on the development of an operating system that could run on multiple computer platforms. The most successful of these was the Unix operating system and the C language used for developing Unix applications. These were originally developed at AT&T's Bell Laboratories. The software was installed across institutions, being transferred either freely or for only a nominal charge. Sites where the software was installed made further innovations and these were in turn shared with others.

The process of collaborative software development and the sharing of source code were greatly enhanced with the development of the Internet and, specifically, the creation of Usenet in 1979. Usenet was a computer network used to link together the Unix programming community. In 1979 there were just three Usenet sites, but by 1982 this had grown to over 400. The ability of programmers in both corporate and academic settings to rapidly share technologies led to huge technical advances.

In these early years cooperation was of a highly informal character. There was, in fact, no real effort to delineate property rights or restrict the use of software until the early 1980s. With the rapid growth of the commercial use of computer systems, AT&T began laying claim to intellectual property rights relating to Unix, despite the fact that hundreds of programmers at other institutions had contributed to its development.

Only with the advent of the personal computer as a mass consumer item in the early 1980s did the idea of closed proprietary systems come to be considered the corporate norm.

As operating systems produced by Apple Macintosh and Microsoft began to establish a mass user base, a parallel development took place to maintain the open source structure for software development. In the form of the GNU Project and the Free Software Foundation established by Richard Stallman in 1982, this took on a directly political and ideological character.

Stallman worked at the Artificial Intelligence Lab of MIT between 1971 and 1983. The AI lab used a timesharing operating system called ITS (the Incompatible Timesharing System) developed by the lab's staff programmers. Stallman explains that while the term "free software" was not yet invented, this is what the system was. "Whenever people from another university or a company wanted to port and use a program, we gladly let them. If you saw someone using an unfamiliar and interesting program, you could ask to see the source code, so that you could read it, change it, or cannibalise parts of it to make a new program," he says.

In a paper entitled *The GNU Project*, Stallman writes, "The idea that the proprietary software social system—the system that says you are not allowed to share or change software—is antisocial, that it is unethical, that

it is simply wrong may come as a surprise to some readers. But what else could we say about a system based on dividing the public and keeping users helpless? Readers who find the idea surprising may have taken [the] proprietary social system as given, or judged it on the terms suggested by the proprietary software businesses. Software publishers have worked long and hard to convince people that there is only one way to look at the issue."

On the issue of software companies having "an unquestionable right to own software and thus have power over all its users", Stallman claims, somewhat naively, that the US Constitution and legal tradition reject this view. "Copyright is not a natural right, but an artificial government-imposed monopoly that limits the users' natural right to copy," he writes. Stallman opposes the conception that "the only important thing about software is what jobs it allows you to do—that we computer users should not care what kind of society we are allowed to have."

The GNU project progressed steadily over the next few years with volunteer programmers working on different parts of the system, making them ready to run with Unix. The only thing missing was the guts of the system, the kernel. In 1991 the GNU kernel that had been developed was abandoned in favour of another, the Linux kernel developed as a hobby by a young student, Linus Torvalds, at the University of Helsinki in Finland.

The merger of the GNU project with Linux to produce GNU/Linux brought the work of these early volunteer programmers to fruition, and in so doing provided powerful evidence of the technical superiority of the open-source method of working.

Taking the small Unix system Minix as his point of departure, Torvalds had developed a system that exceeded the Minix standards. His work began in 1991 when he released version 0.02. It progressed steadily until 1994 when version 1.0 of the Linux Kernel was released. The current version of the kernel is 2.3.

Torvalds' kernel and the work of the GNU project came together to make a fully-fledged system free of the controls of proprietary software companies. Today there are dozens of commercially available distributions of Linux containing thousands of utilities and applications available for a fraction of the cost of either the Microsoft or Apple operating systems.

Linux is arguably the most stable, widely supported, flexible and powerful operating system available today. It runs on a variety of computer hardware including Intel clones and Apple Macintosh computers. Distributions come complete with the free Apache web server, which is used on over 55 percent of public web sites on the Internet.

The success of Linux lies precisely in its openness. Users of the operating system are themselves developers. If the software does not perform a function adequately, someone, somewhere will fix it. This fix will be made publicly available and incorporated speedily into the official release of the Linux Kernel. Even non-technical users can take advantage of the open-source system. There are literally thousands of discussion forums within the Internet where users can get answers to questions regarding the use of the operating system and possible errors that may occur.

Contrast this with the attitude of Microsoft CEO Bill Gates. In an "open letter to Hobbyists" written in 1976 Gates regaled against amateur computer users or "hobbyists", accusing them of "theft":

"One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3 man-years into programming, finding all bugs, documenting his product and distributing it for free? The fact is, no one besides us has invested a lot of money in hobby software ... but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft" (Published in the Homebrew Computer Club newsletter on February 3, 1976).

The issue of open source versus proprietary systems goes beyond that of

personal ethics and choice. Whatever the benefits of the Linux operating system, it has until now been largely the province of skilled users. One of the reasons for this is lack of support for computer peripherals such as graphics cards, printers, scanners, etc. This is not due to a shortage of skilled people to write the drivers necessary to interface this equipment with the operating system. The problem is the reluctance of some hardware manufacturers to release the information necessary to facilitate such a development.

In the last few years this has begun to change, with companies such as Compaq and Hewlett Packard selling computers with Linux installed.

A number of leading companies have also released their software as open source. Most notably among these is Apple. In unveiling the Mac OS X Server, CEO Steve Jobs surprised an audience of reporters and software developers last year by announcing that the software would be put into open source. On April 11 Intel Corporation announced that its Common Data Security Architecture (CDSA) software would also be released as open source.

There are definite economic antagonisms at work in the campaign against Microsoft. Ultimately, the government's proposal to break up the software giant arises from a ferocious struggle between rival corporate powers for control of markets and profits.

But the case against Microsoft also reflects a growing recognition that the speed of technological change and the demand for new and better systems requires a technical leap that is being stifled by Microsoft's continued dominance.

The emergence of the Internet as a mass medium, which itself largely conforms to open standards, demands far more flexibility in the next generation of computer software. With new devices such as mobile phones and wireless applications emerging at a fantastic pace, manufacturers are demanding software that can be modified and extended, i.e., they demand access to the source code.

The open-source environment stands in such a stark contrast to the established practice of Microsoft that powerful sections of business, ultimately backed by the government, concluded that it was necessary to bring Microsoft into line. How far this will be pursued by the government is by no means certain, not least because Microsoft has a bigger financial war chest than the US Justice Department.

It is impossible at this point to predict the final outcome of the lawsuit against Microsoft. Ultimately however, the conflict is not the product of the subjective avarice of Bill Gates, on the one hand, or some new-found democratic impulse on the part of the US Justice Department, on the other. At a basic level it is an expression of increasing tensions between the development of technology, especially the Internet, and the subordination of this technology to the capitalist market and the system of private property upon which it is based.

The fundamental issues raised in the Microsoft case have a great significance for masses of working people. What has been the effect on the working class of the control of the new technology by big business?

Phenomenal technological advances have done nothing to eliminate poverty for masses of working people in America and elsewhere. Far from reducing social inequality, the technological advances, under the control of huge corporations, have been used to further enrich the privileged few at the expense of the majority of society. No actions taken by the US government and Microsoft's corporate rivals will do anything to change this situation.

It is worth noting that the case against Microsoft was instigated by Netscape corporation, which is now owned by another of the main instigators, America Online. While, for their own reasons, they raise the negative impact of Microsoft's monopoly in the software market, we have yet to hear similar noises in relation to the recent merger of AOL and Time Warner, which represents the most direct attempt by business to monopolise the Internet.

Monopolisation is inherent to the capitalist system itself. The struggle against it requires a political struggle against very real class interests, which are represented by a system in which production as a whole is organised not for social need, but private profit.

Explanatory Notes

1. Here we use the term *proprietary software* to describe the closed character of computer platforms such as Microsoft. Under the proprietary system, both the computer code and the details needed to develop applications to run under Microsoft operating systems are closely guarded secrets, released only to companies which have signed deals with Microsoft. (return to text)

2. The term *open source software* refers to computer applications and operating systems in which the code as written by the programmer is made available alongside the distributed software. Written in a language understood by any computer programmer, the application can then be modified or fixed, without the purchaser being dependent upon the company that developed it. (return to text)

3. GNU stands for "G'NU not Unix". It is the name give to the system developed by a group of volunteer programmers led by Richard Stallman. His GNU Project is most famous for its association with the Linux operating system. (return to text)

To contact the WSWS and the Socialist Equality Party visit:

wsws.org/contact