From: John Harlien III [e-mail redacted]
Sent: Sunday, September 26, 2010 3:37 PM
To: Bilski_Guidance
Cc: [e-mail redacted]
Subject: The USPTO's new guidance should include a strong stand against software patents

Software patents hurt individuals by taking away our ability to control the devices that now exert such strong influence on our personal freedoms, including how we interact with each other. Now that computers are near-ubiquitous, it's easier than ever for an individual to create or modify software to perform the specific tasks they want done -- and more important than ever that they be able to do so. But a single software patent can put up an insurmountable, and unjustifiable, legal hurdle for many would-be developers.

The Supreme Court of the United States has never ruled in favor of the patentability of software. Their decision in Bilski v. Kappos further demonstrates that they expect the boundaries of patent eligibility to be drawn more narrowly than they commonly were at the case's outset. The primary point of the decision is that the machine-or-transformation test should not be the sole test for drawing those boundaries. The USPTO can, and should, exclude software from patent eligibility on other legal grounds: because software consists only of mathematics, which is not patentable, and the combination of such software with a general-purpose computer is obvious. Most software patents describe the problem that the claimed "invention" solves but they don't detail *how* it is solved e.g. at least with design and implementation. As a result, all solutions to the problem are patented (and not just the one which is implemented by the patent applier).

Software Development unlike other forms of mechanical invention is already afforded protection within existing copyright laws.