From: [e-mail redacted] **On Behalf Of** Paul Haahr
Sent: Sunday, September 26, 2010 6:02 AM
To: Bilski_Guidance
Cc: [e-mail redacted]
Subject: please end software patents -- feedback on Bilski v. Kappos

To the United States Patent Office,

I am a US citizen and software engineer. I am a named inventor on at least three patents (numbers 7,346,839, 7,409,383, and 7,783,639) and am named as an inventor for numerous pending patent applications. I am employed by Google, Inc.; this letter represents my personal opinion and not necessarily that of my employer.

Software patents are a significant threat to innovation in the software industry and, by extension, all of America's technology-related businesses. While I understand the theoretical case that software patents can foster innovation – by encouraging investment and advancing the state of the art through disclosure – I have not seen this to be the case in any way in the software industry.

On the issue of investment, there is no case I can think of in the software industry where a patent has lead to investment in a company that succeeded due to the patent's franchise preventing competitors from developing an equivalent product. Instead, the pattern for patent litigation and threatened litigation – as you are well aware – is for successful companies to be approached by non-practicing entities for damages or fees, with little or no money ever reaching the original inventors or their investors. Copyright and trade secrets do not have the same problems as patents and have proven very successful for protection in the software industry.

On the issue of disclosure, it should be noted that many, but not all, software "inventions" described in patents are inherently disclosed by making a product available to users. In many other cases, rather than disclosure, realizing that a problem has been solved encourages others to attempt solutions. Some of those solutions may be independent rediscoveries of the same underlying algorithm; others may be different algorithms, but in either case. The lifetime of patents make disclosures from software patents nearly useless – and fundamentally detrimental – as contributions to the state of the art; by the time the application period and twenty years have passed, many generations of software technology have passed.

Because software patents inherently give an exclusive franchise for algorithms that may be independently discovered, it is impossible for a software engineer to opt-out of the patent system. This coercive nature of software patents has forced many practitioners, myself included, to apply for patents against our wishes, because we want to ensure that we are allowed to exploit our inventions. And it is this potential for independent discovery of algorithms that exist independent of their applications that underlies a moral argument against patents.

In addition, the practical implementation of software patents has been terrible and damaging to the industry: the criteria for novelty are far too loosely applied, allowing many obvious applications to covered by patents; multi-year application processes lead to widespread adoption of techniques – including as part of industry standards – before a patent covering them is issued; and the language of patents and claims are so far removed from the working language of computer scientists that it is often difficult for an inventor to read his or her own patent and understand whether it covers the intended invention.

Speaking as someone who has worked in the software industry for more than two decades and as a former entrepreneur, a world where software companies competed on building the best products and independent discovery of algorithms is recognized as a legitimate, non-infringing activity would be vastly preferable to the current state of affairs, where work that appears obvious can lead to years of litigation about infringement. I urge you to reject all software patents, on both moral and practical grounds.

Thank you,
Paul Haahr
4019 Cesar Chavez Street
San Francisco, CA 94131