



## Software Freedom Law Center

Eben Moglen  
212-461-1901  
moglen@softwarefreedom.org

November 19, 2009

1995 Broadway, 17th Floor  
New York, NY 10023-5882

tel +1-212-580-0800  
fax +1-212-580-0898

[www.softwarefreedom.org](http://www.softwarefreedom.org)

## Introduction

Professor Eben Moglen of Columbia University Law School, Founding Director of the Software Freedom Law Center, both of New York City, respectfully declares:

## Qualifications

1. I am a member in good standing of the Bar of the State of New York in the United States, first admitted to practice there in 1988. I have been a teacher at Columbia Law School since 1987, Associate Professor of Law until 1994, and thereafter as a full Professor of Law. I have a JD from Yale University conferred in 1985, and an MA, an M.Phil. and a Ph.D. in History from the same University, conferred respectively in 1983, 1985 and 1993. Before teaching at Columbia I was law clerk to Judge Edward Weinfeld in the United States District Court for the Southern District of New York, and to Justice Thurgood Marshall of the United States Supreme Court.
2. I have been practicing the law of free software since 1993, when I first provided legal counsel to Mr Richard Stallman and the Free Software Foundation. I was General Counsel of the Foundation from 1995 until 2005, and a member of its Board of Directors from 2000 to 2007.
3. In 2005, I formed the Software Freedom Law Center (“SFLC”), a non-profit legal services organization, established as a charitable corporation under the law of New York State and accorded tax-exempt status by the US Internal Revenue Service under section 501(c)(3) of the Internal Revenue Code. SFLC is the only organization of its kind in the world, completely dedicated to providing free legal counsel and representation to individuals and non-profit makers and distributors of free and open source software.

## Disclosures

4. SFLC’s mission is to represent, *pro bono publico*, those whose relation to free and open source software generates no revenue stream from which traditional legal representation and counsel can be paid. We are enabled in large part to carry out this mission by voluntary donations provided by commercial parties who benefit from free and open source software in their businesses, and who appreciate that all parties are better off when the people who make the software can make it safely. In addition to this support, we also receive donations from individuals and grant making organizations.
5. Oracle is a donor to SFLC, in an ongoing relationship of regular financial support. Oracle’s scale of support for SFLC is comparable to that of other major US and international information technology firms, including firms with a diversity of positions concerning the proposed acquisition of Sun Microsystems. SFLC

has also received substantial personal financial support from Mr Michael (Monty) Widenius. Neither the funding from Oracle nor the funding from Mr Widenius has exceeded 5% of our total funding since SFLC's inception. I have known Mr Widenius since 2001, when I assisted him and the other principals in MySQL AB in connection with legal matters in the United States, including the NuSphere (Progress Software) v. MySQL litigation, in which I appeared on MySQL's behalf as an expert witness and submitted a declaration to the Court. I severed my relationship with MySQL AB in 2005, when I withdrew from all commercial representations as part of the process of SFLC's formation.

6. I am submitting this Comment and Opinion to the Commission at the invitation of Oracle's counsel, but I am not receiving any compensation, fee or reward for so doing. I represent the interest of no party before the Commission. My goal is simply to offer some observations on issues raised by the present inquiry about which I believe myself to have uniquely specialized knowledge and experience.

## General Comments on GPL and Competition

7. The Commission's Statement of Objections (hereafter "SO"), only a part of which was furnished to me, depends upon an assumption, unquestioned here, that MySQL under GPLv2 provides a significant competitive constraint on Oracle's pricing and non-price commercial distribution terms for its flagship database product, despite very significant technical and enterprise performance differences between MySQL and Oracle 11. The Commission expresses concern that although everyone wishing to support, sell or improve MySQL has unquestioned legal right to do so under GPLv2, Oracle could use its power of centralized copyright ownership in the current MySQL codebase to lift that competitive constraint in ways that could not be timely or sufficiently resisted by free software developers working under the terms of GPLv2.
8. I have no knowledge of Oracle's business plans or intentions, and I have no comment on the Commission's assumption that Oracle experiences "competitive constraint" as a result of MySQL's existence as free software. I nonetheless believe that the Commission's analysis overlooks both general and specific reasons why acquisition of existing centralized copyright in MySQL is not in fact the threat to competition that the Commission is concerned about.
9. GPLv2 and GPLv3 are copyright licenses engineered by their makers to ensure the free flow of ideas and improvements to software *regardless of who holds the copyrights*. The underlying purpose of the GNU General Public License (which we now all abbreviate GPL) was to enable diverse developers to collaborate on the long-term construction of an entire computer operating environment (which Richard Stallman dubbed "GNU") that would preserve its own freedom no matter who came into ownership of the code from a legal point of view, and no matter what their commercial motives were with respect to it.
10. The "copyleft" properties of GPLv2 and GPLv3 have proved themselves enormously robust over the last twenty years. Since the promulgation of GPLv2 in 1991, the technical, commercial and development environment surrounding GPL'd software has changed profoundly. Firms with business models directly hostile to the "free software" idea pioneered by Richard Stallman have directed substantial adversary attention to the GPL. GPL'd programs competing effectively against offerings of the richest and most powerful monopoly in the history of information technology have resisted the efforts of the monopolist to find a chink in its armor.

At the same time, the copyleft's friendliness to large-firm business models has attracted other IT giants such as IBM and Hewlett-Packard to the value of supporting the development and distribution of projects that cannot be deployed with proprietary improvements or restrictions, sharable by all users and improvers.

11. MySQL is now and always has been an atypical GPL software project, because its copyright was highly centralized inside a small commercial firm that considered dual licensing its only commercially attractive strategy for survival. But even MySQL AB's atypical business model, which was highly unreflective of the mass of GPL'd software development, occurred within the parameters of the GPL's overall design, which is to ensure the freedom of the software it protects regardless of the commercial motivations or behaviors of the parties distributing the primary upstream version.
12. The Commission's analysis underestimates the distinctive operation of GPL's copyleft under market conditions, arriving at conclusions that insufficiently distinguish between the real-world behavior of GPL'd software communities and those operating under "permissive" open source licenses such as BSD or Apache.
13. The SO contains, for example, substantial analysis of the history and commercial prospects of PostgreSQL as a possible competitive constraint on Oracle (714–728). The Commission observes that development resource for the commercial exploitation of PostgreSQL has been divided among numerous ventures. Projects under GPL, on the other hand, precisely because they do not permit proprietary enhancement, channel the investments of interested parties into an ecologically sounder pattern, in which technical innovation occurs throughout the project, rather than concentrating on the points of attachment for proprietary add-ons or feature modules.
14. For this reason, other basic infrastructure components of modern computer systems (operating system kernel, C compiler, shared file system protocol implementation, networking stacks, and web content management systems) used by industry and commerce have tended to emerge from the GPL'd commons, where both firms and non-firm actors have been able to collaborate in confidence that their improvements would not be proprietyized by other, more powerful or less ethically motivated parties. Linux, GCC, and Samba, with all of which the Commission is familiar from other investigations, are examples of the superior resistance of the GPL'd commons to fractionation of effort or misallocation of resources.
15. As another example, I think the SO (759) may be mistaken in asserting that only the permissive nature of the Apache Software License enables the close collaboration between IBM and the Apache Software Foundation. IBM has also collaborated on equally close terms with projects and project clusters under GPL, including for example its leading role in the Free Software Foundation's GCC Steering Committee, which is backed by an effort of IBM's Linux Technology Center involving dozens of IBM programmers directly paid to work on the compiler and its toolchain; or IBM's very significant efforts to enhance and improve the utility of the Linux kernel for a variety of hardware, including the IBM S/390 mainframe architecture. In that last case, as early as 2000, IBM completed at great expense and solely using dedicated IBM resources, a port of the Linux kernel, C Library, and toolchain for this proprietary IBM architecture, and then donated all copyrights to the Free Software Foundation.
16. In general, we think of free software projects in ecological rather than purely commercial terms, because the breadth of possible contributors to technical innovation in free software far outstrips the number of venture capital-funded small

firms likely to play a role. Such firms *do* have an important role in the development of free software assets, but their economic health is a product of, rather than a producer of, the overall health of the commons to which they contribute.

## GPL and MySQL

17. For these reasons, the Commission’s analysis of the probable consequences of Oracle’s acquisition of the MySQL copyrights seems to me to overlook important elements in the model of competition appropriate to sketching the future of the commons at stake here. The Commission assumes that both the technical structure of MySQL itself and the structure of firms and non-firms contributing to its growth must be in the future more or less what it was in the past. But MySQL’s technical and commercial history is a product of the originating firm’s perception that it required dual license revenue, not a necessary condition of its future.
18. “Storage engines,” for example, are much discussed in the SO, under the impression that it is a virtue of MySQL that it is designed around replaceable storage engines, which the SO asserts can only be created and implemented by venture-funded firms needing proprietary distribution opportunities in order to recoup development costs. This, I think, is a misunderstanding of the technical history. MySQL’s design was an outcome of the need for dual-license revenue, not a precondition to it. PostgreSQL, for example, does not have or need multiple storage engines: it contains a single highly-configurable storage manager which is not independent of the database engine. MySQL’s use of multiple storage engines resulted from MySQL’s need to have something to sell dual-license customers different and better than that provided to users under GPL. The resulting uncertainty that could arise over whether a table had been created using a transaction-safe storage engine or by the original freely-available MyISAM storage engine contributed to the problem with occasional corruption of MySQL databases that was long a primary drawback to the enterprise use of MySQL. MySQL AB contemplated and designed a single multi-purpose high-performance storage engine, named Falcon, that would have substantially replaced all existing MySQL storage engines, but the business-model consequences of that step were sufficiently negative for MySQL AB, because of its need to produce dual-license revenue, that the project was never completed.
19. Similarly, MySQL experienced great difficulty in creating a vibrant ecology around the program, because the perceived need for dual-license revenue meant that MySQL AB believed it could not afford to accept *any* third-party code contributions unaccompanied by copyright assignment, because without 100% ownership of the copyrights in the base code, it could not provide OEM or “commercial” licenses to users who were frightened of or ignorant about GPL. By late in the independent history of MySQL AB, before the acquisition by Sun Microsystems, the firm was actively seeking to expand the scope of its unaffiliated developer community, but was hampered in doing so by the need to retain entire control over copyright in order to maintain the dual-licensing business.<sup>1</sup>
20. The SO raises (663) the question of third-party storage engines under non-GPL license being combined with MySQL under GPL. The SO states that “for third-party storage engines aimed at specific market segments a business model based on code under the GPL is often not sustainable.”

---

<sup>1</sup>Because, prior to the formation of the Software Freedom Law Center in 2005, I consulted with MySQL AB and its counsel on numerous legal matters, I cannot comment on the nature of those measures or the difficulties experienced, but there are substantial indications concerning this process in the public record. See <http://www.crn.com/it-channel/187000558>

21. In the narrowest sense, the SO's question can best be answered by saying that under limited circumstances, a non-GPL storage engine could be combined with MySQL by a dual-licensing commercial producer. Under GPLv2, distribution with MySQL of a non-free storage engine designed to work *only* with MySQL would be infringement of the MySQL copyright unless permitted by the copyright holder. But a storage engine that actually qualified as a separate and independent work—either because it worked with other database server engines or because it could perform useful work standing alone—could be distributed separately and combined with MySQL by end users without violating the license. See GPLv2 section 6. This would be a conceivable business model for a dual-licensing storage engine manufacturer. When distributed *along with* MySQL, only the GPL version of such a storage engine could be provided without violating the license, but separate and independent works separately distributed would not have to be licensed under GPL. Innobase could have explored such a business model itself, before its acquisition by Oracle. InnoDB could be used as a storage engine for other data, besides MySQL database tables, standing alone. Innobase could therefore have provided an enhanced version of InnoDB, like Oracle's current InnoDB+, under non-GPL license, for end users to replace on a plug-in basis in their MySQL installations.
22. The SO proceeds under the assumption that a non-dual-licensing model for the distribution of specialized storage engines would not be commercially viable for a small or nascent firm. But the diversity of GPL-surrounding business models actually observable in the market suggests that this assumption is hasty, and probably incorrect. GPL'd programs can support a diverse ecology of surrounding commercial enterprises. CodeSourcery, for example,<sup>2</sup> has since 2004 sustained itself commercially with specialized libraries and toolchain improvements to be used with GCC under GPL in a license compliant fashion that end-users combine with their GPL'd GCC installations to improve compiler performance for specialized hardware or in niche applications. A storage engine developed for MySQL to be used under GPL, accompanied by tuning tools, monitors and other supplementary software under non-free license would be an equivalent and entirely GPL-compatible model for a growing software house. A highly specialized storage engine meant for niche application use would generate not large-scale service obligations requiring high levels of personnel and large initial outlays, but rather highly precise expert consulting services, such as are added to the R statistical computing environment by firms such as REvolution Computing, XL Solutions, and Random Technologies. Niche delivery of specialized versions of R, still under GPL but at very high prices surrounded by tools enabling multiprocessing clustering of very large scale statistical computation by, inter alia, pharmaceutical manufacturers, have been discussed in the public press.<sup>3</sup> This too is a feasible non-dual licensing model for a storage engine producer under GPL capable of adding additional tools in niche environments.
23. These and other similar examples that could be provided broaden somewhat the available information bearing on the risks and rewards of supplementing MySQL under GPL for small or nascent software firms. The SO states (663) that “the one precedent in this regard” is NuSphere (Progress Software) v. MySQL. The SO concludes that “This precedent illustrates why any third-party provider of a storage engine for MySQL that does not exclusively rely on the GPL itself will be wary to conduct business without a commercial relationship with MySQL's owner.” But this conclusion is based on a misunderstanding of the factual setting of that dispute.

---

<sup>2</sup><http://www.codesourcery.com>

<sup>3</sup><http://www.nytimes.com/2009/01/07/technology/business-computing/07program.html>

24. I was directly involved in the dispute between NuSphere and MySQL.<sup>4</sup> Both parties approached the Free Software Foundation and me during the course of their dispute seeking our advice and support. I attempted to mediate the dispute, unsuccessfully. Ultimately, I determined that MySQL AB was in the right, and offered them legal assistance and expert testimony. I helped to arrange the settlement under which the Free Software Foundation audited the NuSphere and MySQL software offerings, made rulings on various GPL issues raised by both codebases, and brought both the parties into full GPL compliance. I speak about the facts of that situation on personal knowledge, and without reference to privileged or confidential material.
25. The SO incorrectly assumes that there was no commercial relationship between the parties. At the commencement of the dispute, NuSphere was MySQL AB's exclusive North American marketing partner. The origin of the dispute was a claim by MySQL AB for allegedly unpaid trademark royalties in excess of \$400,000 due from NuSphere. NuSphere sued MySQL AB alleging contractual defaults and denying liability for trademark royalties, and MySQL AB defensively brought to court its GPL counterclaim concerning NuSphere's proprietary storage engine. It was precisely the two firms' highly-entangled commercial relationship that made it impossible to reach a peaceful settlement of the GPL issues, which standing in isolation would have been easily resolved as I had resolved many similar disputes before and have resolved many since. Far from demonstrating, as the SO suggests, the pitfalls of arms'-length enhancement and redistribution of GPL'd code, NuSphere v. MySQL AB might more accurately be said to demonstrate the comparative legal fragility of dual-licensing business models compared to pure commons production.
26. For these reasons, I believe that the SO is far too pessimistic in its evaluation of the admittedly rather specialized example of the storage engine, which exists architecturally, as I have said, primarily in order to sustain dual licensing but which, owing to the flexibility and robustness of the GPL, could be turned to a number of business model templates that do not involve dual-licensing at all.
27. MySQL AB showed considerable ingenuity in the management of its precarious dual-license model, and was commercially successful, but it did not require proprietary storage engines to succeed. Distribution of InnoDB under GPL provided MySQL a transaction-safe storage engine that could be offered to its free software users.<sup>5</sup> Acquisition of storage engine technology and associated patents from Sony-Ericsson offered opportunities on the proprietary side of the business that could have been further used to differentiate free and non-free versions of MySQL, but MySQL's Enterprise edition remained technically identical to the version distributed under GPL.
28. Any more diversified acquirer of MySQL, not entirely dependent on revenue *from* MySQL to support its development and distribution, would be aware of the benefit that would accrue from a wider adoption of the free version and a broader community for development of improvements than could be achieved within the narrow confines of the business model that had hobbled MySQL AB. Sun Microsystems, during its comparatively brief period in possession of the MySQL copyrights, showed awareness of the business logic counseling reduced concern with dual-licensing revenue.

---

<sup>4</sup>See <http://www.gnu.org/press/mysql-affidavit.html>

<sup>5</sup>The acquisition of Innobase by Oracle in 2005, which put Oracle in control of an essential component of MySQL as used by enterprise, did not result in either competition law inquiry or anti-competitive actions by Oracle. [http://www.oracle.com/corporate/press/2005\\_oct/inno.html](http://www.oracle.com/corporate/press/2005_oct/inno.html)

29. MySQL is not now distributed under the “pure” GPLv2. Both MySQL AB and Sun provided a “linking exception” to permit MySQL client libraries to be linked into applications programs not released under GPL. Sun broadened somewhat the FOSS license exception,<sup>6</sup> to allow any program released under a major FOSS license, whether or not compatible with GPL, to link with the client libraries distributed under GPL. Sun put substantial marketing stress on the encouragement of free downloads of the GPL’d MySQL product, because it had a diversified line of hardware, software, and systems integration services to offer customers, and realized that it didn’t need MySQL OEM licenses to pay for all the development and support of MySQL itself.
30. In this context, I am surprised to find nowhere in section 4.7 of the SO a mention of Microsoft SQL Server in relation to the MySQL ecology. Whether or not MySQL is a significant “competitive constraint” on the selling of Oracle 11, MySQL is an effective, indeed potentially lethal competitor for the much less powerful Microsoft product. For Sun, and *a fortiori* for Oracle, the rather scant dual-license revenue available from MySQL, or the long-term negative consequences from competition between MySQL and the top-drawer product line, must be significantly outweighed by the value of GPL’d MySQL, available at near-zero price, as a formidable competitor against Microsoft’s only database offering.

## Scenarios of Possible Oracle Anti-Competitive Conduct and the Resulting GPL Dynamics

31. From the ecological perspective most useful in analyzing the consequences of potential anti-competitive conduct in a GPL’d commons (a form of analysis familiar to me from my work as chief legal drafter of GPLv3—an activity which required countless exercises in scenario definition and analysis in order to optimize the terms of GPLv3—and from my daily work with my colleagues at the Software Freedom Law Center in considering potential legal threats to and responses by our clients, many if not most of whom employ GPLv2 or GPLv3) there are three possible courses of conduct that could be engaged in by an acquirer of MySQL’s copyrights against which GPL might need to provide. I am content that in each of the three scenarios it is difficult or impossible for a party in possession of the copyrights and associated legal privileges effectively to extinguish the free competitor. The SO analytically conflates the three possible courses of action. Clear specification of each makes the analysis simpler.
32. In the first scenario, which is referred to in SO at 736 and 754, Oracle decides to “abandon” MySQL, ceasing to provide either maintenance or enhancements. Because, as the SO notes, all parties agree that there would be no restriction on other parties copying, modifying and redistributing MySQL code, Oracle could not thereby require the forced migration of users from MySQL to its own products (a scenario common to the proprietary database market, as when Oracle purchased PeopleSoft or IBM purchased Informix in order to acquire their users). Given the free software nature of MySQL, the result of such a step by Oracle would effectively be to make a very expensive present to HP, IBM, and any other companies who wanted to make marginal investments in ensuring the longevity of this “competitive constraint” on Oracle.
33. In this scenario, the SO’s long estimate of the time necessary for a “fork” to gain acceptance and compete advantageously would be reduced to zero. A large existing user base requiring support and willing to pay for it would constitute a guaranteed

---

<sup>6</sup><http://www.mysql.com/about/legal/licensing/foss-exception/>

market, and the subsidy provided by Oracle through the effective abandonment of a fully-functional and popular codebase could be expected to attract both firm and non-firm developers. Aside from making poor use of the investments of Oracle shareholders, abandoning a GPL'd project is an outstandingly poor way to prevent it from competing against the owner who abandons it.

34. In the second scenario, Oracle does not abandon the existing MySQL user base, nor does it invest in enhancing MySQL, but it does everything possible outside the realm of the GPL to discourage community development efforts that might make MySQL more competitive with Oracle's own products. (See SO at 766) It promulgates a trademark policy that prevents any modified version of MySQL from using the name. It threatens enforcement of its patent claims against any modified versions of MySQL. It inhibits development of third-party software to interoperate between MySQL and Oracle databases, etc.
35. In this possible course of events, Oracle tests all non-copyright mechanisms, which lie outside the scope of the GPL, to burden the commons. Experience shows that the trademark attack is likely to fail. Popular free software programs with large personal as well as enterprise install bases have been denied use of their "brands" in the past. GAIM was denied use of its name by AOL and became Pidgin; Phoenix became Firebird became Firefox, which in turn denied use of its name to the Debian browser version now known as Iceweasel. Lawyers for free software projects have learned that name changes are of minor concern to projects substantially based around commons production, and that trademark forms a poor way to restrict commons development.
36. Patent aggression against modifiers of one's own GPL'd programs is also difficult or impossible. The breadth of the implicit patent license under US patent law extends, according to the cases, to "repair" as a limited form of "making." So maintenance to a product distributed under GPL by third-party maintenance providers is unlikely to result in patent liability for either the provider or its customers, who can always download another GPL'd copy from the patent-holding upstream copyright owner and apply third-party patches themselves. Oracle, I am sure, is well aware that suing one's customers is a poor business model.
37. In this connection, I would be more cautious about the legal conclusion reached in SO 765, involving the example of feature Z, placed in a copy of a GPLv2 MySQL distributed by Oracle, and then reimplemented a second time, elsewhere in the MySQL codebase by a subsequent modifier. The conclusion drawn may be correct under one or more patent systems, but in my own, I am doubtful. If, for example, we were talking about a literal copy of the patent-covered code duplicated from one portion of the codebase to another, I very much doubt that the second invocation of the same code could be patent infringing, while the first is covered by implicit license. In any event, the implicit license is to use the patent claims, and it is hard to see how the license to use could fail to cover both copies. Whether the provision of a second literal copy of the code constituted "making," would I suppose hinge at most on whether the code was actually duplicated, or was turned into a subroutine called from more than one location. In any event, I would not expect the threat of such a patent infringement action to be taken seriously by any US party engaged in modifying the code.
38. I should also note an error in the legal analysis of the consequences of Oracle's putative decision "ensuring that the code cannot be acquired under [GPL] anymore from the moment the transaction is completed." SO 766. Under GPLv2 any party distributing the code commits to automatic issuance of a new license, from itself to any party to whom a copy is distributed. GPLv2 section 6. In this



scenario then, contrary to the conclusion of the SO, all recipients of the code from anyone, regardless of the continued availability of that code from Oracle, receive automatically a copyright license to the code, from the copyright holder, which conveys the same implicit patent license as though the code were acquired directly from it.

39. For these and other reasons, a decision by Oracle to keep MySQL nominally alive as an Oracle product but to actively discourage commons co-development is likely to fail. It would require investments, produce little revenue, abandon an opportunity to cause disproportionate competitive harm to Microsoft, and it would irreparably damage Oracle's standing with the free and open source software communities on which it will be inclined to cooperate over other assets, including OpenOffice, OpenSolaris, and Java, which it has acquired from Sun Microsystems and in which it is also already significantly invested. Such a course is therefore neither rational nor likely to be successful, and I think it is reasonable to predict that it is not going to be tried.
40. The third significant possible course of conduct is that Oracle might progressively proprietize MySQL, by adding features, including features affording heightened interoperability with Oracle databases, only to customers buying commercial licenses, while meagerly enhancing the free version and refusing to integrate third-party contributions. (See SO at 733, 736 and 772)
41. This strategy is not unknown for firms in more or less complete control of the copyright for commercially significant free software products. But accumulated experience indicates that it usually works poorly. Many redistributors and users believed, for example, that Sun Microsystems itself essentially practiced such a strategy with respect to OpenOffice. Sun had its own proprietary version of the Star Office suite available under commercial license, it had sold to IBM the right to produce another proprietary version under the Lotus Symphony name, and downstream free software redistributors found it extremely difficult to get patches into the official OpenOffice.org distribution under Sun control.
42. The result was a *de facto* fork under the management of Novell. Go-oo.org, and the Novell Edition of OpenOffice (which like OpenOffice are LGPL licensed works, though like all LGPL licensed programs they can also be distributed under GPL), show the comparative ease with which other firms with a commercial interest in free software programs can counteract a perceived "go slow" development approach by a copyright holder with conflicting commercial interests.
43. The OpenOffice situation is one reason among many to doubt the conclusion reached by the SO (760) that "in fact, a fork of a major piece of software that is available under the GPL with a view to create a commercially successful product against the wishes of the project (and copyright) owner has never happened." Two of the Free Software Foundation's most significant and commercializable projects, GCC and GNU Emacs, have historically forked against the Foundation's wishes. The EGCS fork of GCC eventually rejoined GCC, while Lucid Emacs, now known as XEmacs, remains a separate development path with a codebase that cannot be merged with GNU Emacs as a result of their long genetic separation. Samba, a GPL'd project with whose commercial value the Commission is well-acquainted,<sup>7</sup> was forked by Luke Leighton to create Samba-TNG. The Joomla! content management system, a more recent example, forked from the parent Mambo CMS in 2005, and now has almost entirely eclipsed the parent project.

---

<sup>7</sup>This is one reason to distrust the expertise of Microsoft in opining on GPL-related issues, rather than relying upon it, as the SO does (758) in concluding that "Linux is the only commercially viable software product that is distributed exclusively under the GPL."

44. The goal of GPL is to create development ecosystems, in which diverse parties with diverse commercial and non-commercial interests interact on a basis of guaranteed trust to achieve results unattainable by individual firms or even business models standing alone. GPL's success in achieving its design objectives lies in the elegance with which its basic rules align the incentives of common participants and reduce the benefits of defection. The GPL ruleset has enabled competitors and cooperators to produce collectively all sorts of specialized software while resisting the various strategies for appropriating the commons. The MySQL ecosystem never attained maximum strength, precisely because the dual-licensing business model of MySQL AB forced it to turn away community resources available for the program's improvement. Far from being a challenge justifying a further weakening of the ecosystem, by replacement of the GPL by a permissive license for example, the Oracle acquisition of the MySQL copyrights could be the occasion for a significant strengthening and broadening of the community.
45. The crucial issue is not the license under which MySQL is distributed, although GPLv3 might be preferable to GPLv2 if one were writing on a clean slate. Rather, the central issue is an increase in the copyright diversity of the project, in which multiple parties have significant code in the main line. This would be sufficient to prevent anyone having an exclusive right to make proprietary enhancements or to undertake distribution under non-free licenses. While it is unlikely that Oracle will in the short term decide to incorporate third-party copyrighted modifications into the Oracle base product, thus terminating its dual-license business opportunities, in the long run the cost of maintaining *two* MySQL source trees (one for GPL and one for commercial versions) is more than the dual-license business is worth, particularly since the GPL branch in its wholly-owned source trees must be prepared to compete with any third-party GPL'd modified version that might be supported by Novell, Red Hat, Ubuntu, and other distribution vendors, as well as IBM, HP, and other systems companies.

## Conclusion

46. The analysis presented by the SO contains a number of legal and factual judgments about which there seems to me sufficient basis for doubt. The primary judgment to be made, as the SO says, is whether there would be "sufficient" development and distribution alternatives, assuring the continued vitality of MySQL. (731) In my view, the SO unduly emphasizes some of the aspects of the situation and undervalues others. The general design of GPL and the robust diversity of business models and non-firm development ecologies it creates seem to me the central sources of doubt. Secondary concerns arise from the places where the analysis seems to me to rest on legal conclusions that are unestablished or factual assertions made by others to the Commission which are demonstrably incorrect.
47. In my view, the SO reaches an incorrect conclusion in supposing that GPL itself provides insufficient protection against anti-competitive conduct by a copyright holder in Oracle's putative position. All scenarios likely to result from Oracle's acquisition of the copyrights, whatever Oracle's business intentions may be, are tolerable from the point of view of securing the freedom of the codebase, which is the primary concern of the communities I represent.
48. Further measures for the security of the software's freedom are possible. The improvements to GPL's effect introduced in GPLv3 would be of some additional value. Incorporation of third-party contributions in the version of MySQL distributed by Oracle—which in my view it is likely Oracle will find it desirable to

accept in the medium term, as its own experiences of the benefits and drawbacks of dual licensing for a wealthy and diversified firm lead it to choose its own business direction—would provide further security for the health of the MySQL ecosystem, taking it beyond any point reachable by the original MySQL AB given the scale of its resources. Without expressing any opinion on any other aspect of the Commission’s ongoing merger investigation, I believe that the issues raised concerning the GPLv2 status of the MySQL codebase do not warrant a conclusion that this transaction threatens significant anti-competitive consequences.

Respectfully submitted.