Collaborative Innovation and the Patent System – Replacing Friction with Facilitation

David J. Kappos
Ray Strimaitis

> "Why do we pay the open source and free software movements [so much]… attention?
>
> We [find tremendous novelty in]… a community that engages in decentralized intellectually creative activity, marked by overt borrowing, without asserting intellectual property rights – relying instead on a service industry model, prestige economies and substantial amounts of both donated labor and fee-for-service labor.
>
> We [find tremendous novelty]… in a community that is also marked by an insistence that it is a fundamental right of both insiders and outsiders to get meaningful access to the code [produced by the community]… to read, tinker or simply copy that code wholesale."[1]

Many wonder at this novel community, thinking it unsustainable, comparing it with failed social models.

Consider for a time just how novel, just how improbable, that community really is. We'll return to that community after discussing the challenges IBM sees for the patent system in addressing persistent issues with software-based inventions, in an era now undergoing fundamental change to how innovation occurs.

The Changing Nature of Innovation

Intellectual property protection in the United States is based on the mandate for Congress 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". Those of us practicing in the field of intellectual property in the United States for more than a few years have seen this mandate employed, both legislatively and judicially, to address the changing needs of the information technology industry. Indeed, it was the need to change that led to the declaration of software's patentability.

We will present IBM's views of how a changing information technology industry has driven the need to adapt IBM's intellectual property strategy, how software

---

[1] James Boyle, Presentation to the American Association of Law Schools Annual Meeting 2005 (on file with author).

patents can (and must) coexist in both open and proprietary technology development models, why in response to the changes taking place in the IT industry IBM pledged 500 patents to the open source community, and how changes to the patent system would lead to better quality and certainty in software patents.

To fully appreciate the criticality of the problems implicated by software patenting, it is important to understand the context in which we find ourselves. The National Innovation Initiative addressed this subject in its December 2004 report entitled Innovate America, in which it identified an 'innovation imperative'. This imperative challenges us to rethink our education system, our system for financing business, and our intellectual property regime. The report states:

> **Innovation will be the single most important factor in determining America's success through the 21st century... America's challenge is to unleash its innovation capacity to drive productivity, standard of living and leadership in global markets. At a time when macro-economic forces and financial constraints make innovation-driven growth a more urgent imperative than ever before, American businesses, government, workers and universities face an unprecedented acceleration of global change, relentless pressure for short-term results, and fierce competition from countries that seek an innovation-driven future for themselves. For the past 25 years, we have optimized our organizations for efficiency and quality. Over the next quarter century, we must optimize our society for innovation.**

The issues identified are as pertinent to other countries as they are to the U.S. How do we optimize a "society for innovation"? For starters, we must acknowledge that certain phenomena drive innovation more than others. In today's information technology environment, **collaboration** is one such phenomenon. The open source movement and open standards facilitate innovation by taking advantage of collaborative problem solving and efficient interoperability.

Open Source Software as Hallmark of Collaborative Innovation

Consider open source. Open source generally refers to software for which source code is available to the public. Any user of open source can modify that source code, but must make such modifications available to other users.   Open source is typically created as a collaborative effort among a number of programmers with the goal of improving the code and sharing the changes within the community. This is in sharp contrast to the proprietary software development model – in which one company or organization controls the development of the software, and makes it available in object code form only.

We know from history that both models work, and both have a role to play. The open source model enjoys the advantage of very high efficiency attributable to collaborative problem solving and broad sharing of development expense, and flourishes in areas involving infrastructure software which attract broad communities of interest. The Linux operating system is an example. Contrast this with highly specialized software attracting a much smaller base of interest. This is where the proprietary development model will continue to flourish.

Open Standards as Enablers of Collaborative Innovation

Consider open standards as another example. Open standards refer to hardware or software specifications developed by neutral consensus-based organizations in an open participatory environment, and made publicly available for all to implement on equal terms. Vendors compete based on features and functions in their product offerings, consistent with the open standards upon which those products are built. In so doing, one vendor's offering can be interchanged with another's – so long as each complies with the given open standard. The acceptance of open standards ensures not only efficient interoperability – it fosters healthy competition and collaboration. Most importantly, it puts governments, citizens, and customers in control of their information, freeing them from dependence on any single technology vendor for access to their own information.

The Role of Patents in Collaborative Innovation

Where do patents, or any form of intellectual property, fit into the paradigm of these innovation accelerators? As mentioned at the outset, United States intellectual property laws are predicated on the premise of promoting "progress of science and useful arts". Wouldn't granting any company or inventor exclusive rights to ideas affecting innovation accelerators (e.g. open source and/or open standards) stifle progress of science and useful arts? Is it better to stockpile patents for defensive purposes or is it better to share them with a community who can develop and bring those ideas to market faster?

It is these considerations that led IBM to its patent pledges. Key to those pledges, IBM patents may be used to create open standards, open source efforts, and innovation.

IBM believes the innovation system is changing. A new, more open model of innovation is taking shape. This new model comes about because of the complex nature of many problems faced by the technology field. No individual or company has the resources to address all aspects of today's toughest problems. Companies, academia, and individuals must work together to find the collective skills across the diverse technical fields involved. For example, the emerging

field of nanotechnology involves the combined disciplines of physics, chemistry and electrical engineering; challenges confronting health care require the viewpoints of researchers, doctors, patients, insurers, governments and others. No one person, no one company, no single government body has all the answers. The same basic collaborative concepts underlying open source supply the means already being adapted to address these problems across enterprises, universities, and individuals.

While open source may seem to be the antithesis of proprietary software, it is IBM's view that open and proprietary software must coexist and complement one another. It was no coincidence that IBM pledged not to assert over 500 patents against open source the same day IBM reported that it had been issued over 3200 United States patents in 2004, leading the world for the 12th consecutive year. These two announcements underscore the interplay of open and proprietary, and that intellectual property is central in striking the balance between the two.

As noted at the outset, changes in United States patent practice have come from both the legislature and the judiciary – it was the need to change that lead to the development of software's patentability. In a recent Wall Street Journal editorial, former Commissioner of Patents Bruce Lehman observed:

> ***In the U.S. it is our courts -- not Congress -- that have interpreted our patent laws to encompass new subject matter. Twenty-four years ago the Supreme Court held that software could be patented in the landmark case of Diamond v. Diehr. Justice William Rehnquist's opinion observed that patents can "include anything under the sun that is made by man." Since then tens of thousands of software patents have been issued in the U.S.***

As noted by former Commissioner Lehman, the U.S. Courts (not Congress) judicially extended patent laws to encompass software. Unlike laws created by Congress, there is no Congressional record, background or insight into the intent behind this expanded interpretation. In light of the "anything under the sun" mandate, should disclosure and examination requirements be tailored to accommodate the expanded scope of patenting?   Has a judicially expanded scope of patenting run afoul of the objective of promoting "the progress of science and useful arts"? What is more, if harmonization of United States patent laws with those of other countries is also an objective, how does one reconcile broad patent coverage in the United States with narrower coverage in other countries?

Gaps in the Current Patent Laws That Inhibit Quality in Software Patents

IBM is a strong supporter of patent reform needed to improve software patent quality and aimed at preventing overbroad patents, enabling challenges to

patents that may have been improperly granted, and maintaining public trust in the standard of patentability. The current law has gaps in this regard.

Gap between the Disclosure/Enablement Requirement and Scope of Claim in Software Patents

One major gap lies between the disclosure and enablement requirement, and scope of claims obtainable based on that disclosure/enablement. Under current law and practice a patent applicant is able to disclose and enable only a single embodiment, but obtain claims covering subject matter far beyond what is disclosed and enabled. Worse yet, many applicants obtain claims covering ALL solutions to the problem addressed by their invention. It is difficult to conceive of a disclosure so omnipotent that it adequately discloses and enables all possible solutions to a given problem. But under present law an applicant can make and describe a narrow invention, and obtain claims covering much more subject matter than was disclosed and enabled, to the detriment of the public.

In exchange for the exclusive rights provided to patentees, it is important that patentees provide to the public a disclosure that adequately teaches the patented invention to those skilled in the relevant art. Anything less deprives the public of meaningful access to the inventive concept, while unjustly rewarding the patentee with a monopoly on an ill-described concept. Indeed, this is often a criticism of software patents in the United States. Not only are competitors and innovators unable to utilize the patented teachings to advance the state of the art, but perhaps worse, they cannot determine the scope of the patent and consequently do not know where they are free to act. Furthermore, inadequately described patents deprive patent examining authorities of an adequate base of prior art for purposes of examining subsequent new patent applications. No good comes from a gap between disclosure/enablement and claim scope.

The field of software presents major challenges to determining the appropriate standard of disclosure and applying it consistently. The inability to meet these challenges well gives rise to many overbroad software patents that have received much notoriety, often in the context of exposing the poor quality patents produced by the US Patent Office. In order to ensure an adequate level of disclosure in patents, clear guidance is needed on what constitutes a proper description. Such guidance must of course be tailored to the subject matter of the invention but at the same time must be applied consistently, reflecting the same standards of patentability regardless of subject matter. Although it is understood that a patent document is written to an audience of those skilled in the relevant art, the level of skill is, and should be acknowledged as, a function of the field of technology. Therein lies the means to close the gap.

The problem with the level of disclosure versus claim scope is rooted in the inordinately high level of skill accorded to the person of ordinary skill in the

software arts, leading to the issuance of software patents exhibiting a low and often inadequate level of teaching to the public. In their paper **Is Patent Law Technology-Specific?,** Dan Burk and Mark Lemley address what, in their view, is a diverging approach between how the Patent Office treats software inventions/patents and those inventions/patents in other fields. They observe:

> **Of late, however, we have noticed an increasing divergence between the rules actually applied to different industries. The best examples are biotechnology and computer software. In biotechnology cases, the Federal Circuit has bent over backwards to find biotechnological inventions nonobvious, even if the prior art demonstrates a clear plan for producing the invention. On the other hand, the court has imposed stringent enablement and written description requirements on biotechnology patents that do not show up in other disciplines. In computer software cases, the situation is reversed. The Federal Circuit has essentially excused software inventions from compliance with the enablement and best mode requirements, but in a way that raises serious questions about how stringently it will read the nonobviousness requirements. As a practical matter, it appears, while patent law is technology-neutral in theory, it is technology-specific in application.**

Indeed, in many cases software patents merely state a desired objective while providing little or no guidance on how to achieve it. In other cases there is description and enablement provided for a single embodiment of the invention, while the claims are so broad as to preclude all possible solutions to the underlying problem. In effect it is presumed that software developers are extraordinarily skilled, such that the mere disclosure of a problem is sufficient to teach all possible solutions. Of course this cannot be the case.

Genus/Species Concepts from Chemical Arts Should be Adopted in Software Arts to Limit Allowable Claim Scope Based on Disclosed and Enabled Subject Matter.

The solution to closing this gap lies in two fundamental changes needed in patent examination. First, the scope of protection afforded an invention should be consistent with the scope of the description provided in the patent. Public trust in the patent system requires that the extent of patent protection be commensurate with the disclosed inventive contribution. Otherwise, competitors will find themselves unduly precluded from competing and innovators will find their inventions dominated by overbroad claims. It is thus important to match the scope of protection to the scope of description provided.

Where the specification describes only a species (in any field, e.g. electrical,

chemical, or mechanical), a claim should not be allowed or valid for a broad genus unless the public is instructed as to how the broader technology is implemented. A claim that recites a result (as opposed to a specific structure or series of steps that produce the result) should be automatically reviewed for adequate description. In that such claims may cover any and all ways of achieving the result, and thus preclude others from entering an entire field, such claims should be closely scrutinized to ensure that all covered embodiments are described and enabled by the specification. Section 112 and its implementing rules should be revised to prevent such overbroad claiming. Specifically, statutory revision is needed to restrict valid claim scope to only those species fully capable of implementation based on the teachings of the specification, without undue experimentation. Legislation is also needed to provide "results-directed" claims be given a special review for overbreadth. Moreover, in that "equivalents" are generally an uncertain penumbra of protection around a literal claim, Section 284 should be revised to preclude enhanced damages when infringement is based on an equivalent under the doctrine of equivalents or under 35 USC 112 paragraph six.

Prior Art References Directed to the Field of the Invention should be Presumed Combinable

The second change needed to close the gap between disclosure and claim scope involves presuming the same capability of skilled artisans to apply the prior art, as is presumed of their ability to divine meaning from the patent disclosure. The scope of the problem solved by a claimed invention should be interpreted as broadly during patent application examination as will invariably be argued by the patentee in later asserting the patent. Thus, the examiner should be able to look at the patent as a whole to broadly determine the field of the problem solved and should be empowered to presume that an artisan faced with that problem would be motivated to combine any references directed to the field of the problem.

Where all claim elements are found in a combination of prior art references, and those references are directed to the same field of problem as the invention, motivation to combine the references to support the rejection of the claim as obvious should be presumed. Under current law, references may not be combined to render a claim obvious unless there is a motivation for such combination. But references generally do not include explicit statements describing combinations with other references. Thus, even when all claimed elements are found among two or more references, examiners are frequently unable to reject the applicable claim because none of the references includes the required teaching, suggestion, or motivation to combine it with the other references. Motivation to combine references may be implied in certain circumstances, but the law governing such implication is unclear and not easily or evenly applied. In addressing obviousness, an examiner should have the

power to combine prior art references that are directed to the same field of problem as the invention.

Motivation to combine can, under present law, be found in the "nature of the problem" solved by the invention. There is little guidance, however, for determining the nature of the problem, and "the nature of the problem" is often read narrowly by focusing on one or another aspect of the patent specification. The test should be clarified so that the problem solved by the invention must be defined as broadly as possible by examining the entire patent application, including the claims and the references cited by the applicant.

Interoperability Must Play a Role in Patent Enforcement

As previously stated, IBM is a supporter of patent reform – both in the United States and in other countries. The Computer Implemented Inventions (CII) provision, recently the subject of active debate in Europe, is an example of an issue closely linked to the interoperability and collaborative innovation, whose time has come. Combine the current scenario of patents blocking interoperability between software products with the previously cited shortcomings in our patent system not applying sufficiently stringent tests in evaluating the sufficiency of disclosure and patentability, and one obtains a sense for the overall scope of the challenge facing the patent system in dealing with software patents. Respecting software patents, we recognize the polarizing nature of the issue as some seek to preserve broad patent protection for software and others seek to significantly reduce such protection. An exception to patent enforcement for computer implemented inventions may serve the larger goal of promoting innovation through increased interoperability.

The issue at the core of interoperability is whether patents should be used to prevent one software product from working with another software product where the software product is marketed by the patent owner with the intent of interoperating and it enjoys the benefits of interoperating with other products. For our industry -- and our customers -- to realize the maximum possible value from IT investment, there needs to be a high degree of interoperability among products. An interoperability exception would provide the necessary clarity to allow competing and complementary products to work together. It would apply only to the level of interface functionality necessary for two products to communicate and use the data communicated.

A Vision for a Patent Office of the Future

The concerns we've raised relate to patent quality and efficiency in the prosecution of patent applications. How could a patent office be structured to take advantage of new technologies – with the ultimate goal of improving patent quality and enhancing efficiency? Just as traditional "bricks and mortar"

industries have evolved their operations to take advantage of the Internet and network synergies, it is conceivable that patent offices could adopt similar creative approaches.

For example, why couldn't a patent office automatically issue customized alerts, based on a given user's key words, search strategy or technological field of interest, notifying that user of published patent applications meeting the search criteria or field of interest?   Would it not improve patent office efficiency to offer the public an on-line tool for submitting relevant prior art that would go directly to the patent examiner? Isn't it likely that the submitted prior art from interested members of the public would be as relevant (or even more relevant) than the results of a prior art search conducted by the patent examiner? Wouldn't this be a good way to unlock the non-patent software prior art that is woefully lacking in most patent office libraries?

Upon beginning review of a patent application, certain details may not necessarily be clear to the patent examiner. To the extent that reasonable questions arise where the knowledge is in the possession of the applicant, why not ask the applicant to provide and/or confirm the examiner's understanding of certain issues pertinent to the examiner's review, including: the location within the specification of corresponding support for particular claim elements where support for cannot be found elsewhere in the application; the location within the specification of where certain figures or drawings are explained when unclear to the examiner; and when using "means or step plus function" claim language (e.g. 35 USC 112, paragraph 6), where corresponding structure, materials or acts are described in the specification and/or equivalents thereof.

Wouldn't information obtained through this initial exchange result in far fewer unjustified or inaccurate rejections being imposed in the first action on the merits by the examiner? Aren't examiners more likely to identify the key issues governing patentability faster than is typical under existing practice? Shouldn't a better record regarding the application under examination result?

The motivation behind these questions is not change for the sake of change. The motivation is to improve the quality of patents and the efficiency of patent office operations. Isn't this similar to the collaborative approach of the open source community?

The Open Source Model May Not Be As Novel As it Appears

> "Why do we pay the open source and free software movements
> [so much]… attention? This is the question with which we started.
> We [find quite novel]… a community that engages in decentralized
> intellectually creative activity, marked by overt borrowing,
> without asserting intellectual property rights, – relying
> instead on a service industry model, prestige economies

and substantial amounts of both donated labor and fee-for-service labor.

We [find quite novel]… a community that is also marked by an insistence that it is a fundamental right of both insiders and outsiders to get meaningful access to the code that is the community's output….

Why then [do]… we not [find so novel]… the legal profession [itself]?

The first lawyer to bring a palimony suit or to turn the law of intentional infliction of emotional distress into a sexual harassment claim gets no right to exclude.

[We]… can all follow in her footsteps, copy her coding, the very words and form of her filings – indeed it is a requirement of much of the "executable" part what we do [as lawyers] that our coding be made publicly available, open to the scrutiny and the subsequent tinkering of all.

Not only do we seem to think that the open source and free software movements are somehow special, we make claims about those communities which we would never make about other more familiar examples such as the legal profession itself.

[Critics declare that]… open source is unsustainable… because innovators must be able to capture the values of their innovation… Without such strong rights of exclusion, the community is doomed. No one seems to have told the [legal profession]… of this impending doom….

[T]he whole legal [tradition we so take for granted]… of borrowing and innovation and recoding seems to have survived [for hundreds of years despite that it would appear by comparison completely unsustainable]…."[2]

Perhaps the same will some day be said of the open source community.

---

[2] Boyle, Id.