SOOWOOK KIM

CAPITORGS AND FREE/LIBRE AND OPEN SOURCE SOFTWARE (FLOSS): TOWARD CRITICAL TECHNOLOGICAL LITERACY AND FREE/LIBRE AND OPEN SOURCE SOCIETY (FLOSS)

Major economic and cultural resources such as oil, agricultural products, and digital technologies are under the control of global multinational corporations. The world under neoliberal capitalism witnesses "global economic apartheid," "the existence of widespread poverty on the one hand and extraordinary wealth on the other in marketised and supposedly 'civilised' societies" (Darder, 2002, p. 10; Roberts, 1999, p. 101). High rates of "poverty, social injustice, economic instability, environmental destruction, and war" are the result (Callinicos, 2003, p. 136). Neoliberal perspectives on human beings reflect the "antisolidarity nature" of contemporary capitalism (Freire, 1997a, p. 88). Neoliberal human beings represent "homo economicus, individualism and self-interest" and "unconnected individualism, greed, and profit" (Apple, 1990, p. xv; Peters, 1999, p. 119). Educational discourses of neoliberalism, promoting literacy for job opportunities, economic advancement, and individual success are of paramount importance to produce *human capital* rather than *human beings* (Apple, 1990, p. xiv; Freire, 1997b, pp. 314-315; Graff, 1995, pp. 51, 55; McLaren, 2000, p. 169; New London Group, 1996; Rikowski, 2002, p. 111; Williams, 1989, p. 14). A hidden agenda of neoliberal literacy includes training students and workers to accept "a new work discipline" and conditioning their will to maximize the accumulation of capital and wealth (Graff, 1995, p. 54).

As students increase their marketability, they "are always already shaped by the labyrinthine circuits of capitalist desire" (McLaren & Farahmandpur, 2002, p. 241). Our life is constrained by the social relations capital coordinates (Adamson, 2002, p. 1; Rikowski & McLaren, 2002, p. 4). We are not just learning, teaching, and living in neoliberal capitalist societies, but are becoming "a new life-form: human-capital" through "the capitalization of humanity" (Rikowski, 2002, p. 111). Is there is an unfettered capitorg (capitalist organism) in all of us (Figure 1)?

Figure 1. Capitorg.
http://flickr.com/photos/michaelhenderson/23059611
Modified by Soowook Kim & Stephen Petrina

CAPITORGS MEET TECHNOLOGICAL LITERACY

Capitorgs learn, use, and live with technologies that are driving forces of contemporary capitalism. A cyborg (cybernetic organism) (Haraway, 1991) is an incomplete concept. Rikowski and McLaren (2002, p. 4) argue that "what postmodernists and post/trans-human theorists – protagonists for a cyborg future – blatantly ignore or deny is that our lives and 'selves' are, after all, very much centered by *capital*, as social force and social relation." "Modern Western technology," claims Feenberg (1999, p. 222), "has been profoundly shaped by capitalist enterprise. As such it privileges the narrow goal of production and profit." However, the mainstream computers-in-education literature focuses on "how to make the most intensive use of high-technology, and how to integrate the computer across the curriculum," but not on "why, whether, and where technology should be employed in schools, and to what end" (Arnold, 1996, p. 237).

A primary question of technological literacy is whether we view ever-emerging digital technologies in education as just additional tools (technocentric technological literacy) or as social forces with political, economic, social, and ideological implications (critical technological literacy) (Petrina, 2000, 2003). While "learning with technology" is emphasized in individualistic constructivist theories of learning (Jonassen, Howland, Moore, & Marra, 1999, p. 11-12), learning *about* technology is marginalized. Even though the individualistic constructivist school acknowledges a need for some level of learning *about* technology, its scope is often limited to "a very minimal practical knowledge about computers" (Papert, 1993, p. 52). In other words, theorists and practitioners of individualistic constructivism tend to take digital technologies for granted, rarely question political implications, and merely encourage students to construct knowledge with computers. This is a serious shortcoming. Even though theories of constructivism have benefits, this apolitical stance toward technology is problematic.

Learning *about* technology means more than acquiring technical skills. Learning *about* technology includes learning how technology is shaped in society. When we reduce the scope of learning *about* technology to skill acquisition, we are left with technocentric literacy, a variant on functional literacy. Only when we expand the scope of learning *about* technology beyond acquiring technical skills, do we venture on critical technological literacy, a version of critical literacy (Petrina, 2007, pp. ix-x; Petrina, Volk & Kim, 2004).

Technocentric technological literacy means that "our children's future depends on their ability to effectively use the new literacies of networked ICT (information and communication technology) such as the Internet" (Leu & Kinzer, 2003, p. 27). This is another epitome of neoliberal agendas, economic rationalism, and *homines oecomomici*. Moreover, skill acquisition is often limited to certain monopolistic or oligopolistic platforms, thereby paradoxically leading to "a simultaneous and connected *de-skilling* of the labor force" (Pippin, 1995, p. 44, italics in original).

Questions about computer-related tools are central to democratic governance. Two dominant social issues in discourses of technological literacy are access (or digital divide) and gender. When enthusiasts conclude that technology is just a tool, they rarely ask *"whose tool?"* (Moll, 2001, p. 43). For example, proprietary and closed source hardware and software are generally uncontested tools. Contemporary neoliberal societies buttress functional technological literacy with proprietary and closed source software (PCSS), which is full of monopolistic and oligopolistic intellectual property rights. Although an alternative, critical technological literacy combined with free/libre and open source software (FLOSS), has emerged, technological literacy educators rarely acknowledge this while focusing on the effective use of educational technology and PCSS-oriented gender or digital divide issues. I take this as a crisis of technological literacy.

Proprietary and Closed Source Software (PCSS) and Intellectual Property

The majority of discourses of technological literacy relate to PCSS. When we talk about technological literacy, it is difficult *not* to think about Microsoft's Windows, Internet Explorer and Office, Macromedia's Flash and DreamWeaver, Adobe's PhotoShop and Premiere, Apple's Mac OS and iMovie, and so forth. Even though they are effective applications and operating systems, PCSS programs pose several dilemmas for educators and students. First, except for educational institutions and students with huge financial resources, it is difficult to maintain software licensing budgets. Although schools license various kinds of software, they are hard pressed to buy upgrades. Schools are trapped in a never-ending cycle of software upgrades and purchases. Educators and students are tied to certain PCSS platforms, which creates a proprietary "monoculture of networked computers" (Geer, et al., 2003, p. 7).

Second, since PCSS buyers cannot read source code, two rights are fundamentally denied (Pfaffenberger, 2000). If licensees cannot customize PCSS, their modification rights are denied. However, most users rarely challenge this restriction on modification rights. In addition, since licensees cannot know what is included in PCSS (for example, privacy-leaking functions), their right to know what those programs actually do is denied.

Third, the school-work correspondence theory (Arnold, 1996) is promoted with few criticisms when educators choose PCSS to prepare students with software knowledge they seemingly need to flourish in workplaces. Although vocational post-secondary schools may necessarily have to teach specific PCSS applications, it is not necessary for public school teachers to promote PCSS, consciously or unconsciously. Noble (1984a) is quite critical of the education of technological tools which will be obsolete or change significantly in a few years.

Fourth, PCSS manufacturers curb royalty-free open standards for files and services. By maintaining closed standards file formats (e.g., .doc for Microsoft Word and .xls for Microsoft Excel) and Internet services (e.g., ActiveX technology for Internet Explorer available only on Windows), PCSS companies secure and

monopolize markets through vendor dependence (Babe & Stallman, 2004). One of the key technologies behind the Internet is HTML (HyperText and Markup Language). Every web page is supposed to display correctly on any computer system following royalty-free open HTML specifications. However, the reality is that many web pages do not display properly on Macintosh, Linux or Unix because the pages contain proprietary and closed Internet technologies. Most South Korean educational web sites discriminate against educators and students using Macintosh, Linux or Unix. We need a "computing world moving toward "technical diversity," "technological pluralism," and open standards not owned by any one company" (Lohr, 2003, para. 5; Petrina, 2007). Lessmann (2003, para. 5) succinctly states that "as long as exchanged documents are based on open standards, the software choice in company A has no impact on the software choice in company B." From where do these problems originate? The regime of intellectual property rights is one strong reason for these problems (Petrina, Volk & Kim, 2004).

Software is a collection of numerous creative algorithms that are rarely created by one single person or group. Software is typically a product of collective work. Even though a company designs unique algorithms to produce a software program, those algorithms cannot be created without previous collective work done in mathematics, computer science, and software engineering. Algorithms are social. However, producers of mainstream technologies motivated by the accumulation of capital endeavor to put algorithms in the realm of private intellectual property. When algorithms become a source for generating unlimited capital, its social characteristics are considerably buried.

Historically, a small number of intellectual property owners monopolize with tremendous control over new technologies (Martin, 1995; Shell, 2003). Intellectual property is a major pillar of neoliberalism. It remains as a philosophical and legal question whether people can copyright or patent algorithms. Koepsell (2000, p. 61) reported that "in the early days of computer programming, university students and professors would freely exchange code to run on their time-shared computers." A typical example is the source code of UNIX that was eventually locked and privatized in the 1980s (Vaidhyanathan, 2001, p. 155). Drahos notes that IBM shared the source code of its System/360 with its customers between 1964 and 1983: "In 1966 the vice president of IBM, sitting on a President's Commission on the Patent System, proposed that patents should not be granted on software.... in 1983... it [IBM] began to withhold source code from other programmers" (Drahos, 2002, p. 170). Why then, is it so difficult to publicly share software? Intellectual property (IP) and trade laws enable an unfettered accumulation of capital. IP and trade laws help establish a form of "information feudalism in a world of free markets" (Drahos, 2002, p. 1; Petrina, Volk & Kim, 2004). IP litigation threatens innovation (Kapica, 2003).

FREE/LIBRE AND OPEN SOURCE SOFTWARE (FLOSS)

For democratic, sustainable practice, people need freedom to develop and use software applications without fear of infringement on corporate IP rights. Advocates of FLOSS simply changed the rules of the game. Stallman (1983) launched the GNU (GNU's Not Unix) project, an essential landmark of the free software movement. Linus Torvalds, a university student from Finland back in 1991, provided the GNU project with the most important part, a kernel system called LINUX (LINUs + uniX) (Moody, 2001; Torvalds & Diamond, 2002) (Figure 2).

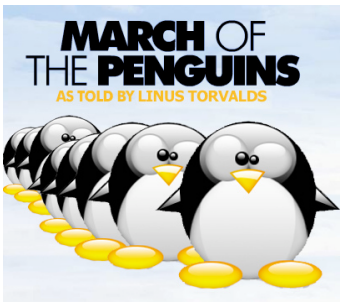

Figure 2. March of the Penguins. http://flickr.com/photos/reality/49435130

FLOSS is often called freedom software. FLOSS is not a static product with secret source code but an evolving process with open source code in which everyone is welcome to choose, use, modify, improve, and share it everyday (Murdock, 2003). FLOSS overcomes "the 'priesthood of technology', which keeps the general public in ignorance of how technology works" (Stallman, 2003b, para. 7). It poses a serious challenge to software monopolies and oligopolies (Negroponte, 1995; Williams, 2002; Stallman, 2003a; Stallman & Hill, 2003). In effect, FLOSS challenges fundamental assumptions about enclosed, self-contained, completed end products. GNU/Linux and various FLOSS applications prove that high quality decentralized software can be developed through a bazaar model while challenging traditional hierarchical software development or cathedral models (Raymond, 2001). FLOSS returns software to the public domain as public knowledge (Holtgrewe & Werle, 2001; Weber, 2004).

Of particular importance is the GNU license represented by a General Public License (GPL). Under GPL, modifications of original source code are shared free of charge and without conventional IP rights. GPL is sometimes called copyleft, defined as "a general method for making a program free software and requiring all modified and extended versions of the program to be free software as well" (Free Soft Foundation, 2003).

Free/Libre and Open Source Software (FLOSS) in Education

What are the implications of FLOSS for technological literacy? Even though a vast amount of time, money, and efforts are invested in promoting computer literacy, "the vast majority of people are still unable to read or write any kind of computer language" (Evans, 2003, para. 1). Computer literacy is narrowly defined by mastering how to create digital documents and surf the net. The expectation is that learners master how to click mouse buttons, type with a keyboard, execute certain

functions and choose from menus of specific software applications. Most do not know how to read or hand-code the simplest computer language called HTML. But, language is only one aspect. Most who label themselves technologically literate do not understand basic socio-political dimensions of technology (Petrina, 2004) or how PCSS and FLOSS work in society. Most who are labeled technologically literate possess superficial consumer literacy at best (Noble, 1984a & 1984b). Therefore, when educators encourage students to develop technological literacy through PCSS, they actually promote a form of illiteracy within narrow contexts and neoliberal agendas in larger contexts, whether aware or not.

FLOSS is ideal for schools where sharing knowledge within a public domain is important (Pfaffenberger, 2000; Thompson, 2002). Though FLOSS is not a panacea for technological literacy, it offers educational possibilities. Stallman (2003b) explains six reasons why schools should use *exclusively* free software:

1. Free software can save the schools money (para. 3);

2. School should teach students ways of life that will benefit society as a whole (para. 5);

3. Free software permits students to learn how software works (para. 6);

4. Free software encourages everyone to learn (para. 7);

5. We expect schools to teach students basic facts, and useful skills, but that is not their whole job. The most fundamental mission of schools is to teach people to be good citizens and good neighbors--to cooperate with others who need their help (para. 8);

6. Teaching the students to use free software, and to participate in the free software community, is a hands-on civics lesson. It also teaches students the role model of public service rather than that of tycoons (para. 9).

Moreover, through a special technology called LTSP (Linux Terminal Server Project: http://ltsp.org), schools can utilize old Pentium-I-class computers and run them at the speed of the current Pentium-IV-class computers, thereby supporting cost-effective and environment-friendly computing.

In spite of obvious advantages of FLOSS, few schools have turned to FLOSS. Educating students merely with PCSS produces docile future consumers (Giroux, 1998). It makes students internalize a neoliberal regime of intellectual property and accept monopolistic and oligopolitical production and ideation processes. By disadvantaging students whose families have limited incomes, PCSS-based education reinforces the reproduction of social class through education (Bourdieu & Passeron, 1990).

Students attending near-demolished and impoverished schools are experiencing a qualitatively and quantitatively different world than students attending prestigious schools. Human existence is bound by time and space. Within the limits of time and space, teachers cannot teach everything and students cannot learn everything. Selection will always occur within curriculum. Educational technology is also bound by time and space. The space created by educational technology is limited by aggregate numbers of computers, which create a digital divide based on access. Even if there were an adequate computer-to-student access ratio, inability to upgrade every PCSS program creates a digital divide based on software. "The limits of *language* (of the language which alone I understand) mean the limits of *my* world" (Wittgenstein, 1922/1974, p. 57, proposition 5.62). When educators fail to introduce FLOSS, they limit the world, language, time and space of their students. In order to provide wider, more critical perspectives on technology for students, educators are "positively obligated to use open source software" (Pfaffenberger, 2000, p. 5).

TOWARDS FREE/LIBRE AND OPEN SOURCE SOCIETY

When educators and students learn more about FLOSS and its implications for society, they also extend their critical consciousness beyond software. FLOSS can inspire educators to enter into a realm of open access, and develop and share educational materials cooperatively on the Internet (Hirst, 2001; Moore, 2002) just as researchers, scientists, authors, and artists collaborate in computer networks (Young, 2002). As Moody (2001) and Lessig (2001) argue, FLOSS is not merely about software. "Open code," another term for FLOSS, "is a foundation to an open society" (Lessig, 1999, p. 108). What about free/libre and open source hardware (FLOSH), democracy (FLOSD), creative work (FLOSCW), and medicine (FLOSM)? Free/libre and open source hardware is important given that most computer and communication hardware manufacturers provide drivers (software operating specific hardware) for only Windows or Macintosh (Rahmat, 2003; Spooner, 2003). Free/libre and open source democracy (FLOSD) proponents note that "one model for the open-ended and participatory process through which legislation might occur in a networked democracy can be found in the 'open source' software movement" (Rushkoff, 2003, p. 56). Creative Commons (http://creativecommons.org/license) provides authors with a variety of flexible licenses to enable the use, duplication, modification, or display of creative works under certain conditions specified by the authors. This enables a possibility of free/libre and open source creative work (FLOSCW). Free/libre and open source medicine (FLOSM) utilizes the FLOSS concept to produce generic, affordable medicine for needy patients around the world.

A core idea running through FLOSH, FLOSD, FLOSCW, and FLOSM is the free/libre exchange of ideas and free/libre production and distribution processes outside of IP and trade laws (Petrina, Volk & Kim, 2004). The goal is simply "open and collaborative projects to create public goods" (Lessig, 2004, p. 262). This represents a challenge to governance systems that discipline students and

workers with capitalist agendas (Bowles & Gintis, 1976; 2001; Gramsci, 1975/1992, 1975/1996; Kroker, 2004; Noble, 2002; Schmidt, 2000). Vaidhyanathan (2004) describes this as a clash between capitalism and egalitarian, democratic society.

One of Lao Tzu's famous proverbs is that "one brings good things about, but has no intention of possessing them" (Lao Tzu, circa 300 B.C./1995, p. 8). Russel rephrased this as "production without possession" (1966, p. 194). Every human activity, including the process of creating, modifying, distributing, and using FLOSS, is part of an ever-continuing cosmic movement. When a process is stopped, locked and possessed by a person or group, it becomes dead (Kim, 1999, p. 136). IP legal systems and neoliberal capitalism work directly against the cosmic law of production without possession. Life is an everlasting process of becoming (Choe, 1857/2004). FLOSS is important for that process. A society populated by capitorgs (capital organisms) laboring for the unfettered accumulation of capital is the "one-dimensional" world Marcuse described. We need FLOSS (Free/Libre and Open Source Software) to foster FLOSS (Free/Libre and Open Source Society). A society designed through FLOSS is governed by principles of *pletharchia* (*pl_thos* [a whole populace] + *arch* [origin]) (Kim, 2004, p. 44) and works for life, not merely for capitorgs' desires (Figure 3).

Figure 3. Free your world. http://www.kde-look.org/content/show.php?content=38973

REFERENCES/BIBLIOGRAPHY

Adamson, Gregory Dale. (2002). *Philosophy in the age of science and capital*. London, UK: Continuum.

Apple, Michael. (1990). *Ideology and curriculum*. New York: Routledge.

Arnold, M. (1996). The high-tech, post-Fordist school. *Interchange*, *27*(3/4), 225-250.

Babe, Anarcho. (2004). *Interview with Richard Stallman - Free software, free society!* Retrieved November 15, 2004, from http://www.scotland.indymedia.org/newswire/display/266/index.php

Bourdieu, P., & Passeron, J. (1990). *Reproduction in education, society, and culture*. London: Sage.

Bowles, S., & Gintis, H. (1976). *Schooling in capitalist America: Educational reform and contradictions of economic life*. New York: Basic Books.

Bowles, S., & Gintis, H. (2001). Schooling in capitalist America revisited. Retrieved November 20, 2004, from http://www.umass.edu/preferen/gintis/soced.pdf

Callinicos, Alex. (2003). *An anti-capitalist manifesto*. Malden, MA: Blackwell.

Choe, Han-Gi. (2004). *Science of Gi* (Byeong-Uk Son, Trans.). Seoul, Korea: Tong-na-moo. (Original work published 1857).

Darder, A. (2002). *Reinventing Paulo Freire: A pedagogy of love*. Boulder, CO: Westview.

Drahos, Peter. (2002). *Information feudalism: Who owns the knowledge economy?* New York: New Press.

Evans, D. (2003, November 6). Smash the Windows. *The Guardian*. Retrieved November 10, 2003, from http://www.guardian.co.uk/comment/story/0,3604,1078616,00.html

Feenberg, Andrew. (1999). *Questioning technology*. New York: Routledge.

Free Software Foundation. (2003). *Categories of Free and Non-Free Software*. Retrieved December 11, 2003, from http://www.gnu.org/philosophy/categories.html

Freire, P. (1997a). *Pedagogy of the heart*. New York: Continuum.

Freire, P. (1997b). A response. In P. Freire, J. W. Fraser, D. Macedo, T. McKinnon, & W. T. Stokes (Eds.), *Mentoring the mentor: A critical dialogue with Paulo Freire* (pp. 303-329). New York, NY, USA: Peter Lang.

Geer, D. et al. (2003). *CyberInsecurity: The cost of monopoly – How the dominance of Microsoft's products poses a risk to security*. Retrieved September 25, 2003, from http://www.ccianet.org/papers/cyberinsecurity.pdf

Giroux, Henry. (1998, October). Education incorporated. *Educational Leadership*, 12-17.

Graff, Harvey J. (1995). *The labyrinths of literacy: Reflections on literacy past and present*. Pittsburgh: University of Pittsburgh.

Gramsci, Antonio. (1975/1992). *Prison notebooks*, Vol. 1 (J. A. Buttigieg & A. Callari, Trans.). New York: Columbia University Press.

Gramsci, Antonio. (1975/1996). *Prison notebooks*, Vol. 2 (J. A. Buttigieg, Trans.). New York: Columbia University Press.

Haraway, D. (1991). A cyborg manifesto: Science, technology, and socialist-feminism in the late twentieth century. In D. Haraway, *Simians, cyborgs, and women: The reinvention of nature* (pp.149-181). New York: Routledge.

Hirst, Tony. (2001). *The open source teaching project (OSTP): Research note*. Retrieved August 1, 2004, from http://ict.open.ac.uk/reports/7.pdf

Holtgrewe, U. & Werle, R. (2001). De-commodifying software?: Open source software between business strategy and social movement. *Science Studies*, *14*(2), 43-65.

Jonassen, D. H., Howland, J., Moore, J., & Marra, R. M. (1999). *Learning to solve problems with technology: A constructivist perspective, 2nd ed*. Upper Saddle River, NJ: Merrill Prentice Hall.

Kapica, Jack. (2003, December 11). Copyright litigation threatens innovation. *The Globe and Mail*. Retrieved December 11, 2003, from http://www.globetechnology.com/servlet/story/RTGAM.20031211.wtxkapica1211/BNStory/Technology

Kim, Y.-O. (1999). *Lao Tzu and the 21st century [1]*. Seoul: Tong-na-moo.

Kim, Y.-O. (2004). *Classics of Eastern Learning and pletharchia*. Seoul: Tong-na-moo.

Koepsell, D. R. (2000). *The ontology of cyberspace: Philosophy, law, and the future of intellectual property*. Chicago, IL: Open Court.

Kroker, A. (2004). *The will to technology: Heidegger, Nietzsche, and Marx*. Toronto: University of Toronto Press.

Lao Tzu. (1995). *The complete works of Lao Tzu: Tao Teh Ching and Hua Hu Ching* (H. Ni, Trans). Santa Monica, CA, USA: Seven Star. (Original work published circa BC 300).

Lessig, L. (1999). *Code and other laws of cyberspace*. New York: Basic Books.

Lessig, L. (2001). *The future of ideas: The fate of commons in a connected world*. New York: Vintage Books.

Lessig, L. (2004) *Free culture: How big media uses technology and the law to lock down culture and control creativity*. New York: Penguin Press.

Lessmann, J. (2003, September 23). An alternative to so-called "Open Source Bills." *DesktopLinux.com*. Retrieved October 19, 2003, from http://www.desktoplinux.com/articles/AT9595145449.html

Leu, D. J. & Kinzer, C. K. (2003). Toward a theoretical framework of new literacies on the Internet: Central principles. In J. Richards, & M. McKenna (Eds.), *Integrating multiple literacies in K-8*

classrooms: Cases, commentaries, and practical applications (pp. 18-37). Mahwah, NJ: Lawrence Erlbaum Associates.

Lohr, S. (2003, October 27). Two companies at odds over the Internet's future. *The New York Times*. Retrieved October 28, 2003, from http://www.nytimes.com/2003/10/27/technology/27neco.html?pagewanted=2

Martin, B. (1995). *Against intellectual property*. Retrieved August 8, 2003, from http://www.eff.org/IP/against_ip.article

McLaren, P. (2000). *Che Guevara, Paulo Freire, and the pedagogy of revolution*. Lanham, MD, USA: Rowman & Littlefield.

McLaren, P., & Farahmandpur, R. (2002). Recentering class: Wither postmodernism? Toward a contraband pedagogy. In D. Hill, P. McLaren, M. Cole, & G. Rikowski (Eds.), *Marxism against postmodernism in educational theory* (pp.239-273). Lanham, MD: Lexington.

Moll, M. (2001). If it's only a tool, whose tool is it?: teachers, technologies and the new world of work. In M. Moll (Ed.), *But it's only a tool!: The politics of technology and education reform* (pp. 43-63). Ottawa: Canadian Centre for Policy Alternatives.

Moody, G. (2001). *Rebel code: The inside story of Linux and the open source revolution*. Cambridge, MA: Perseus.

Moore, A. H. (2002). Lens on the future: Open-source learning. *Educause Review*, *37*(5), 43-51.

Murdock, I. (2003, July 30). Reconsidering Linux. *CNET*. Retrieved August 25, 2003, from http://news.com.com/2010-1071_3-5057321.html

Negroponte, N. (1995). *Being digital*. New York: Alfred A. Knopf.

New London Group. (1996). A pedagogy of multiliteracies: Designing social futures. *Harvard Educational Review*, *66*(1), 60-92.

Noble, D. (2002). *Digital diploma mills: The automation of higher education*. New York: Monthly Review Press.

Noble, D. (1984a). Computer literacy and ideology. *Teacher College Record*, *85*(4), 602-614.

Noble, D. (1984b). The underside of computer literacy. *Raritan*, *3*(4), 37-64.

Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. New York: Basic Books.

Peters, M. (1999). Freire and postmodernism. In P. Roberts (Ed.), *Paulo Freire, politics and pedagogy: Reflections from Aotearoa – New Zealand* (pp. 113-122). Palmerston North, NZ: Dunmore.

Petrina, S. (2000). The politics of technological literacy. *International Journal of Technology and Design Education 10*(2), 181-206.

Petrina, S. (2003). 'Two cultures' of technical courses and discourses: The case of computer-aided design. *International Journal of Technology and Design Education*, *13*(1), 47-73.

Petrina, S., Volk, K. and Kim. S. (2004). Technology and rights. *International Journal of Technology and Design Education*, *14*(3), 181-204.

Pfaffenberger, B. (2000, March 2). Linux in higher education: Open source, open minds, social justice. *Linux Journal*. Retrieved July 20, 2003, from http://www.linuxjournal.com/article.php?sid=5071

Pippin, R. B. (1995). On the notion of technology as ideology. In A. Feenberg, & A. Hannay (Eds.), *Technology and the politics of knowledge* (pp. 43-61). Bloomington: Indiana University Press.

Rahmat, O. (2003, July 4). Will open source save hardware? *Tom's Hardware*. Retrieved September 25, 2003, from http://www6.tomshardware.com/column/20030704/index-03.html

Raymond, E. S. (2001). *The cathedral and the bazaar: Musings on Linux and open source by an accidental revolutionary (revised)*. Sebastopol, CA: O'Reilly.

Rikowski, G..(2002). Education, capital and the transhuman. In D. Hill, P. McLaren, M. Cole, & G. Rikowski (Eds.), *Marxism against postmodernism in educational theory* (pp. 111-143). Lanham, MD: Lexington.

Rikowski, G., & McLaren, P.(2002). Postmodernism in educational theory. In D. Hill, P. McLaren, M. Cole, & G. Rikowski (Eds.), *Marxism against postmodernism in educational theory* (pp. 3-13). Lanham, MD: Lexington.

Roberts, P. (1999). Freire, neoliberalism and the university. In P. Roberts (Ed.), *Paulo Freire, politics and pedagogy: Reflections from Aotearoa – New Zealand* (pp. 97-111). Palmerston North, NZ: Dunmore.

Roberts, P. (Ed.). (1999). *Paulo Freire, politics and pedagogy: Reflections from Aotearoa – New Zealand*. Palmerston North, New Zealand: Dunmore.

Rushkoff, D. (2003). *Open source democracy: How online communication is changing offline politics*. Retrieved October 28, 2003, from http://www.demos.co.uk/opensourcedemocracy_pdf_media_public.aspx

Russell, B. (1966). *The problem of China*. New York, NY, USA: Century.

Schmidt, J. (2000). *Disciplined minds: A critical look at salaried professionals and the soul-battering system that shapes their lives*. Lanham, MD, USA: Rowman & Littlefield.

Shell, G. R. (2003, October 8). Suing your customers: A winning business strategy? *Knowledge at Wharton*. Retrieved October 13, 2003, from http://knowledge.wharton.upenn.edu/articles.cfm?catid=7&articleid=863&homepage=yes

Spooner, J. G. (2003, September 17). AMD chief salutes standard PC chips. *ZD Net*. Retrieved September 25, 2003, from http://zdnet.com.com/2100-1103_2-5078381.html

Stallman, R. (1983, September 27). New Unix implementation. *net.unix-wizards, net.usoft*. etrieved August 31, 2003, from http://groups.google.com/groups?selm=771%40mit-eddie.UUCP

Stallman, R. (2003a, September 21). Open source *Is* politics. *Linux World*. Retrieved September 22, 2003, from http://www.linuxworld.com/story/34189.htm

Stallman, R. (2003b, November 8). Why schools should use exclusively free software. *News Forge*. Retrieved November 9, 2003, from http://www.newsforge.com/software/03/11/09/0228239.shtml

Stallman, R, & Hill, N. (2003, June 5). That's fighting talk. *The Guardian*. Retrieved September 1, 2003, from http://www.guardian.co.uk/online/story/0,3605,970294,00.html

Thompson, A. (2002). The open source software movement. *Journal of Computing in Teacher Education*, *18*(4), 110, 123.

Torvalds, L. & Diamond, D. (2002). *Just for fun: The story of an accidental revolutionary*. New York: Harper Collins.

Vaidhyanathan, S. (2001). *Copyrights and copywrongs: The rise of intellectual property and how it threatens creativity*. New York: New York University.

Vaidhyanathan, S. (2004). *The anarchist in the library: How the clash between freedom and control is hacking the real world and crashing the system*. New York: Basic Books.

Weber, S. (2004). *The success of open source*. Cambridge: Harvard University.

Williams, R. (1989). *Resources of hope*. New York: Verso.

Williams, S. (2002). *Free as in freedom: Richard Stallman's crusade for free software*. Sebastopol, CA: O'Reilly.

Wittgenstein, L. (1922/1974). *Tractatus Logico-Philosophicus,* trans. D.F. Pears & B.F. McGuinness. London: Routledge.

Young, B. (2002). The open source movement, publishing, and the dissemination of knowledge. In *E-Learn 2002 World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education*. *Proceedings* (7th, Montreal, Quebec, Canada, October 15-19, 2002).

AFFILIATIONS

Soowook Kim, Ph.D. Candidate

http://www.korean.arts.ubc.ca/kim

Department of Curriculum Studies,

University of British Columbia