

Software Patents: Why Should We Care?

If nature has made any one thing less susceptible than all others of exclusive property, it is the action of the thinking power called an idea, which an individual may exclusively possess as long as he keeps it to himself; but the moment it is divulged, it forces itself into the possession of every one, and the receiver cannot dispossess himself of it.

—Thomas Jefferson

Imagine a world in which Socrates not only originated the dialogue form of inquiry but also filed a patent claiming “intellectual property” rights for his invention and then vigorously enforced his patent against the Sophists to ensure the “purity” of his learning method. Imagine an Oxford scholar traveling to the University of Bologna in the thirteenth century, observing the “lecture” style of teaching, adapting it for use at Oxford, and then filing for patent protection. A cross-licensing agreement permits Bologna faculty to use the lecture format, but Cambridge University faculty are locked out. Imagine an MIT faculty member today taking out a patent for a novel application of “game theory and team-based problem-solving techniques” in physics laboratory instruction. A dispute soon erupts when Harvard claims that the underlying idea of the patent was derived by the faculty member when she was previously employed at Harvard. Finally, imagine a world in which individual faculty, educational institutions, and for-profit companies rush to stake out patent claims for every conceivable teaching style and method, competing to develop the largest and most aggressive patent portfolios, based in part on the ingenuity of the faculty and in part on a series of

strategic acquisitions and cross-licensing agreements.

These imaginary examples from a parallel universe likely strike us as fanciful if not absurd, for in our world, teaching methods are simply not patentable. *Or are they?* As the world becomes increasingly digital, as more educational technology tools become commonplace, as more faculty utilize cyberspace for some part of their teaching, and as the boundary between teaching in the traditional classroom and teaching online begins to blur, our intuitions about property, collaboration, and academic freedom will be put to the test. In our universe—and not just in the parallel one I have described—there is a rush under way to carve out and “own” the basic building blocks of ideas that enable all educational technology. The prospector’s gold this time around is the software patent.

Software and Intellectual Property

Let’s quickly review the software development process in order to understand where intellectual property enters into the picture and how the case law has evolved radically during the past decade. The basic process of developing software is similar to that of building a house. Step one in building a house is the idea phase: specifying the house’s design. The design, in turn, is used to construct a building, which represents the physical embodiment or expression of the architectural idea. Finally, a set of laws governs the appropriate use of the building.

All software, as a first step, originates with an idea or conception. As a second step, the idea is given *physical expression* in the form of computer code. And as the third and final step, the author of

the code attaches a license governing the code’s distribution, use, and modification. These three basic steps underlie all software development.

Until very recently, the assertion of ownership entered into the picture only during the second and third stages of the software’s lifecycle. Copyright, which is one form of intellectual property protection, applies only *after* a programmer has given the design ideas physical expression in the form of code. You and I could take the same design ideas as a starting point and program them independently. I would then own my code, and you would own your code, but neither of us would own the ideas on which our respective software was based. The code—the *physical expression* of the ideas—was and still is protected by copyright. If I were to steal your code, I would be in violation of the copyright. The ideas and methods, however, were until recently freely available to all.

Software patents, which protect ideas and methods, are a relatively new development. The U.S. Patent and Trademark Office (USPTO) had previously been reluctant to grant patents for software. The case law began to change radically with a set of Supreme Court rulings in the 1980s and a federal circuit court case in 1998 (*State Street Bank & Trust Co. v. Signature Financial Group*). These court rulings opened up the software patent floodgate. More specifically, the *State Street Bank* case upheld the notion that “business methods”—that is, *ideas*—are patentable subject matter.

This shift in the case law means that the ideas and methods underlying software can now be protected and assigned ownership. The *Blackboard Inc. v. Desire2Learn Inc.*

