From: [e-mail redacted] **On Behalf Of** Ian Clarke
Sent: Monday, September 27, 2010 4:54 PM
To: Bilski_Guidance
Cc: [e-mail redacted]
Subject: Feedback on application of Bilski ruling on software patents

Dear Sir/Madam,

I am the CEO of the Austin, Texas-based Uprizer Labs LLC, and a US Citizen. My company develops predictive analytics technology used by some of the most popular websites on the Internet. You can learn more about our software at http://sensearray.com/. We work at the cutting-edge of applying artificial intelligence to solve commercial problems.

I have a personal history of innovation in the software industry, creating Freenet, the first anonymous decentralized peer-to-peer network, and co-founding Revver, the first online video sharing website to share revenue with video creators.

Given my background I hope it is clear that if software patents had the effect that their proponents claim, I would be their greatest advocate. They don't and therefore I am strongly opposed to them.

To summarize: **It is my strong belief that innovation in the United States would be best served if computer software were excluded from patentability, and it is my hope and recommendation that USPTO applies Bilsky in a way that makes it as difficult as possible to obtain and enforce patents on software.**

In general, I consider a "software patent" to be any patent that can be infringed by writing or executing a piece of software on a general-purpose computer. People sometimes use the term "computer-implemented invention" to describe what is covered by a software patent.

As of this time, I know for a fact that original software written by my company probably violates hundreds or thousands of software patents. This is not because I misappropriated anyone else's work, nor because I copied them, but simply because the nature of software is such that it is very common for many different software engineers to come up with a similar solution, given the same problem. The process of creating software involves solving tens or hundreds of these small problems per day.

Unfortunately, patent law, as enforced by the USPTO over the past two decades, have encouraged individuals and companies to seek patents on many of these small solutions. The availability of these patents did not encourage any innovation, rather they are acquired so that companies can amass a "war chest" of patents, which they can use to enter cross-licensing deals with other companies, effectively non-aggression pacts. The mere existence of these pacts is strongly indicative that there is a problem with software patents. Regrettably new innovative companies do not have the time nor resources to acquire these war chests.

Worse, some unscrupulous companies, commonly known as "patent trolls", amass patents for the sole purpose of threatening other companies, in the hope of extracting a lucrative pay-off. Ironically, these patent trolls avoid writing software themselves, this being the only way that they can be immune to a retaliatory patent suit. In this case, the beneficiary of the patent system scrupulously avoids innovation, while its victim is typically the real innovator.

Constitutionally, the role of patents is to promote the sciences and useful arts. In the field of software, patents have the opposite effect. They increase uncertainty, increase legal costs, entrench monopolies, and enrich patent parasites at the expense of actual innovators.

I can honestly say, and would sign an affidavit to this effect, that having watched software patents closely over my entire 15 year career in the software industry, and I have never once seen a useful innovation in software that would not have occurred were it not for the possibility of obtaining a patent. Even where a patent covers a genuinely useful innovation, typically the patent is an afterthought, it is not the motivation for the innovation. This observation alone demonstrates that software patents fail to meet the constitutional test of promoting the sciences.

I would further point out that almost all of the innovations in software that most affect our lives today, like the personal computer and the Internet, were created prior to 1990, in a world largely without software patents. It makes me wonder how much further along we might be were it not for the drain on innovation our industry has suffered since as a result of these patents.

Lastly, I know that you will be receiving a lot of feedback on this issue, and I expect that most of it will be from lawyers who make their living from filing for patents, and the litigation that results from them. It should therefore not be a surprise that their position will typically be in favor of broader patentability, just as an arms dealer might hope for more conflict.

I on the other hand have no ulterior motive, my purpose in writing this is simply because I seek the freedom to innovate without having to be sued by a competitor or a patent troll over a software patent on an obvious solution to a problem.

I hope you don't mind if I close by quoting a far more accomplished innovator than myself, John Carmack, co-founder of Id Software, one of the most successful computer games companies in the United States. Here is his view of software patents:

Before issuing a condemnation, I try hard to think about it from [a Lawyer's] point of view -- the laws of the land set the rules of the game, and lawyers are deeply confused at why some of us aren't using all the tools that the game gives us.

Patents are usually discussed in the context of someone "stealing" an idea from the long suffering

lone inventor that devoted his life to creating this one brilliant idea [...]

But in the majority of cases in software, patents affect independent invention. Get a dozen sharp programmers together, give them all a hard problem to work on, and a bunch of them will come up with solutions that would probably be patentable, and be similar enough that the first programmer to file the patent could sue the others for patent infringement.

Why should society reward that? What benefit does it bring? It doesn't help bring more, better, or cheaper products to market. Those all come from competition, not arbitrary monopolies. The programmer that filed the patent didn't work any harder because a patent might be available, solving the problem was his job and he had to do it anyway. Getting a patent is uncorrelated to any positive attributes, and just serves to allow either money or wasted effort to be extorted from generally unsuspecting and innocent people or companies.

Yes, it is a legal tool that may help you against your competitors, but I'll have no part of it. Its basically mugging someone.

Your's Sincerely,

Ian Clarke.