NEAL ▪ GERBER ▪ EISENBERG
thirty years of trusted partnerships

Software Patents Aren't Inherently Abstract—Patent Appeals Court Clarifies and Enhances Software Patent Eligibility

May 16, 2016

In *Enfish, LLC v. Microsoft Corp.*,[1] the U.S. Court of Appeals for the Federal Circuit reversed a California district court's summary judgment that two software patents were directed to an "abstract idea" without "significantly more" and therefore patent-ineligible. Importantly, the Federal Circuit clarified that software patents are not inherently abstract or automatically subject to scrutiny under the second step of the *Alice* subject matter eligibility test.

This case clarifies how courts and the U.S. Patent and Trademark Office (USPTO) should apply the two-step patent eligibility test the Supreme Court set forth in *Alice Corp. Pty Ltd. v. CLS Bank Int'l*.[2] The first step of the *Alice* test asks "whether the claims at issue are directed to a patent-ineligible concept," such as an abstract idea.[3] If not, the claims are patent eligible. But if the claims are directed to a patent-ineligible concept, the second step of the *Alice* test asks whether claims include "significantly more" than the patent-ineligible concept to "'transform the … claim[s]' into a patent-eligible application" of the patent-ineligible concept. If so, the claims are patent eligible, and vice-versa.

Unfortunately, the Supreme Court never explained how to determine what constitutes an "abstract idea" sufficient to satisfy the first step of the *Alice* test. This has caused lower courts (like the district court in this case) and the USPTO to often trivialize the first step of the *Alice* test by simply: (1) describing the claimed invention as a very broad concept untethered from the actual claim language, (2) deeming that broad concept an "abstract idea," and (3) moving on to the second step of the *Alice* test.

The Federal Circuit admonished this approach. It first put to rest any argument that software patents are by definition abstract, stating that *Alice* does not hold "that claims directed to software, as opposed to hardware, are inherently abstract and therefore only properly analyzed at the second step of the *Alice* analysis."[5] Courts and the USPTO must carefully consider the first step of the *Alice* test for software patents, because "[s]oftware can make non-abstract improvements to computer technology just as hardware improvements can."[6]

The patents at issue in *Enfish* were directed to a self-referential computer database. In reviewing these patents, the Federal Circuit determined that the claims were directed to an improvement in computer functionality, not economic or other tasks for which a computer is used in its ordinary capacity. Specifically, the court found that the claimed self-referential computer database invention is an improvement over standard relational databases, and enables faster data searching, more efficient data storage, more flexible database configurations, and faster "on-the-fly" database software launching. The court held that this specific improvement to the way computers operate was not an abstract idea, and found the claims patent eligible.

---

[1] No. 2015-1244 (Fed. Cir. May 12, 2016).
[2] ___ U.S. ___, 134 S. Ct. 2347 (2014).
[3] *Id.* at 2355.
[4] *Id.* (citations omitted).
[5] *Enfish*, slip op. at 11.
[6] *Id.*

The Federal Circuit also recognized that tying the software to a general-purpose computer did not "doom[ ] the claims" to being abstract.[7] In fact, the Court clarified that a claim need not define an improvement in computer technology "by reference to 'physical' components" to be patent eligible. "To hold otherwise risks … creating a categorical ban on software patents."[8]

Here are some ways you can use this case to your benefit to procure software patents and rebut *Alice* arguments:

1. Rebut arguments that all software-type claims are inherently directed to abstract ideas or that software-type claims are categorically patent ineligible.[9]
2. Rebut arguments that the physical components of a software-type claim must improve a computer's functionality for the claim to be patent-eligible.[10]
3. When arguing that a claim isn't directed to an abstract idea, maintain focus on how the claimed invention improves computer-related technology, functionality, or capabilities.[11]
4. When arguing that an identification of the abstract idea is inappropriate, prevent attempts to trivialize the claimed invention or to downplay its benefits, and rebut a broad characterization of the claimed invention that isn't tethered to the claim language.[12]
5. When drafting a patent application, discuss the problems inherent in the prior art and explain how the invention improves computer-related technology, functionality, or capabilities to solve those problems. How the claimed invention improves on existing, conventional technology is an important consideration when determining whether the claim is directed to an abstract idea.[13]

---

[7] *Id.* at 16.
[8] *Id.* at 17.
[9] *Id.* at 11, 17.
[10] *Id.* at 17.
[11] *Id.* at 12.
[12] *Id.* at 14, 15.
[13] *Id.* at 18.

This alert was authored by James P. Muraff (312-269-8034, jmuraff@ngelaw.com) and Kevin J. Cukierski (312-269-8448, kcukierski@ngelaw.com).
If you have any questions related to this article or would like additional information, please contact your attorney at Neal Gerber Eisenberg, any attorney in the Intellectual Property & Technology Transactions practice group, or the authors. Click here for a full listing of our Intellectual Property & Technology Transactions attorneys.