CHAPTER 5

Resisting Incorporation and Reclaiming the Commons: The Case of Oracle and Sun Microsystems

The previous two chapters focused on case studies of Microsoft and Red Hat, and discussed the ways in which the *processes* and *products* of FLOSS production became incorporated into capitalist production.[33] The chapter on Microsoft demonstrated how the company initially built its business model on strong protection of its intellectual property and fended off challenges from the emergent open-source models that proved to be an effective and efficient model of software production. Microsoft eventually shifted to embrace open source, albeit only in certain limited ways. The chapter on Red Hat demonstrated how free software could be transformed into a profitable business model by harnessing the labour power of the free software community and transforming its productive activity into commodities that could be customised, sold, and serviced for its customers. Furthermore, the chapter focused on the specific ways in which Red Hat negotiated its relationship with its free software project, Fedora, through the boundary organisation of the Fedora Project Council. This chapter will look at how a community of FLOSS developers deals with unwanted corporate encroachment into its community governance model. In other words, this chapter focuses on the *politics* involved in negotiating the boundaries between FLOSS communities and corporations. The focus on politics here is not only concerned with the governance structures in place for negotiating boundaries between the corporation and the FLOSS community, as was discussed in the previous chapter. Rather, the focus on politics here also specifically investigates the ways in which FLOSS communities can assert their interests against unwanted corporate attempts to influence production within the community. As such, politics here has the dual meaning of collective action as well as an ethical horizon toward which collective action can be directed.

How to cite this book chapter:
Birkinbine B. J. 2020. *Incorporating the Digital Commons: Corporate Involvement in Free and Open Source Software*. Pp. 89–100. London: University of Westminster Press. DOI: https://doi.org/10.16997/book39.e.

This framing of politics, then, focuses on both the moral economy (Thompson, 1971) of the FLOSS community but also the specific tactics used in resisting unwanted corporate influence.

To do so, I focus on one of the largest software companies in the world, the Oracle Corporation (simply 'Oracle' hereafter), and its acquisition of Sun Microsystems (simply 'Sun' hereafter). Whereas Sun maintained a good relationship with the open source community by sponsoring various projects and allowing those projects to enjoy relative creative autonomy, those relations became strained after Oracle acquired Sun in 2010. After the acquisition, Oracle used a different strategy toward Sun's open source projects. In certain cases, Oracle ended open source activities, in others it tried to influence open source development to meet its own goals, and in others again it altered the way that the project was governed. In response, the community employed different strategies to protect their commons-based resources.

In this chapter, I focus on the histories of three such projects: the OpenSolaris operating system, the MySQL relational database management system, and the OpenOffice productivity software that was designed as an alternative to Microsoft Office. Throughout the chapter, I focus on the ways that the FLOSS community maintains a unique ability to leverage its collective labour power against corporate encroachment into its projects by using technical, legal, and governance strategies that allow them to abandon a project without losing the products of their labour. This has a similar effect to a factory walk-out, whereby workers halt the productive process by abandoning the site of production. When dealing with software, however, production is not reliant on a particular space. Rather, productive activity can simply be moved to a new location. And, because of the unique legal institutions and technical features of open source software, a project can be 'forked' whereby the project can be copied and production can continue under a new name without violating the intellectual property protections of the original project. As we will see, this is one of the primary ways that the FLOSS community leverages its collective labour power against undue corporate influence.

5.1. The Oracle Corporation and Sun Microsystems

Oracle Corporation is one of the largest software companies in the world. The company has three main operating segments: cloud and licence business, hardware, and services.[34] From these Oracle earns approximately 82% of its total revenue from the cloud and licence business segment. In 2018 alone, the company earned more than $39 billion in total revenues and employed approximately 137,000 people. If calculated by total revenues, Oracle is the third largest company in the global software market behind only IBM and Microsoft. Oracle has remained competitive within the global software market, in part, because of its strategic acquisitions. One of the company's largest acquisitions took place

when it acquired Sun Microsystems in 2010. While the company's net profits dipped in 2001 after the dot-com bubble burst, the company has enjoyed a steady rise in profits since that time, with a noticeable spike in profits between 2010 and 2013. As such, the company's profitability can be directly tied to its acquisition of Sun Microsystems.

Prior to its acquisition by Oracle in 2010, Sun Microsystems provided network computing infrastructure solutions, which included software, systems, storage, and microelectronics. In 2009, the final year of its independent operation, Sun reported approximately $11.45 billion in revenues and employed approximately 29,000 employees in more than 100 different countries. The lion's share of the company's revenues (42%) came from its Systems operating segment, which included the sale of servers that provide computing and storage power to customers as a key part of Internet infrastructure. The other core brands owned by Sun Microsystems were the Java technology platform, the Solaris Operating System, MySQL database management software, Sun StorageTek storage solutions and the UltraSPARC processor. Because the company relied on the provision of infrastructure-based services and products, the company was a large supporter of interoperability. Interoperability, here, is simply defined as the ability for different programs to exchange data with one another by using common formats. To facilitate innovation and interoperability, Sun made its key intellectual properties freely available as a strategy to support open standards, open interfaces, and open source software. By making a commitment to open source, Sun was viewed favourably by the open source community and maintained a relatively good relationship with the community because it was transparent about its corporate goals. To better understand the reasons for Sun open-sourcing some of their key intellectual properties, we need to consider some of the historical development for corporate involvement in FLOSS projects.

5.1.1. A Brief History of the Market for Operating Systems

Throughout the 1980s, the market for operating systems was dominated by proprietary versions of Unix-based operating systems. For example, Hewlett Packard offered HPUX, IBM offered AIX, and Sun Microsystems offered SunOS. These operating systems dominated high computing, or infrastructural level computing, while the consumer market was dominated by Microsoft DOS, which was not based on Unix but developed entirely by Microsoft. Importantly, the proprietary Unix-based systems were source-incompatible. In effect, although these systems were all based on Unix, the development of separate proprietary versions had caused the code to diverge in such a way that programmers could no longer assume interoperability between the systems. As a result, programmers had to maintain separate code bases for each system, and companies could sell entire stacks of software to their customers who had to accept the entire stack. This resulted in an inefficient system that

was dominated by proprietary software vendors, while simultaneously increasing the workload for programmers. During the mid-1980s, however, the Free Software Foundation began as a response to the overly protective intellectual property restrictions placed on software. This, in turn, led to the development of free and open source software, which was collaboratively developed as a commons-based resource for others to study, use, adapt, or modify in any way.

Because this model of development was so successful, by the mid-1990s Linux, an open source operating system, had become the dominant Unix-like operating system. Linux undercut the competition by offering a comparable product at a significantly lower cost. Furthermore, because Linux is distributed under the GNU General Public License (GPL), an alternative form of intellectual property ('copyleft'), improvements to Linux could be shared by everyone, which improved its quality and stability. The proprietary companies could not compete with Linux because the commons-based peer production driving it constituted a larger labour force than any of the individual companies could employ. Rather than competing directly with Linux, certain proprietary companies began to open source their products as a way of joining forces with the free and open source software community. Sun Microsystems was one of those companies. Although Sun supported many different open source projects, I will focus on just three here. Sun open-sourced their Solaris operating system, which became OpenSolaris. They also open-sourced the MySQL database management software, as well as StarOffice, which became OpenOffice. As I mentioned earlier, Sun maintained a good relationship with the broader FLOSS community because of their commitment to and support for FLOSS projects. After the company was acquired by Oracle, this relationship was strained in certain ways. In what follows, I will discuss how the developers working on the three projects mentioned above – OpenSolaris, MySQL, and OpenOffice – strategically resisted the corporate acquisition.

5.1.2. OpenSolaris

In 1987, Sun Microsystems and AT&T announced that they were going to merge some of the most popular Unix-based operating systems into a single project. This project eventually became Solaris, which was a proprietary operating system held by Sun that contained both open-source and closed-source components. To attract interest in the project and build a community of users and developers around it, Sun Microsystems created OpenSolaris. OpenSolaris was an open-source version of the Solaris operating system, although it did contain some elements in its code that were not open source. After attracting a larger community of interest to the project, a Community Advisory Board (CAB) was created to direct it. The CAB served as a boundary organisation for negotiating boundaries between the OpenSolaris community and Sun. The

CAB was comprised of two Sun employees, two members who were elected by the broader community, and one member who was appointed by Sun from the broader free software community. In effect, most of the CAB members were connected with or appointed by Sun, and Sun made clear what its intentions were for the OpenSolaris project.

Sun's strategy for the OpenSolaris project was to incorporate some of the developments from OpenSolaris into their proprietary Solaris operating system. In turn, Sun could sell the proprietary version of Solaris to other enterprises. The money earned from sales of the Solaris project could then be used to support the developers and community involved in the OpenSolaris project. To facilitate this type of strategy, Sun protected OpenSolaris under a free software license created by the company called the Common Development and Distribution License (CDDL). This license enabled Sun to include proprietary, free software, or software protected under any other license in their Solaris and OpenSolaris operating systems. Consequently, Sun could use the OpenSolaris community as a way to drive development, quality control, or innovation that could be included in their proprietary Solaris offering. Importantly, however, Sun made this strategy very clear to the OpenSolaris community and was supportive of the broader FLOSS community, which gave it a good reputation within the community. Once they acquired Sun, Oracle took a very different approach to this strategy.

After Oracle acquired Sun, they announced plans to discontinue the regular distribution and development model of OpenSolaris (Laishram, 2010). Instead, Oracle would focus its development strategy on a new proprietary version of Solaris called Solaris Express. In effect, the new strategy from Oracle would not allow the community of developers that supported OpenSolaris to continue their work. In response, the Community Advisory Board directing the OpenSolaris project decided to fork the project. When a project is forked, developers take a copy of the source code and begin to develop it as a distinct form of software. The resulting fork of the OpenSolaris project is called OpenIndiana, which was created to continue the development and distribution of the OpenSolaris project. Currently, Oracle still continues development on the proprietary Solaris Express operating system, while the community of developers supporting OpenSolaris have left Oracle to work on the forked version of OpenSolaris called OpenIndiana.

In the case of the OpenSolaris operating system, Oracle's strategy was simply to discontinue the open source project and focus development on a proprietary version of Solaris under the new name Solaris Express. This represents the most direct strategy for ending open development. Oracle announced that the open source project would be discontinued and, in response, the community had to fork the project to continue development under a new name. This also illustrates how a FLOSS community can also continue working on a project even after production on a corporate-sponsored project was abandoned. This is a

similar fate to that of MySQL and OpenOffice, but Oracle's strategy for ending development took different forms in each case.

5.1.3. MySQL

In 2008, Sun Microsystems acquired MySQL AB for approximately $1 billion (PC World, 2008). At the time, MySQL was growing in the market for relational database management software (RDBMS), and Sun's acquisition of MySQL would allow the company to compete directly with Oracle in that particular market. Only one year later, however, Oracle acquired Sun, and MySQL was one of the key properties that drew Oracle's interest. Indeed, the Sun-Oracle merger was originally approved by regulators in the United States, but the European Union (EU) did not immediately approve the deal specifically because of concerns that Oracle's acquisition of the MySQL property would lead to an anti-competitive market for RDBMS in Europe (Bloomberg, 2013). Consequently, the EU pressured Oracle to divest itself of the MySQL property as a condition for approval of the merger. As leaked documents provided to the whistleblowing site WikiLeaks have since shown, the United States Department of Justice communicated directly with the European Commission's Directorate General for Competition in support of the merger in October of 2009 (United States Mission to European Union, 2009). Less than three months later, in December of 2009, the merger was approved without the divestiture conditions sought by the EU.

MySQL relied on a dual licensing approach that was similar to the licensing of OpenSolaris. The dual licence model for MySQL would allow the code base for MySQL to be protected by the GNU GPL copyleft licence, but proprietary versions could be created for enterprises that wanted customised installations. When the Sun-Oracle merger was approved, employees working for MySQL had reservations about Oracle's intentions for the GPL-protected code base of MySQL. Most notable among them was Michael 'Monty' Widenius who authored the original version of MySQL and co-founded MySQL AB, which was the original owner of MySQL. Widenius later sold MySQL AB to Sun before Sun was acquired by Oracle. Widenius along with other MySQL developers were concerned that Oracle would try to discontinue MySQL or make it a closed-source program by using the same strategy it had with OpenSolaris. In response, Widenius urged MySQL users to 'Help MySQL' by starting an online petition. Leading up to the acquisition of Sun, however, Oracle pledged to keep the same licensing strategies in place that had been negotiated with current customers for an additional five years (Whitney, 2009). That commitment expired in December of 2014.

Fuelled by the concerns about Oracle's intentions for MySQL, the developers forked the project to create MariaDB.[35] The code base for MariaDB is protected by the GNU GPL, and is designed to be a drop-in replacement for MySQL.

As a forked project of MySQL, MariaDB allows its community of developers and users to ensure that the code will continue to be protected by the GNU GPL regardless of what Oracle decides to do with MySQL. Furthermore, although MySQL remains dominant in the RDBMS market with an approximately 58% market share, MariaDB grew to claim approximately 18% of the market (Fydorenchyk, 2014). MariaDB has experienced increased growth in the database market in part because of some notable companies switching from MySQL to MariaDB, including Google and the Wikimedia Foundation.

MariaDB once again illustrates how the community of developers and users of open source software can protect their projects from unwanted corporate encroachment. In the case of MariaDB, the project has gained additional attention from some of Oracle's competitors who have invested directly in it. Most notably, SkySQL recently invested nearly $20 million to support the growth of MariaDB. Backed by capital from Intel and from other venture capital firms, SkySQL is directed by some of the founding members of MySQL as well as former Sun executives who left the company after Oracle acquired the project. SkySQL announced a merger with The Monty Program AB, which is led by Monty Widenius, the original author of MySQL. The merger reunites the original members of MySQL and transfers ownership of the MariaDB trademark to SkySQL. The resulting partnership will focus on developing MariaDB to compete with MySQL.

Furthermore, both the Monty Program AB and SkySQL belong to the MariaDB Foundation. The MariaDB Foundation is a non-stock, non-profit corporation, which was established to provide legal and technical support for the MariaDB project and to provide a platform for supporters to contribute money to the project. For example, the MariaDB Foundation sells corporate memberships ranging from $5,000 to $100,000. According to the Foundation's web site, corporate memberships allow for the 'best opportunity to influence the future and present a point of view', although no further details are provided about exactly what that entails (MariaDB Foundation, 2018).

In sum, MariaDB represents another example of how FLOSS communities maintain the ability to protect their commons-based resource against unwanted corporate influence. In this case, however, Oracle's strategy was not to discontinue the open source project, per se. Rather, Oracle's acquisition of Sun allowed the company to gain a greater share of the RDBMS market, and Sun's ownership of MySQL was one of the primary properties that attracted Oracle to acquire Sun. Although development of MySQL still continues under Oracle, many of the community members resigned from Sun, and Oracle's commitment to maintain the same licensing agreements for MySQL expired at the end of 2014. To resist what could ultimately have been a similar fate to that of OpenSolaris, the MySQL community forked the project to develop MariaDB. In this case, Oracle seemed to violate the moral economy of the FLOSS community, but the community coped with that unwanted influence by forking the project to continue development under better conditions. Again, this represents a

moment when the FLOSS community asserted a specific politics in protecting their working conditions; the community abandoned development on MySQL and moved to MariaDB. Furthermore, MariaDB has the additional benefit of having received investment capital from some of Oracle's competitors, which ensures the survival of the project for at least the foreseeable future. By establishing the MariaDB Foundation, the community has a legally recognisable organisation to provide technical and legal support for the project, while also collecting additional donations to the project. In the third and final example provided in this chapter, I focus on a series of office productivity software that eventually led to another forked project.

5.1.4. StarOffice, OpenOffice, LibreOffice

During the dot-com bubble in the mid- to late-1990s, Sun Microsystems experienced dramatic growth that allowed the company to make some key acquisitions. In 1999, Sun acquired the German company, StarDivision which developed StarOffice. StarOffice was designed as proprietary office software featuring word processing, spreadsheet, presentation, drawing, database, and formula programs. When Sun acquired StarDivision, the company continued to develop StarOffice as proprietary software. However, Sun forked the project and relicensed the software so that the source code could be made open source under a free and open source licence. Once again, Sun's strategy was to use the newly open-sourced software, known as OpenOffice, to develop new features and fix bugs in the software. Then, the changes made to OpenOffice could be integrated into StarOffice, which contained certain proprietary elements. OpenOffice could continue to remain free to consumers, while Sun would try to monetise StarOffice by selling the software and services to customers who wanted the additional features. The upshot for Sun was the maintenance and support for essentially two different versions of the same software: OpenOffice 1.0 was a forked version of StarOffice 6.0, and Sun maintained the legal rights to both properties, although they were protected by different licences.

The early versions of OpenOffice were protected by the Sun Industry Standards Source License (SISSL) and the GNU Lesser General Public License (GNU LGPL). Later versions were protected by an updated version of the LGPL after Sun discontinued the SISSL. The LGPL was chosen because it had less restrictive requirements for integrating free and open source software components into proprietary versions of the software. Although a full discussion of the distinctions between free and open source software licences is beyond the scope of this chapter, the basic differences between the GNU General Public License (GPL) and the GNU LGPL can be summarised quickly. The GPL requires that any modified or derivative software produced using GPL-protected software as its base must be redistributed under the same licensing requirements. This ensures that free software remains free software rather than being exploited

by commercial companies. The LGPL is a more permissive licence that allows free software elements to be incorporated into proprietary software. The only restriction on using LGPL-protected software is that the end-user must have the ability to modify the source code. By protecting OpenOffice in this way, Sun could ensure that developments in OpenOffice could be used in their proprietary StarOffice.

Thus, the symbiotic relationship between StarOffice and OpenOffice continued under Sun because Sun was transparent about what its intentions were for the two properties. Importantly, however, OpenOffice was governed by a Community Council comprised primarily of members from the broader OpenOffice community but also including a Sun employee as well. The Community Council effectively served as a boundary organisation (O'Mahony and Bechky, 2008) between the community and the corporation. The Sun member on the Community Council was responsible for communicating Sun's intentions to the community. Once again, however, this relationship was strained when Oracle acquired Sun in 2010.

Since Oracle had discontinued the OpenSolaris operating system, members of the OpenOffice Community Council decided to create The Document Foundation and fork the OpenOffice project under the name LibreOffice until Oracle made its intentions clear for the OpenOffice project. Both The Document Foundation and LibreOffice were established with the intention of being temporary projects until Oracle made its intentions clear. In the event that Oracle ultimately decided to discontinue OpenOffice, however, the Community Council would be able to move development to the newly created LibreOffice. Furthermore, The Document Foundation was established as a non-profit organisation to manage the LibreOffice project and promote the use of open source document software more broadly. The initial governance of The Document Foundation was directed by a temporary steering council featuring some of the same members of the OpenOffice Community Council. Oracle viewed the Community Council members' positions on two governing boards as a conflict of interest and asked members on the Community Council to step down from their positions (OpenOffice Community Council, 2010). This move effectively ended community support for OpenOffice and the project was renamed Oracle OpenOffice. Oracle OpenOffice became the proprietary software offering from Oracle that was meant to replace Sun's StarOffice.

While the official position of Oracle was to cite a conflict of interest, members of the broader open source community viewed Oracle's broader strategy as simply wanting to discontinue open source projects that existed under Sun because they did not provide any real value to the company. In effect, not only did the governance structure change under Oracle's ownership, but Oracle also seemed to have violated the moral economy (Thompson, 1971) of the FLOSS community. In response to this, however, The Document Foundation continued its development of LibreOffice. Since LibreOffice had strong community support, LibreOffice essentially surpassed OpenOffice within one release. In

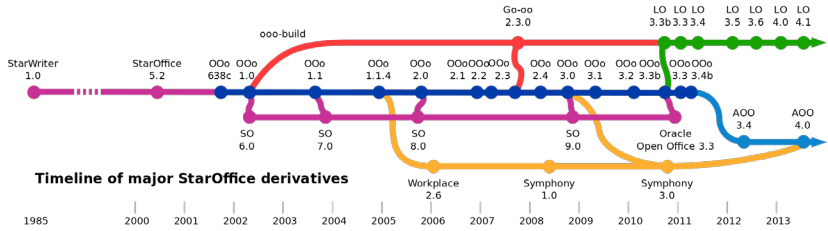

Figure 5.1: Major StarOffice Derivatives (image has been released to the public domain and is available from https://en.wikipedia.org/wiki/StarOffice#/media/File:StarOffice_major_derivatives.svg)

effect, all of the collective labour behind the development of OpenOffice abandoned the project but continued to work on LibreOffice. Because OpenOffice had been abandoned, Oracle announced that it would end development on the project entirely and fire the majority of OpenOffice developers. Ultimately, Oracle donated the code base for OpenOffice to The Apache Software Foundation, which has resumed development on the project under the name Apache OpenOffice.

To summarise this somewhat confusing history of a software that has been forked numerous times, Figure 5.1 illustrates the development history of StarOffice, its transition to OpenOffice (OOo) under Sun, the dual development of StarOffice (SO) alongside OpenOffice, the forks into LibreOffice (LO) and Oracle OpenOffice after Oracle acquired Sun in 2010, and the donation of OpenOffice back to The Apache Software Foundation to be developed as Apache OpenOffice (AOO). Figure 5.1 also includes additional forked projects that have not been discussed in this chapter, which include IBM Lotus Symphony (Symphony) and Go Open Office (Go-oo). As illustrated in the figure, the developments offer examples of how the FLOSS community uses legal, technical, and governance strategies to protect their commons-based resources.

5.2. Protecting the Commons

Throughout this chapter, I have demonstrated how the FLOSS community maintains the ability to leverage its collective labour power against undue corporate influence by employing technical, legal, and governance strategies to protect its commons-based resources. On the one hand, FLOSS has unique technical characteristics that allow it to be reproduced and distributed widely without any significant cost. This allows FLOSS projects to be forked so that development can occur collaboratively, simultaneously, and continuously throughout the life of the project. Although dispersed development occurs, however, the community employs certain governance strategies for effectively

coordinating development and protection of the project. These governance strategies include the establishment of non-profit organisations, which hold the intellectual properties for projects. These organisations provide a legally recognisable entity that can more effectively defend the intellectual property and licensing requirements of the project. Furthermore, more direct governance of the development project can occur through governing councils that are democratically elected or appointed by the community.

The legal strategies for defending FLOSS projects rely on alternative intellectual property protections like copyleft or other free and open source software licences. These licences free the software from overly protective copyright and allow the community to fork the project in the event of undue corporate influence. On the other hand, corporations can also use licensing strategies to their benefit as well. In the case of Sun, the company used licensing that allowed for free and open source software development but that was less restrictive to the corporation. These licences allowed the company to incorporate some of the commons-based peer production of FLOSS projects into their proprietary offerings. This strategy was understood and accepted by the FLOSS community because Sun was clear about its strategies but also because Sun supported FLOSS development projects. In a sense, then, licensing a project becomes a site of struggle, especially because a single project may contain code that is protected by different licences. These licences may have competing or conflicting terms that need to be resolved or the project becomes susceptible to intellectual property litigation. As was the case during Oracle's acquisition of Sun, the licences can be changed as a way to direct development toward different ends. Sun was transparent about its licensing strategies as a part of its broader commercial strategies, while Oracle made either temporary commitments to use existing licensing strategies (e.g. MySQL) or sought to change those licensing requirements altogether (e.g. OpenSolaris).

However, the dynamics that exist between FLOSS communities and corporations are comprised of a combination of technical, legal, and governance strategies. The particular forms that these strategies take will vary depending on the individual project, but the FLOSS community's ability to defend its commons-based resources depends, in part, on a shared consciousness of what is permissible within the community. In a sense, this shared consciousness constitutes a sort of moral economy (Thompson, 1971). The FLOSS community leverages its collective labour power against corporate power by protecting its commons-based resources. When a corporation infringes on the moral economy of the community, the community rebels by forking the project and abandoning the project that has been overly influenced by the corporation. This moral economy has foundations in the shared ideals of peer-to-peer relationship building, collaborative development, transparency, and community.

Even though the FLOSS community maintains the ability to leverage its power against undue corporate influence, community members are still in a somewhat precarious position as digital labourers. One definition of success

in open source projects is to receive backing from a company, which at least ensures the project's survival if not its overall attractiveness. However, the FLOSS community depends on keeping projects protected under free software licences, albeit of many different types, so that the community maintains the ability to keep the code for the program open. This is particularly true in cases where hybrid models of proprietary and free software are used in FLOSS projects. Throughout this paper, I have demonstrated how such struggles can occur, particularly after corporate mergers, acquisitions, or takeovers.

In the face of growing corporate involvement in FLOSS projects, the broader FLOSS community must maintain its ability to protect its commons-based resources. At the same time, however, the protection of these resources depends, at least in part, on a shared collective understanding of how the community can leverage its collective labour power against increasing corporate involvement. The lessons to be learned from Oracle's acquisition of Sun Microsystems need to remain salient if similar strategies are to be effective. Most important, however, is the recognition that the struggles taking place within the FLOSS community are just one part of a broader social struggle. As Christian Fuchs (2008) has observed, commons-based production is not truly possible until we have a commons-based society. Until that time, commons-based movements like FLOSS will be subjected to increasing corporate encroachment that threatens to abate, assimilate, or altogether annihilate progress toward alternative economic configurations.

Notes

[33] An earlier version of this chapter appeared as Benjamin Birkinbine 2016b. Conflict in the Commons: Toward a Political Economy of Corporate Involvement in Free and Open Source Software. *The Political Economy of Communication* 2(2): 3–19.

[34] Unless otherwise noted, all of this information was derived from Oracle's annual filings (Form 10-K) with the Securities and Exchange Commission (SEC) of the United States, which is available here: https://investor.oracle.com/financial-reporting/sec-filings/default.aspx (last accessed 2 January 2019)

[35] MariaDB is just one fork of the MySQL project. Percona Server is another that is still actively developed as of the time of writing.