

From: stephen o'grady [e-mail redacted]
Sent: Sunday, September 26, 2010 10:42 PM
To: Bilski_Guidance
Cc: [e-mail redacted]
Subject: Re: Bilski Guidance

145 Newbury St
Suite 2
Portland, ME 04101

September 26, 2010

Mail Stop Comments—Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

To Whom it May Concern,

I am writing in response to the request for comments re: *Bilski v Kappos* (<http://edocket.access.gpo.gov/2010/pdf/2010-18424.pdf>).

By profession, I am a software industry analyst. This means, in part, that my function is to study the software industry and make determinations on its future based on an analysis of the past and present. Regarding patents as they pertain to software, the only supportable conclusion I believe that can be reached is that they are acting against their original purpose. The purpose of a patent, broadly, is to incent and thereby encourage innovation. Software, by its very nature, is has become sufficiently complex relative to other industries so as to be effectively unmanageable by any current or projected grant system.

I include below the text of an article authored earlier this year on this subject, which also happens to be the most widely read item my firm has published in the last five years.

I hope and trust that you will exclude software from patent eligibility, because this is the logical conclusion if the goal is to encourage and foster innovation in this industry.

Sincerely,

Stephen O'Grady

Enclosure

=====

<http://redmonk.com/sogrady/2010/03/19/software-patents/>

Why I Am Against Software Patents

The surprise to most people isn't that I do not believe that software should be patentable. Given my long term interest in and coverage of free and open source software, I'm supposed to be at least mildly anti-establishment. It is also statistically unlikely that I would be in favor of patents, because industry sentiment is overwhelmingly against them at the present time (as the author acknowledges here <<http://entrepreneur.venturebeat.com/2010/03/04/in-favor-of-software-patents/>>).

Most expect me to argue, as has Brad Feld's anonymous lawyer <<http://www.feld.com/wp/archives/2010/02/sawyer-weighs-in-on-intellectual-ventures.html>>, James Surowiecki <http://www.newyorker.com/talk/financial/2008/08/11/080811ta_talk_surowiecki> or Red Hat <<http://bx.businessweek.com/open-source-software/view?url=http://arstechnica.com/open-source/news/2009/10/red-hat-bilski-brief-says-software-patents-stifle-innovation.ars>>, that patents are actually counterproductive with respect to innovation. That the entire purpose of a patent – to stimulate invention by granting the inventor wide-reaching protections – is subverted as broad, over-reaching patents are accumulated like mercury by competing organizations that are unwilling, unable or both to work together to advance markets. But while I agree with the sentiment, that's not why I am against software patents.

Others expect me to assert, as did <<http://www.unionsquareventures.com/2010/02/software-patents-are-the-problem-not-the-answer.php>> Union Square Ventures' Brad Burnham, that software is, by its nature, different from physical inventions and innovations. That it does not require the same protections to stimulate invention that physical goods do. But while I believe this to be true, this is not why I'm against software patents.

Others expect me to argue that, as Stephan Kinsella has <<http://mises.org/daily/4018>>, that patents are part of a system that is a net drain on the global economy (\$31B, by his estimate). This is not why I'm against software patents.

Still others expect me to argue that the greater good – a dangerous phrase if ever there was one – demands that software be unpatentable. That Nathan Myhrvold's Intellectual Ventures <<http://www.nytimes.com/2010/02/18/technology/18patent.html?scp=1&sq=Nathan%20Myhrvold&st=cse>> is the epitome of evil in the world, with a revenue model based strictly on extracting value from an antiquated patent system that has been mistakenly applied to an industry that requires no such protections. But while I personally believe that Myhrvold's company is based entirely on extracting profit from a broken system rather stimulating invention as he claims – that Intellectual Ventures is just a version of those infomercials seeking ignorant "inventors" to exploit writ large – this isn't why I'm against software patents.

One last group expects me to contend that those in favor of more limited patent grants, such as FairSoftware's Alain Ranaud <<http://entrepreneur.venturebeat.com/2010/03/04/in-favor-of-software-patents/>>, are failing to acknowledge – deliberately or otherwise – the speed of the software industry for which anything measured in years is an eternity. But though it is true that even a two year limit, let alone Ranaud's proposed seven, is a lifetime in this business, that isn't why I'm against software patents.

The reason I am against software patents is, by contrast, very simple. It's not rooted in philosophy, it doesn't involve theories of good or evil; it's not even about debating what is likely to spur more or less innovation.

I am against software patents because it is not reasonable to expect that the current patent system, nor even one designed to improve or replace it, will ever be able to accurately determine what might be considered legitimately patentable from the overwhelming volume of innovations in software. Even the most trivial of software applications involves hundreds, potentially thousands of design decisions which might be considered by those aggressively seeking patents as potentially protectable inventions. If even the most basic [elements](http://www.cnn.com/2010/TECH/02/26/facebook.patent/index.html) [of these are patentable](http://www.techflash.com/seattle/2010/03/amazons_1-click_patent_confirmed_following_re-exam.html), as they are currently, the patent system will be fundamentally unable to scale to meet that demand. As it is today.

In addition to questions of volume are issues of expertise; for some of the proposed inventions, there may only be a handful of people in the world qualified to actually make a judgment on whether a development is sufficiently innovative so as to justify a patent. None of those people, presumably, will be employed by the patent office. Nor are the incentives for fact witnesses [remotely sufficient](http://pmuellr.blogspot.com/2010/02/prior-art.html). Nor will two developers always come to the same conclusions as to the degree to which a given invention is unique.

I have no relevant expertise in other physical or science industries, and as such I have no educated opinion on whether innovations there should or should not be patentable. I can however state with confidence that the patent system as applied to software today does not work, nor is there is any reasonable expectation that it will or could in future.

If we acknowledge that this is the case, which I believe one must if the available evidence is considered, then it is no longer possible – whatever your philosophical viewpoint – to be in favor of software patents.

And so I am not. Just like Tim [Patent-Fail](http://www.tbray.org/ongoing/When/201x/2010/02/22/Patent-Fail).

Update: Stefan Gustavson was kind enough to translate this article into Swedish. Get that version here [here](http://www.openstandards.se/ogradymotpatent).