Error 404: Software Patent Clarity Not (Yet) Found

Amundsen Davis Intellectual Property Alert
April 27, 2018

RELATED SERVICES

Intellectual Property

Last week, the Senate Judiciary Committee held an oversight hearing on the U.S. Patent and Trademark Office (USPTO). Newly-appointed Director Andrei Iancu provided **refreshing dialogue** for inventors—especially those in the software industry—indicating that much needed support for software patents is high on his agenda. But where do we stand today?

Software *can be* patented and is patented fairly frequently. But, software provides a unique challenge to patent applicants in most instances; below is a quick tutorial (or refresher) on what you can expect from the software patent examination process.

In addition to evaluating whether the invention is useful and novel, the USPTO considers whether the invention claims a law of nature, natural phenomenon, or an abstract idea. If the invention falls under any of these categories then it is *not eligible* for patenting. Essentially, the USPTO is ensuring that no one can patent, for example, the Pythagorean Theorem, naturally occurring genetic sequences, or fundamental economic practices. However, because the inventive aspects of software often lie inside a virtual black box, software inventions may be too quickly classified as abstract ideas and rejected by the USPTO. Making matters worse, the term "abstract idea" has no official definition.

One approach available to the patent applicant is to argue that the software invention is not an abstract idea, such as by showing that the software is directed to an identifiable improvement in computer functionality or in another technology. For example, computer animation methods automatically synchronizing lips and facial expressions using computer-implemented rules have been found to be a patentable improvement in computer technology, as has other software improving the method used by a computer to store and retrieve data from memory. On the other hand, concepts such as generating menus on a computer, filtering content, and collecting, classifying, encoding, and decoding data have each been upheld as unpatentable abstract ideas.

Another approach available to the patent applicant is to argue that, although the software appears to be directed to an abstract idea, the software includes additional features that amount to *significantly more* (again, no precise definition) than the abstract idea. The applicant can highlight the inventive steps which take the software beyond the abstract. For example, the patent applicant could argue that the software provides a solution to a known problem, practices a particular

method to achieve a desired outcome (as opposed to merely claiming the idea of the outcome), operates with a certain machine, or transforms an article to a different state or thing.

Patent eligibility for software can be highly subjective and may even depend on which examiner is assigned to the application.

Software architects and developers appreciate the complexity and novelty of their work, but the USPTO isn't always as understanding. Hire a professional to guide you through the messy landscape of software patent eligibility, and cross your fingers that Director Iancu helps pave a clearer path sooner than later.