

Writing in the Disciplines versus Corporate Workplaces: On the Importance of Conflicting Disciplinary Discourses in the Open Source Movement and the Value of Intellectual Property

Brian D. Ballentine, West Virginia University

Abstract: Writing programs and more specifically, Writing in the Disciplines (WID) initiatives have begun to embrace the use of and the ideology inherent to, open source software. The Conference on College Composition and Communication has passed a resolution stating that whenever feasible educators and their institutions consider open source applications. Although well-intended, open source ideology often runs counter to a corporate business culture that produces revenue from aggressively protecting intellectual property. Such aggressiveness helps contribute to the perception of a rigid binary between open source and corporate communities. This article challenges WID scholars and instructors to better acknowledge the complexity of corporate discourses and to develop more nuanced stances in regard to open source and intellectual property. The merits of open source are not simply attacked here but instead the article insists that as WID ponders open source that the discourses from the corporate workplace are not ignored. The concern is that the values inherent in the open source community and built into the technologies it produces are the dominant or only influence on how writing instructors approach intellectual property. This article begins with an examination of open source values and their tenuous relationship to intellectual property law. The remainder of the article is an examination of *why* and *where* WID is appropriating open source technologies and its influential development model. Brief recommendations are offered for specific technologies and the article explains how regardless of monetary savings, open source applications enable a more robust critical literacy in the WID classroom.

English departments, writing programs, and to a certain extent the humanities in general display a growing enthusiasm for embracing open source software and along with it, its development ideology. Embedded within that ideology is an attitude toward intellectual property that endorses alternative licensing options in order to promote knowledge sharing. In addition, the open source development model welcomes collaboration, favors interaction, and insists on an intense focus on the audience or end-user. All of these concepts, even imported from the world of software, resemble familiar values to those who teach writing and adhere to a writing pedagogy advocating revision and multiple iterations of student work. As scholars who research intersections of writing and technology have noted, "[w]henever humans invent and build a technology, they build in values, even if they do not intend to" (Gurak, 2005, p. 209). And as responsible,

Across the Disciplines
A Journal of Language, Learning and Academic Writing

wac.colostate.edu/atd
ISSN 554-8244

Across the Disciplines is an open-access, peer-review scholarly journal published on the WAC Clearinghouse and supported by [Colorado State University](#) and [Georgia Southern University](#). Articles are published under a [Creative Commons BY-NC-ND license](#) (Attribution-NonCommercial-NoDerivs) ISSN 1554-8244. Copyright © 1997-2017 The WAC Clearinghouse and/or the site's authors, developers, and contributors. Some material is used with permission.

critical users of technology, humanists have begun to parse the complexities of open source, which is thought to have potential application in addressing a variety of needs ranging from alleviating technology costs to realigning publishing models.

However, even with the best of intentions, open source ideology often runs counter to a corporate business culture that often produces revenue from licensing and aggressively protecting a variety of intellectual property. Such aggressiveness helps contribute to the perception of a rigid binary between open source and corporate communities. This article challenges scholars and instructors, specifically those involved with writing in the disciplines (WID) curriculum, to better acknowledge the complexity of corporate discourses and to develop more nuanced stances in regard to open source. As this special collection recognizes, technologies and the norms for producing writing change rapidly. As we respond to these changes, we cannot afford to proceed armed only with a binary perspective. WID is a powerful movement because it not only helps "students to think critically about disciplinary content; but WID also helps students to develop their writing skills as they articulate their understanding of content in genres appropriate to professional audiences" (Carpenter & Krest, 2001, p. 47). In the WID classroom, we encounter students who are future engineers, scientists, and business professionals whose disciplines' approach to intellectual property may differ greatly from the open source ideology being embraced by our discipline. This article does not seek to debunk the merits of open source, but instead will insist that as the humanities adopt and adapt various technologies and ideologies from the open source community, the interests, perspectives, and discourses from the corporate workplace are not ignored by WID. The concern is that the values inherent in the open source community and built into the technologies it produces are the dominant or only influence on how writing instructors approach intellectual property. As the norms associated with writing production shift, we cannot allow a division to form between how our own discipline comes to understand, integrate, and teach intellectual property and how corporations understand and value it.

This article begins with an examination of open source values and their tenuous relationship to intellectual property law. Open source applications are protected with a variety of available licenses which have been crafted to ensure that "property" within the community continues to circulate freely. The remainder of the article is an examination of *why* and *where* our discipline is appropriating open source in terms of both specific technologies and as an influential development model. As open source initiatives are proving themselves to be stable and competitive next to costly proprietary software, open source has won the attention of the often resource-starved writing community. Most notably, the National Council of Teachers of English and the Conference on College Composition and Communication issued recently the "Proposed NCTE/CCCC Resolution on the Adoption and Use of Open Source Software" in which the members state, "we support open source software initiatives, and would like to encourage English composition teachers and their professional organizations to consider the open source options appropriate to their instructional and professional situations" (Background section, para. 3). I will focus on two of several very strong claims within the NCTE/CCCC resolution pointing in the direction of why and where open source. First, "Open source software has the potential to control spiraling technology costs" and second, "The open source development model parallels the academic model of knowledge creation and distribution" (Benefits of Open Source section, paras. 2 & 5). Critiques of these claims are offered in an attempt to help navigate an often daunting integration of open source technologies and ideals while demonstrating the falsity of a rigid open versus closed source binary. I offer brief recommendations on specific technologies and explain how regardless of monetary savings, open source applications enable a more robust critical literacy in the WID classroom.

Alongside my advocacy, I believe that a vital component to integrating open source is the inclusion of corporate workplace perspectives. Closer inspection of corporate communities reveals not protectionist intellectual property mongers but communities learning to adjust to a shifting landscape by taking cues from the open source community. Throughout the article I will continue to claim that our strategies for adjusting to the changing norms of writing should include using the writing classroom as an environment

to teach a balanced perspective on how intellectual property is used and valued. That is, as instructors "consider the open source options appropriate to their instructional and professional situations" (Background section, para. 3), there must be a candid discussion regarding the values built into those technologies and how those values may or may not be shared by different communities. As WID evolves and grows to offer a variety of composition, professional, and technical communication courses, academic interests and needs should not be allowed to eclipse those of industry and the workplace.

Open Source Values & Intellectual Property Law

In the world of software, assigning intellectual property rights is exceedingly complicated. In 2002, Judge Richard Posner wrote in *Dædalus* that, "the products of the new technologies are sometimes hard to fit into the law's pigeonholes. Computer software is a kind of text, which implies that copyright is the proper regime; a kind of machine, which implies patent is the proper regime; and a kind of algorithm, which has traditionally not been protected by either body of law" (p. 6). In a bit of irony, open source projects need intellectual property law just as much proprietary applications. Open source turns copyright upside down not by releasing creative work unprotected into the public domain but by using the law to secure the right to reproduce and redistribute work. As the well-known legal scholar Lawrence Lessig remarked recently on his blog, "...the freedom to grant freedoms is part of what copyright law gives a copyright owner" (para. 19). Within the corporate community, however, it is common practice to use copyright law as a means to make intellectual property less and not more accessible.

To clarify, the "property" that intellectual property law protects is unlike a car or a bicycle, for example. Unique characteristics distinguish it from other tangible goods. As defined by the World Intellectual Property Organization, "Intellectual property relates to items of information or knowledge, which can be incorporated in tangible objects at the same time in an unlimited number of copies at different locations anywhere in the world. The property is not in those copies but in the information or knowledge reflected in them" (pp. 5-6). In the instance of a car, within the design of the vehicle there may be engineering concepts protected by intellectual property that make it more fuel efficient. As the car and its design are mass produced, intellectual property law does not prevent a buyer from dismantling, modifying, or reselling the car but it does stop a competitor from copying, manufacturing, and profiting from the "information or knowledge" enabling the car to be more fuel efficient. The dilemma, according to legal scholars, is that, "It is especially difficult physically to exclude people from using others' intangible ideas, and sometimes even the physical embodiment of those ideas and to arrange necessary compensation for creators" (Nard, Barnes & Madison, 2006, p. 14).

The problem is exacerbated when the property in question can be digitized. Computers facilitate the copying of works protected by intellectual property laws to the point of effortlessness. The speed and connectivity of the internet means a potentially enormous distribution of that property. In the case of the car, a competitor stealing ideas will at the very least incur engineering and production costs that may or may not deter infringement. This is not to disregard the costs associated with computers and connectivity but for many the cost of access to technology continues to decrease to the point where it is no longer a deterrent. Posner (2002) claims that emergent technologies have "caused the law to lose its bearings" and that there exists "an extreme disparity between the cost of creation and the cost of making and distributing copies, which indeed approaches zero whenever the copy is made electronically and is distributed over the Internet" (p. 5). As a result, corporations clamor for increased legal regulation. That clamor is heard the loudest from organizations such as the Recording Industry Association of America (RIAA) and the Motion Picture Association of America (MPAA) but it also comes from any number of sources including pharmaceutical and chemical companies, publishing houses, and engineering firms. When physicality and materiality no longer provide protection against unwanted copying and distribution of intellectual property, disciplinary values will be at odds with one another.

Among the types of intellectual property law including patents, trademarks, and trade secrets, copyright law has the center stage for contestation. Copyright law in the United States can be traced back to Article 1, Section 8 of the Constitution which reads, "The Congress shall have Power...To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries." The term "Writings" has been interpreted broadly to encompass creative works such as music, drama, film, graphics, sculpture, and now computer code. Copyright law has undergone numerous revisions to not only make it easier for creative works to receive protection but also to extend the length of protection. Creative works produced after January 1, 1978 "are protected by copyright for 70 years past the death of the last surviving author. If a work was made for hire after that date, the term is 95 years from first publication or 120 years from the year of creation, whichever expires first" (ASME, 2001, p. 24). Copyright protection takes effect when an intangible idea takes on a fixed form of some kind. Once in a fixed form, protection is automatic and a creator is not required to register for the copyright unless he or she is going to litigate. As copyright protects an enormous array of human creative works, that fixed format may be a piece of paper, canvas, sculpted stone, or a compact disc. It may also be a computer hard drive housing the underlying source code for either a proprietary or open source software application.

Steven Weber (2004), a political scientist, writes that he understands open source as, "an experiment in social organization around a distinctive notion of property. The conventional notion of property is, of course, the right to exclude you from using something that belongs to [him]. Property in open source is configured fundamentally around the right to distribute, not the right to exclude" (p. 3). Indeed a mantra often heard associated with open source is that *information wants to be free*. Free not just in the sense of no cost, but free as in liberated from exclusionary practices. Weber goes on to remark that if this inversion "feels awkward on first reading, that is a testimony to just how deeply embedded in our intuitions and institutions the exclusion view of property really is" (2004, p. 3). From the vantage of a corporate setting, valuing the right to distribute over the right to exclude, changes conceptions of property in precarious ways.

The key then to protecting the intellectual property of an open source application is a copyright license written specifically to ensure the right to distribute. While there have been numerous licenses crafted in support of open source projects (and other creative works) the GNU General Public License or GPL originally written for Richard Stallman's GNU operating system is the most widely used in the open source community. At present, the GPL covers approximately 70% of the popular Linux operating system distributions (Tiemann, 2005, p. 19). The GPL, which has also become known as a "copyleft" license, is now in its third version that was issued in June of 2007. Its preamble reads:

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users... Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things. (Preamble section, paras. 2-3)

To add to the complexity of open source licensing, there are other popular licenses including several versions of the Berkeley Software Distribution or BSD license. Also, Creative Commons, an organization founded by Lessig, has a variety of alternative licensing options and tools that give individuals the ability to assign different degrees of freedom to their work ranging from "some" to "all" rights reserved. There are ongoing debates over which of these licenses best represent open source values. For example, the GPL does not prohibit anyone from selling open source software for profit. This enables the software company Red Hat to "sell" a Linux distribution with which the company provides the customer varying degrees of service

and support. Versions of the Creative Commons license, however, allow copyright holders to prevent others from making a profit and/or creating a derivative work. But regardless of the specific license selected, the official Open Source Initiative (opensource.org) stipulates that the license ensure that the source code for the application must not only be included with the distribution but that it also is accessible. "Deliberately obfuscated source code is not allowed" (The open source definition, Source Code section, para. 1).

At the extremes of the rigid open versus closed source binary are competing discourses that reveal differing perspectives not just on property and law, but culture itself. In response to the growing stringency of copyright law, Lessig warns: "The law is changing; that change is altering the way our culture gets made; that change should worry you – whether or not you care about the Internet" (2004, p. xv). *If* proprietary companies do insist on using the law to stifle the flow of information, it will mean that we regress to merely using the applications developed for us. We are no longer editing, altering, and sharing with a community in which we could have become co-developers. Competing discourses are silenced and information flows in one direction only. More to the point, we become "Passive recipients of culture produced elsewhere" (Lessig, 2004, p. 37). Placing a premium on the freedom to share information and the need to hear competing discourses is how humanists find a kindred spirit in and an affinity for open source.

Open Source – Why and Where

Costs & Classroom Payoffs

The connection formed between the humanities and the open source community's ideological approaches to distribution and exclusion can, however, have the unfortunate result of silencing corporate discourses. Despite our inclination to view corporate discourses as the *silencers* and not the *silenced*, these discourses must be heard. As humanities instructors in a technological age that complicates and shifts norms for writing production, we have a responsibility to introduce students to critical approaches to technology that interrogate its many relationships with culture, society, politics, and law. Such critical approaches to technology, according to Neil Postman (1995), provide an education on "how the meanings of information and education change as new technologies intrude upon culture, how the meanings of truth, law, and intelligence differ among oral cultures, writing cultures, printing cultures, electronic cultures. Technology education is not a technical subject. It is a branch of the humanities" (p. 191). Although using open source software in the classroom can be defended from the standpoint of achieving this critical pedagogy, the additional claim that open source materials will save money is, if not false, not yet proven. In this section, I will examine the first of the two targeted NCTE/CCCC claims regarding the cost saving potential of open source along with what I believe to be the more important opportunity to increase critical literacy instruction in a WID course.

The need for critical literacy has been advanced by scholars such as Stuart Selber who in his *Multiliteracies for a Digital Age*, details the criteria for what he calls the "multiliterate" student. While Selber agrees that "functional literacy" is necessary and valuable, a curriculum built only on acquiring skill-sets "can take a rather monolithic and one-dimensional approach, ignoring the fact that computer technologies are embedded in a wide range of constitutive contexts, as well as entangled in value systems" (2004, p. 22). As an antidote, Selber advocates the addition of a "critical literacy" designed to scrutinize technology and its embedded values. He writes:

Whereas teachers of writing and communication have increasingly called for reflective approaches, conventional programs rarely dwell on social, political, and economic contexts. As a rule, then, students are not encouraged to ask important questions when it comes to technology development and use: What is lost as well as gained? Who profits? Who is left behind and for what reasons? What is privileged in terms of literacy and learning and cultural

capital? What political and cultural values and assumptions are embedded in hardware and software? (2004, p. 81)

Introducing students to open source technologies provides an excellent contrast to the dominant discourse of the proprietary vendors. It is also an opportunity to highlight Selber's critical literacy. Selber continues by remarking that, "literacy is emancipatory when it encourages students to put multiple discourses in conversation with each other, that is, to critique one discourse with another, in order to develop critical analytical capacities" (2004, p. 98). Obviously, this conversation can only happen if we do not snuff out deliberately the concerns of the proprietary companies.

When using open source applications in the classroom I attempt to introduce students to this conversation immediately. On the first day of class in a WID course like professional or technical communication where there will be an online component, I simply ask students if anyone is familiar with open source. Unless there are computer or software engineers in the room, very few hands go up – sometimes none. Not mentioning that it is free of charge and open source, I then ask if anyone has used Firefox as their browser and often as many as half of the class will raise their hands. It would seem then, that the answer to the titular question posed by Elizabeth Ellsworth (1992), "Why Doesn't This Feel Empowering? Working Through the Repressive Myths of Critical Pedagogy," is not just that we are failing to offer robust discussions on critical literacy, but that we are not situating students amidst these competing discourses.

To do this, however, takes effort on our part by engaging and learning new technologies. Unfortunately, those efforts do not necessarily translate to cost reductions. Scholars such as Susan McLeod and Eric Miraglia have asked that members of WAC programs become "architects of change" and approach the struggles ahead not with the question "Will WAC survive?" but "*How* will WAC survive?" (2001, p. 4). There is no question that advanced writing technologies and software are an additional strain on existing resources. Along with others, I believe that "WID extends WAC" and that "WAC needs its counterpart, writing in the disciplines (Carpenter & Krest, 2001, p. 47). WAC/WID movements using open source applications are not, however, guaranteed an antidote for disappearing resources. Below is the remaining text from the NCTE/CCCC proposal (on April 5, 2008 the proposal was approved as a resolution) that supports their claim for open source technologies in regard to the "potential to control spiraling technology costs."

Open source software and upgrades are often free—although they require an initial commitment on the part of teachers and systems administrators to learn the software and teach it to others. Although support costs may rise in the short term as the new software is introduced, over time the costs should not exceed (and may fall below) those incurred by similar commercial products. Additionally, open source software frees institutions from the costs of licensing and upgrades, and prevents vendor lock. (Benefits of Open Source section, para. 2)

The proposal does not mention specific open source applications, citing the varying needs of individual institutions. Nor does it provide any evidence or hard figures on what industry refers to as "total cost of ownership," or TCO, for open source versus proprietary applications. Those figures, which the proposal suggests will trend in a money saving direction, often favor proprietary software. In a 2005 survey of 500 North American users, the Yankee Group reported that "there is no universal clear-cut TCO basis to compel the corporate masses to do a wholesale switch from Windows to Linux" (2005, DiDio). Linux, the open source operating system and golden child, has been making some significant inroads in the server market. Whenever you run a search with Google for example, massive server-farms running Linux process and send back the results. But buying new versions of Microsoft's server software at thousands of dollars a copy for thousands of machines simply does not make sense for a company like Google that specializes in computer

innovations. Even if corporations and universities employ information technologists with the appropriate training in open source applications, there is no guarantee costs will decline.

In the world of desktop or personal computing different Linux distributions hold an estimated 1% of the market share (Gomes, 2007). Even as Linux distributions have increased in stability and performance, their chances of being your next operating system are limited when so many new computer purchases (both personal and those by a corporation or university) come with a pre-installed operating system like Windows Vista or Mac OS X Leopard. Dell is the only mainstream computer retailer to offer a version of Linux, in this case Ubuntu, as an optional operating system at the time of purchase. Selecting Ubuntu saves the consumer \$50. In his review of Ubuntu for the *Wall Street Journal*, Lee Gomes (2007) wanted to "disabuse an actual Ubuntu user of the notion that a non-Windows operating system is security utopia, where hackers are powerless and children are all above average. [He] recently installed the April version of Ubuntu on his home machine and promptly was informed that more than 50 security patches to problems discovered in the interim awaited [his] downloading" (B1). Gomes quipped, "Who does Ubuntu think it is? Windows?"

The so-called "browser war" between Microsoft's Internet Explorer and Mozilla's open source browser Firefox is another major front between open source and proprietary software. Mozilla claims that there are more than 125 million active Firefox users and most estimates put Firefox as controlling approximately 17% of the browser market (Lai, 2008). But since Microsoft also distributes its browser at no cost, many Firefox users cite their reason for switching as either the principle of using an open source application or the reports that Firefox is more secure than Explorer. Advocates for Explorer claim that the release of version 7.0 addressed the security issues in question and some analysts suggest that it is actually Firefox that is behind in functionality and service especially in terms of mainstream corporate needs. Eric Lai (2008) of Computer World reports, "Mozilla thus far has neglected to develop tools to help IT departments deploy and manage Firefox, and it doesn't offer paid technical support services to risk-averse corporate users" (para. 5).

All of this is to say that where WAC/WID movements may see the opportunity to implement an open source technology is not only quite narrowly context specific but may require a level of assertion or even risk. Why writing instructors choose to take this risk may stem from the promise of saving money or the harsh reality of having no budget for software in the first place. I am also not discounting scenarios where adopters just want to support open source values. Whatever the impetus, I recommend starting small. Many of us are not in the position to make a case for a university-wide migration to Linux, but we may be involved in other important decisions such as selecting course management or learning management software. The formidable and proprietary offering from Blackboard (especially considering its recent purchase of competitor WebCT) is receiving noteworthy competition from open source management applications such as Sakai (sakaiproject.org) and Moodle (moodle.org). In the Spring 2008 edition of *Kairos*, Karen Lunsford, et al. report on initial findings from a collaborative effort at the University of California, Santa Barbara between the Writing Program and library to pilot different open source course management tools. Their research began in August of 2006, and the team has recently selected Moodle. While Lunsford has indicated the initial costs related to the software itself will be relatively inexpensive, the costs associated with employing a systems administrator and maintaining the software are not (personal communication, Aug. 31, 2008).

Looking at initial costs alone can be deceptive and tempting. Since some faculty may have control over what operating systems and software are loaded onto computers in a writing lab and their own machines, many have seen first-hand that pricing software becomes expensive quickly. Even at the education rate, Microsoft Office 2007 retails for \$200 (Microsoft U.S. academic pricing, 2007). While that figure represents a 50% discount from the price in a store, purchasing multiple copies or licenses can quickly mean thousands. Embedded in that cost are degrees of stability, interoperability, and support. In short, proprietary applications often equate to lower risk. I believe that assessing that degree of risk should be among the primary criteria for evaluating free and open source software. Open source applications are only as good as their contributors. The stability and the long-term viability depend on a community of active developers.

There are plenty of open source initiatives that have begun with a flurry of activity only to see developers defect to another project. The good news is that there are free and open source applications that the education community can rely on. In its March 2008 issue, PC Magazine published an article documenting "The Best Free Software" (Griffith). In its elite and short "Hall of Fame" category were both OpenOffice and the Gnu Image Manipulation Program (GIMP).

Installing OpenOffice, a no cost, open source suit of tools that includes a word processor, spreadsheet program, presentation manager, and drawing program, is an excellent, low-risk solution. OpenOffice supports numerous file formats from commercial companies like Microsoft and also supports a standardized open source format known as OpenDocument. Likewise, in writing courses such as technical communication or professional writing and editing, where document design and writing for the web are often part of the curriculum, the well-known proprietary applications from Adobe such as Photoshop and Dreamweaver are offered at budget-breaking prices. At the end of 2007, the cost to outfit two new writing classrooms at my university with Adobe's Design Premium Packages at the educational rate would have required approximately \$18,720. Instead, the 48 computers run GIMP and Sea Monkey (comprised of the older Netscape Composer code base) in place of Photoshop and Dreamweaver respectively. These more stable applications provide instructors the same opportunity to teach fundamental design principles and concepts from the field of visual rhetoric such as proximity, alignment, repetition, and contrast. As noted in the NCTE/CCCC proposal, these new applications require an additional time commitment to learn. However, from a purely functional perspective, many of the interface features from the open source community mirror those of proprietary applications.

It is my claim that by engaging with the open source community, instructors have an opportunity to broaden their knowledge of the discourses involved and achieve a more nuanced understanding of the open source and proprietary polemic. This includes researching not just the upfront costs associated with free and open source software but determining the level of risk associated with implementing that software. More than just abstract discussion, a selective set of open source technologies has matured to the point that they are stable enough to introduce in a classroom setting. Some of the major tools that we tend to rely on from proprietary companies such as word processors, spreadsheets, web browsers, graphics editors, and web editors have been developed from within the community and are available for download at no cost. Regardless of the actual monetary savings (if there are any at all), the return on investment from integrating open source in the classroom should be the success of our ability to implement new critical pedagogy and not dollars saved.

Knowledge Creation & Dissemination

Equally challenging and equally provocative is the second assumption from NCTE/CCCC proposal's statement regarding knowledge sharing that also requires closer scrutiny: "The open source development model parallels the academic model of knowledge creation and dissemination." The remaining text from the proposal to support the claim reads:

Open source software development is built within communities of committed users who are dedicated to maintaining and improving the software. Program code is shared and collectively reviewed by multiple parties, and collaboration drives related tasks such as promotion, documentation, usability testing, and support. This process resembles the ways in which knowledge is constantly made, refined, and re-thought in universities. (Benefits of Open Source section, para.

5)

This description implies that there is another, perhaps less efficient or even undesirable development model that stands in opposition to the open source process. The opposition, or in other words the corporate model,

is much more complex than it seems and requires our attention. In *The Cathedral and the Bazaar* (2001), Eric Raymond, a well-known open source developer and advocate, juxtaposes the title's two metaphors as representations of competing development models. Prior to his witnessing Linus Torvalds, the progenitor of Linux, manage the development of his budding operating system, Raymond "believed that the most important software needed to be built like cathedrals, carefully crafted by individual wizards or small bands of mages working in splendid isolation, with no beta released before its time" (p. 21). In contrast, Linux was a cacophony of voices offering innovative ideas to an ever growing and collective knowledge base all centered on bettering the operating system. Raymond explains:

Linus Torvalds' style of development – release early and release often, delegate everything you can, be open to the point of promiscuity – came as a surprise. No quiet, reverent cathedral building here – rather, the Linux community seemed to resemble a great babbling bazaar of differing agendas and approaches out of which a coherent and stable system could seemingly emerge only by a succession of miracles. (2001, pp. 21-22)

Circulating freely amidst all the babble is the source code for Linux. What would normally be the closely guarded intellectual property of cathedral builders like Microsoft and Adobe may be downloaded and modified by anyone. The effect is an enormous demystification, decryption, and distribution of the language of the sages once thought unintelligible even if it were in a user's possession.

However, anyone who has ever written, shared, and debated over computer code (and the functionality of an application which it controls) within an online community of developers knows not to dream of utopian lands of equality existing in digital spaces. Even Linux has a hierarchy. While Torvalds ultimately has the final say as to what code makes its way into the Linux kernel, the core of the operating system, he has approximately a dozen "deputies" or code "maintainers" who advise him on selections (Rivlin, 2003, p. 156). As a virtual manager of thousands of code contributors, Torvalds inevitably must reject some of these contributions. He is described as "less of a ruler (or a hood ornament for that matter) than an ambassador, roaming his virtual world and exerting his influence to prevent technical fights from devolving into sectarian battles" (Rivlin, 2003, p. 157). As good of a job as Torvalds does of managing and placating his many enthusiastic contributors, he struggles with some of the same issues as the so-called cathedral model of development. Mainly, "[t]hey don't have the institutional resources to ensure that a programmer isn't guilty of plagiarism" (Rivlin, 2003, p. 156).

Even beyond issues of resources, the neat and clean binary of open versus closed development has already been dismantled by scholars like Michael Truscello (2003) who remarked that the "reductive rhetoric" of Raymond's argument "about the natural evolution of 'free markets' is especially problematic for those who profit from the Internet because it was originally established and developed by the government" (para. 12). Between the cathedral and bazaar polemic, Truscello (2003) identifies a "naturalistic fallacy at work here, a troublesome tendency to say the Open Source model is more 'natural' — and therefore better — than closed source" (para. 12). The binary between the cathedral and bazaar is even harder to maintain now that so many commercial vendors have taken cues from the open source community creating a blend of development practices. For example, Scott Guthrie is a General Manager within the Microsoft Developer Division and in January of 2008, he announced (and provided links) on his blog that Microsoft would be making the source code for its .NET Framework Library available to the public. The idea is of course to engage with the enormous community of developers using Microsoft products to write and test code and to keep that community using Microsoft products.

There is not, incidentally, two separate populations of programmers – one open and one closed. According to Red Hat's Vice President of Open Source Affairs, Michael Tiemann, "vastly more than 90%" of open source software developers are employed developing proprietary code (2005, p. 16). While companies do

their best to keep targeted intellectual property "company confidential," even a strict non-disclosure agreement or non-compete agreement signed at the time of hiring will not guarantee ideas do not circulate.

From my own experiences as a software engineer for a major commercial company, the mission to protect intellectual property did not stop engineers from circulating ideas both inside and outside of the company. Inside, before a major section of code was "checked-in" as an official component of a new application, the author and his or her code endured a series of code reviews performed by colleagues within the company. This process would feel very much like a peer review editing process in a writing workshop. Even though the code compiled and ran without errors did not mean that it would not be revised for elegance and efficiency. In order to connect beyond the confines of company walls, funding was available for additional training and travel to workshops and tradeshow to learn and share ideas with other developers. Sharing, of course, did not require leaving the building. Most of my colleagues and I belonged to development forums (much like the Writing Program Administrators or Association of Teachers of Technical Writing list-serves, for example) where programmers from around the world would post problems and assist each other with solutions.

Whether or not a proprietary business model can survive the changes and advances brought by the open source community is another question altogether. Piling on additional protections by way of modifying copyright law to protect proprietary business models should not be the answer. However, if proprietary companies wish to continue to obfuscate their code they should not be denied that right. That is, they should have the right to succeed or fail with that strategy in the marketplace. It seems apparent from the actions of Microsoft and others that companies producing proprietary software recognize the danger in not making adjustments to their old ways of doing business and interacting with users. Even for a proprietary company, the better business model may include opening at least some traditionally protected property. Like Google, companies such as IBM and Hewlett-Packard are investing in the exploration of how Linux and open source ideologies can best merge with their business operations. Martin Fink is the General Manager for Hewlett-Packard's Linux Systems Division and his book, *The Business and Economics of Linux and Open Source* (2003), displays some of the optimism of the NCTE/CCCC proposal for adopting open source but is tempered by warnings over intellectual property:

This new software development methodology and licensing model represents one of the newest opportunities and one of the greatest threats to your organization. It presents opportunity since it gives you new ways to lower costs and create competitive advantage. Open source also creates a threat by exposing your business to serious intellectual property risk. You need to care about open source because the odds are quite good that many developers within your organization are already using open source software. The odds are not as good that these same developers fully understand the risks to your company. (p. 35)

The "opening" of intellectual property, while cast by Fink as a potential "threat" in a corporate setting, may appear from the perspective of the humanities as an unassailably logical approach to how property should be treated. From a corporate perspective, whether that company is producing code or engineering car parts, property has a tradition of being managed much differently. Evidence as to why companies can be so protective of intellectual property is offered in Adam Jolly and Jeremy Philpott's edited collection on how to best leverage it for commercial gain titled, *A Handbook of Intellectual Property Management: Protecting, Developing and Exploiting Your IP Assets* (2004). The lead article by a corporate CEO begins: "Today, intellectual property (IP) underpins between 50 to 70 per cent of a country's private sector gross domestic product (GDP), so it is often the difference between commercial success and failure" (Harvey, p. 3). With the stakes being so high, companies not only hire their own legal teams but join consortiums like the RIAA and MPAA to litigate against file sharers distributing and downloading copyrighted work. These groups actively and often successfully lobby Congress to increase copyright controls. The concern is that the resulting laws, such as the Digital Millennium Copyright Act or DMCA and the Technology, Education,

Harmonization Act or TEACH Act, restrict the use of copyrighted material to the point of impeding First Amendment rights and "chilling" free speech (Porter, 2005, p. 244). Another powerful but perhaps lesser known group is the Software and Information Industry Association (SIIA) that represents the "software and digital information industry." On its "Membership Benefits" web page, the SIIA warns: "Content piracy is still in its infancy and SIIA is at the forefront of combating this growing problem" (para. 6). As one of its combat strategies, the SIIA offers rewards up to \$1 million to anyone reporting a piracy act such as selling illegally copied software for a profit. In August of 2007, SIIA settled a lawsuit with a marketing firm, Knowledge Networks, for content piracy for \$300,000. According to the SIIA's press release, Knowledge Networks distributed "press packets" to their employees and these packets "sometimes included — without authorization or license — copyrighted articles owned by SIIA members such as the Associated Press, Reed Elsevier, and United Press International" (SIIA announces). Humanities professors and administrators should note that among the other names of SIIA's "Participating Companies" is McGraw-Hill.

Conclusion

These actions do, of course, contribute to the fabrication of only a rigid binary existing between open source and proprietary communities and their respective ideologies. While there is no shortage of corporate business models that plan to do everything they can to protect their intellectual property, open source is spurring many businesses to rethink their strategies. Fink has a message that while geared toward corporations, I believe can prove equally true for universities. He writes: "Integrating an open source mindset within your company will be a complex and difficult task, but it will be an accomplishment that will bring many rewards if you do it right" (2003, p. 135).

In reexamining the binary opposition between open source and proprietary communities, I have argued that there is inconclusive evidence to support total cost of ownership reductions in a migration to open source software and that there are significant "costs" to writing instructors in terms of learning and being able to effectively teach with the new applications. While the NCTE/CCCC proposal on the adoption for open source makes a strong claim for the cost-reducing potential of implementing non-proprietary applications, that claim will be difficult to defend as a blanket statement until more evidence and case studies from researchers such as Lunsford are provided from the academic community. I believe that the true benefit and the measurement of success should be weighed in our ability to use open source to implement new critical pedagogy and not necessarily dollars saved. As a former engineer, I not only enjoy the WID classroom, but I believe in its ability to help prepare students to become members of the discourse communities tied to their disciplines. As writing instructors, however, we need to acknowledge and respect when our disciplinary values pertaining to intellectual property may conflict with other disciplines.

Finally, it may be that Raymond's metaphor of the babbling bazaar is still the most useful for the current academic climate and the academy's growing interaction with the open source community and its ideals. Again, we will need to recognize and openly address (both in and out of the classroom) the multitude of discourses and not just those we imagine will meet our immediate needs. Students in WID classrooms in particular stand to be done a disservice by being cut off from their disciplinary community's more complex stance on intellectual property. Willfully discarding the corporate discourses, especially as they begin morph and borrow from open source ideologies, is in effect hypocrisy. Our approaches and our practices shape students. Encouraging an ongoing dynamic of admitting as many voices as there are that wish to speak in the bazaar of an enormous open source community (even those voices that wish to continue to speak out for control of their intellectual property) will not only foster the critical literacy we strive for in the classroom, but will be the most productive as we attempt to understand open source and its potentials to reduce our technology costs.

References

- American Society of Mechanical Engineers (2001). *Intellectual property: A guide for engineers*. New York: ASME Press.
- Carpenter, J. H., & Krest, M. (2001). It's about the science: Students writing and thinking about data in a scientific writing course. *Language and learning across the disciplines*, 5(2), 46–65.
- Creative Commons. Share, reuse, remix—legally. Retrieved December 27, 2007, from <http://creativecommons.org>
- DiDio, Laura. (April 2005). Linux vs. Windows: Total cost of ownership survey. *Institute for advanced professional studies*. Retrieved January 13, 2008, from <http://www.iaps.com/Linux-Windows-TCO-Survey-2005.04.html>
- Ellsworth, Elizabeth. (1992). Why doesn't this feel empowering? Working through the repressive myths of critical pedagogy. In Carmen Luke & Jennifer Gore (Eds.), *Feminisms and critical pedagogy* (pp. 90-119). New York: Routledge.
- Fink, Martin. (2003). *The Business and economics of Linux and open source*. Upper Saddle River, NJ: Prentice-Hall.
- GNU General Public License. (June 29, 2007). *GNU project*. Retrieved January 15, 2008, from <http://www.gnu.org/copyleft/gpl.html>
- Gomes, Lee. Just how far behind is Linux? Depends on where you start. *The Wall Street Journal*. 17 Oct. 2007: Portals. B1.
- Griffith, Eric. (March 2008). The best free software. *PC Magazine*, 27(4), 78-85.
- Gurak, Laura J. (2005). Ethics and technical communication in a digital age. In Carol Lipson & Michael Day (Eds.), *Technical communication and the world wide web* (pp. 209-221). New Mahwah, NJ: Lawrence Erlbaum.
- Guthrie, Scott. (Jan. 16, 2008). .NET framework library source code now available. *ScottGu's Blog*. Retrieved January 18, 2008, from <http://weblogs.asp.net/scottgu/archive/2008/01/16/net-framework-library-source-code-now-available.aspx>
- Harvey, Ian. (2004). Why intellectual property matters. In Adam Jolly & Jeremy Philpott (Eds), *A handbook of intellectual property management: Protecting, developing and exploiting your IP assets* (3-6). London: Kogan Page.
- Lai, Eric. (Jan. 11, 2008). IE still top dog over Firefox in corporate browser kennel. *Computerworld*. Retrieved January 18, 2008, from <http://www.computerworld.com/action/article.do?command=printArticleBasic&articleId=9056780>
- Lessig, Lawrence. (Dec. 31, 2007). Commons misunderstandings: ASCAP on Creative Commons. *Lessig 2.0*. Retrieved January 13, 2008, from http://lessig.org/blog/2007/12/commons_misunderstandings_asca.html
- Lessig, Lawrence. (2004). *Free culture: The nature and future of creativity*. New York: Penguin Books.
- Lunsford, Karen, et al. (2008). The library and the CMS: Establishing library presence in Sakai writing course sites." *Kairos*, 12(2). Retrieved Jan. 18, 2008, from <http://kairos.technorhetoric.net/12.2/binder.html?praxis/lunsford/index.html>
- Lunsford, Karen. (personal communication, August 31, 2008)
- McLeod, Susan & Miraglia, Eric.(2001). Writing across the curriculum in a time of change. In S. McLeod, E. Miraglia, M. Soven, & C. Thaiss (Eds.), *WAC for the new millennium: Strategies for continuing writing across the curriculum programs*. (pp. 1-27). Urbana, IL: NCTE.
- Membership benefits. (2008). *Software & information industry association*. Retrieved January 16, 2008, from <http://www.siiia.net/membership/benefits.asp>
- Microsoft U.S. academic retail pricing. (June 6, 2007). *Microsoft*. Retrieved January 13, 2008, from <http://www.microsoft.com/Education/usacademicpricing.mspx>
- Nard, Craig, Barnes, David & Madison, Michael. (2006). *The law of intellectual property*. New York: Aspen Publishers.
- The open source definition. (July 7, 2006). *Open source initiative*. Retrieved December 9, 2008, from <http://opensource.org/docs/osd>
- Porter, James. (2005). The chilling of digital information: Technical communicators as public advocates. In Carol Lipson & Michael Day (Eds.), *Technical communication and the world wide web* (pp. 243-259). New Mahwah, NJ: Lawrence Erlbaum.

- Posner, Richard. (2002). The law & economics of intellectual property. *Daedalus*. Spring, 5-12.
- Postman, Neil. (1995). *The end of education: Redefining the value of school*. New York: Knopf.
- Proposed NCTE/CCCC resolution on the adoption and use of open source software. (March 3, 2007). *Kairosnews: A weblog for discussing rhetoric, technology and pedagogy*. Retrieved January 10, 2008, from <http://kairosnews.org/ncte-cccc-resolution-open-source>
- Raymond, Eric S. (2001). *The cathedral and the bazaar: Musings on Linux and open source by an accidental revolutionary*. Sebastopol, CA: O'Reilly.
- Rivlin, Gary. (November 2003). Leader of the free world. *Wired*, 11(11), 152-157, 206-208.
- Selber, Stuart A. (2004). *Multiliteracies for digital age*. Carbondale, IL: Southern Illinois Univ. Press.
- SIIA announces first content piracy case settlement. (Aug. 16, 2007). *Software & information industry association*. Retrieved December 29, 2007, from http://www.siaa.net/press/releases/Antipiracy_KN-settlement-release.pdf
- SIIA membership benefits. (2008). *Software & information industry association*. Retrieved December 9, 2008, from <http://www.siaa.net/membership/benefits.asp>
- Tiemann, Michael. (June 2, 2005). Open source defined. *Red Hat*. Retrieved January 11, 2008, from http://www.redhat.com/magazine/009jul05/features/summit/009_tiemann-slides.pdf
- Truscello, Michael. (May 2003). The architecture of information: Open source software and tactical poststructuralist anarchism. *Postmodern Culture*, 13(3). Retrieved, January 13, 2008, from <http://www3.iath.virginia.edu/pmc/issue.503/13.3truscello.html>
- Understanding copyright and related rights. *World Intellectual Property Organization*. Retrieved January 14, 2008, from http://www.wipo.int/freepublications/en/intproperty/909/wipo_pub_909.pdf
- U. S. Constitution, Art. I, § 8.
- Weber, Steven. (2004). *The Success of Open Source*. Cambridge, MA: Harvard University Press.

Contact Information

Brian D. Ballentine
Assistant Professor
Coordinator, Professional Writing & Editing
English Department
West Virginia University
Colson Hall, 1503 University Ave.
Morgantown, WV 26506
Email: brian.ballentine@mail.wvu.edu

Complete APA Citation

Ballentine, Brian D. (2009, January 19). Writing in the disciplines versus corporate workplaces: On the importance of conflicting disciplinary discourses in the open source movement and the value of intellectual property. [Special issue on Writing Technologies and Writing Across the Curriculum] *Across the Disciplines*, 6. Retrieved from <https://wac.colostate.edu/docs/atd/technologies/ballentine.pdf>