In Part 1 of this column, I examined the recent Supreme Court case of Alice Corporation Pty. Ltd v. CLS Bank International and the underlying question of whether patents can be issued on computer-related innovations as well as on business methods and software. In this column, I conclude by describing in more detail the Supreme Court's Alice decision, the industry's reaction, and the apparent consequences for both software and business-method patents.

The Alice decision in the Supreme Court

In interpreting section 101, which defines which kinds of thing can be patented, the Alice Court began, "we must distinguish between patents that claim the building blocks of human ingenuity and those that integrate the building blocks into something more, thereby transforming them into a patent-eligible invention." The former patents preempt others from exploiting the building blocks; the latter do not.

The Mayo framework

The Court looked back to Mayo for "a framework for distinguishing patents that claim laws of nature, natural phenomena, and abstract ideas from those that claim patent-eligible applications of those concepts." The Court sought a single, unifying legal analysis for determining "whether the claims at issue are directed to one of those patent-ineligible concepts," and found it in Mayo. The first step was to determine whether the claim seemed prima facie to fit into one of those patent-ineligible categories. If so, the Court continued, we must then, as a second step, consider the elements of each claim both individually and "as an ordered combination" to determine whether the additional elements "transform the nature of the claim" into a patent-eligible application.

To do that, it is necessary to search for an "inventive concept"—i.e., an element or combination of elements that is "sufficient to ensure that the patent in practice amounts to significantly more than a patent upon the [ineligible concept] itself." The Court identified the underlying abstract idea as that of intermediate settlement. Alice denied that intermediate settlement was an abstract idea. It argued that the rule against patenting abstract ideas applied only to eternal verities—"preexisting, fundamental truths" that "exist in principle apart from any human action." The Court made short shrift of that, pointing to the risk hedging in Bilski that the Court had found to be a patent-ineligible abstract idea. Hedging is simply "a method of organizing human activity, not a 'truth' about the natural world 'that has always existed,'" the Court insisted, and so too is intermediate settlement. Just as hedging was properly deemed an abstract idea, intermediate settlement is an abstract idea and therefore patent ineligible.

The Court refused to try to describe the exact boundaries of abstract ideas. It was enough that intermediate settlement was as abstract as hedging: "[W]e need not labor to delimit the precise contours of the 'abstract idea' concept in this case." (This refusal to speak more broadly exasperated many observers, as I will discuss later.)

Does a computer add enough?

The Court then turned to whether Alice had added something to the abstract idea that would "transform" that abstract idea into a patent-eligible invention. Adopting the framework that Mayo used, the Court asked what Alice had added. Apparently, it added nothing to the idea of intermediate settlement but the further idea of using a computer to do the necessary bookkeeping. That brought the Court to the crux of the case. Did just the use of a computer suffice to establish patent eligibility?

The Court recalled that in Benson, one of the claims called for use of a computer to execute the algorithm. That was insufficient:

But the computer implementation did not supply the necessary inventive concept; the process could be "carried out in existing computers long in use." We accordingly held that simply implementing a mathematical principle on a physical machine, namely a computer, is not a patentable application of that principle.
What's a Nongeneric Computerization?

The Court did not indicate what could constitute a nongeneric computerization. Is there such a thing? Perhaps, that might be a computer-implemented method in which the programmed computer cooperated with its environment or with the other elements of the claim in a novel and unexpected manner. But could that occur? Computer science makes the results of computer programming predictable, and therefore, expectable.

The Court found Flook to be of the same effect. There a computer implemented the idea, but the implementation "was purely conventional." In Diehr, the computer-implemented process was held patent eligible, but not because a computer was used. Rather, it was the novel use of thermocouples inside the rubber mold to provide temperature measurements continually that the computer used to calculate cure time, which "transformed the process into an inventive application of the formula." Moreover, "the claims in Diehr were patent eligible because they improved an existing technological process, not because they were implemented on a computer." In sum, the cases from Benson to the present "demonstrate that the mere recitation of a generic computer cannot transform a patent-ineligible abstract idea into a patent-eligible invention." (By the terms "generic computer" and "generic computer implementation," the Court refers to conventionally programming a conventional general-purpose digital computer. A desktop PC is an example of a generic computer in this sense.) Limiting a patent claim to practice of the invention with a computer, without "something extra," amounts merely to saying, "apply the abstract idea on a computer." The Court said that cannot impart patent eligibility. It just piles one zero onto another zero, and $0 + 0 = 0$. (For more information, see the "What's a Nongeneric Computerization?" sidebar.)

Conventional computerization does not avoid preemption

By the same token, the Alice Court explained, this conclusion fits right in with the concern about preemption "that undergirds our [section] 101 jurisprudence." It does so for two reasons. First, because of the ubiquitous use of computers in modern business and industry to carry out tasks more cheaply, quickly, and accurately, a generic computer limitation is no limitation at all, in a practical sense. Thus, in the Benson case, the Court dismissed the computer limitation in one claim as without practical significance, because the claimed algorithm "has no substantial practical application except in connection with a digital computer, which means that if the judgment below is affirmed, the patent would wholly pre-empt...and in practical effect would be a patent on the algorithm itself." Second, again because of the ubiquity of computers, a legal rule that allowed wholly generic computer implementations to evade the rule against patenting ideas would provide a facile blueprint for patent drafters to monopolize abstract ideas. As the Court explained:

An applicant could claim any principle of the physical or social sciences by reciting a computer system configured to implement the relevant concept. Such a result would make the determination of patent eligibility depend simply on the draftsman's art, thereby eviscerating the rule that laws of nature, natural phenomena, and abstract ideas are not patentable.

Applying these legal principles to Alice's method, the Court found every step of the claims at issue to be "purely conventional." The functions and instructions were "well-understood, routine, conventional [and] previously known to the industry," just as they were in the Mayo case. Examining the steps separately or in combination, the result was the same: "a generic computer perform[s] generic computer functions."

The Court indicated, however, what Alice's claims did not do, thus suggesting what might pass muster in another case:

The method claims do not, for example, purport to improve the functioning of the computer itself. There is no specific or limiting recitation of... improved computer technology... Nor do they effect an improvement in any other technology or technical field.

The system claims and media claims fared no better than the method claims. They too recited generic computer components configured to implement the same abstract idea of intermediated settlement. They too, therefore, added "nothing of substance to the underlying abstract idea" and accordingly "they too are patent ineligible under [section] 101." The Court summarily dismissed the "piano roll blues" argument:

As to its system claims, petitioner emphasizes that those claims recite specific hardware configured to perform specific computerized functions. But what petitioner characterizes as specific hardware...is purely functional and generic. Nearly every computer will include [such equipment]. As a result, none of the hardware recited by the system claims offers a meaningful limitation beyond generally linking the use of the [method] to a particular technological environment, that is, implementation via computers.

Put another way, the system claims are no different from the method claims in substance. The method claims recite the abstract idea implemented on a generic computer; the
system claims recite a handful of generic computer components configured to implement the same idea. 13

The concurring opinion

Justices Sotomayor, Ginsberg, and Breyer agreed that Alice simply claimed a patent-ineligible abstract idea, and therefore they concurred in the main opinion. They would have preferred, however, to rest the judgment on other grounds—namely, that no suggestion existed in the English case law preceding and contemporaneous with the adoption of the American patent system “that processes for organizing human activity were or ever had been patentable.” 13 Since they considered Alice’s intermediate settlement technique to be a process for organizing human activity, rather than an industrial or technological expedient, they regarded it as something outside the patent system—something that the patent system was not designed to protect.

What the Supreme Court did and did not hold

The opinion is rather narrow in that it makes no effort to answer in detail the ambitious (and important) question that the Federal Circuit posed itself for en banc consideration: When, if ever, does the presence of a computer in a claim lend patent eligibility to an otherwise patent-ineligible idea?

The Court considered Alice’s patent to be “squarely within the realm of ‘abstract ideas’ as we have used that term” in such cases as Bilski. Accordingly, “we need not labor to delimit the precise contours of the ‘abstract ideas’ category in this case.” 17

The Court does suggest some factors, however, that help resolve some cases involving computer implementation of an otherwise abstract idea, law of nature, or method of organizing human activity (as in a business-method patent).

The unified field theory

The Alice opinion confirms that the patent-eligibility analysis of Mayo for laws of nature and natural phenomena is also to be used for determining patent eligibility of claims involving abstract ideas and methods of organizing human activity. A single legal theory unifies the several exceptions to patentability based on the nature of the claimed subject matter. This theory applies to algorithms and software implementations of inventions, although the Court did not expressly address software (indeed, it studiously avoided mentioning software).

Additionally, the Court emphasized that it would not interpret section 101 any differently for method or apparatus claims or permit patent eligibility to depend on the patent draftsmen’s art. This could signify that the piano roll blues has played its last riff. The Court said: “[T]he mere recitation of a generic computer cannot transform a patent-ineligible abstract idea into a patent-eligible invention,” 8 and that a “configured” computer adds “nothing of substance” to an abstract idea. 13

The nutshell

Using a computer to implement an abstract idea, law of nature, or way to organize human activity in a trivial or purely conventional manner will not confer patent eligibility. But using a computer to improve a manufacturing process or other technological process in an inventive way could well confer patent eligibility, despite the existence of an underlying abstract idea or principle as a component of or basis for the invention.

Two elements of that nutshell statement beg for amplification.

First, what processes other than manufacturing or industrial processes are “technological” is unclear. Judging by the questioning at the Alice oral argument (see www.supremecourt.gov/oral_arguments/argument_transcripts/13-298.896.pdf and www.oyez.org/cases/2010-2014/2013/2013.1.298), encryption, data-compression processes, and optimizing transmission of data packets might be deemed technological and patent eligible. Software that improves a computer’s internal operation (perhaps, for example, an improved BIOS or other computer interface for an operating system) might also be patent eligible. That might mean that the programming used could be conventional without any patent-eligibility problem, but the claims would still have to be “nonobvious”—that is, display an inventive step—in order to pass muster under the rest of the patent statute.

The Constitution and traditional portions of the patent statute do not define or even use the term “technological.” Some recent patent code amendments do use the term in a new section providing for post-issuance review of “covered business methods,” 14 but the use of the term in that new, specialized part of the statute is not instructive for purposes of cases such as Alice and Bilski. The Constitution uses the term “useful arts” but does not define it. Nonetheless, the term has some historical background, and there is some uncertain relationship between technology and the useful arts; it has been argued that business methods, being nontechnological, are not within the useful arts. 15 It is to be hoped that the courts will develop a principled basis for defining technological processes—something better than “I know it when I see it.” The Alice opinion leaves that task to future litigation.

Second, what is an inventive concept or inventive way to implement an idea? It will likely involve greater creativity than that of a person of ordinary skill. The requirement that something “significantly more” must be added to an abstract idea to confer patent eligibility probably means the same thing as implementing the idea in an inventive way. But the Alice Court said little or nothing about what is “significantly more”—or “something extra,” as the Court put it—other than that it must not be merely “generic computer implementation.” In Flook, the Court said:

Even though a phenomenon of nature or mathematical formula may be well known, an inventive application of the principle may be patented. Conversely, the discovery of such a phenomenon cannot support a patent unless there is some other inventive concept in its application. 16
This passage suggests that the application of an idea or principle must embody an inventive step, which seems to require that the application not be obvious in the sense of section 103 of the patent act. But the obviousness relevant to a section 101 patent-eligibility inquiry might differ from that relevant to a section 103 inquiry. Arguably, different policies are at stake. In the long run, however, courts (particularly trial courts) are likely to assimilate the concepts of "not generic," "significantly more," and "inventive concept" to one another, leading to a single standard—that of an inventive step or nonobviousness. But that remains to be resolved by future litigation.

Business methods

The Court once again refused, as it did in *Bilski*, to hold business methods categorically patent ineligible. But considering the nature of most financial or other business devices (such as, for example, hedging or mitigating settlement risk by using an intermediary, in the *Bilski* and *Alice* cases), most business methods will be held patent ineligible under the *Flook-Mayo-Alice* rationale. That is because their computer implementation is ordinarily conventional or even trivial. Rarely do business-method patents disclose an advance in programming technique. Often they involve routine, or at least un inventive, computerization of a known business expedient (as in *Bilski* and *Alice*), producing only expectable benefits—or, at the very best, a routine computerization of a new business expedient.

In its recent business-method cases, *Bilski* and *Alice*, the Supreme Court defined the abstract ideas of the patents in suit at a high level of abstraction and generality. That made it easy to find the idea an old one long in common use. In principle, establishing that is unnecessary for the analysis, for the legal theory applied here purports to cover $E = mc^2$ (highly novel, initially not in common use) in the same manner as hedging and escrow (very old, long in common use). Yet, to characterize the res of the prospective patent monopoly as public domain long in common use but now under a threat of enclosure does tend to tilt the perceived equities. In any event, operating analytically at a high level of abstraction, as the Court did, makes it easier to find an underlying abstract idea—which is the first step in the analysis *Alice* prescribes.

Some might say that this is as if one said the idea (or plot) of *Romeo and Juliet* is that a boy and girl love one another but encounter personal difficulties because of their parents' mutual dislike. That doubtless overstates the case, but some subjectivity inevitably colors the analysis, as well as some result orientation. By the same token, defining the underlying idea at a highly specific level would shrink the scope of the abstract idea doctrine to negligible proportions. There is no way to fix on a golden mean. The proof of the pudding is only in the eating. Therefore, we ask: What are the consequences of the Court's approach? The Court is concerned that industrial progress will be hindered by monopoly grants that are too broad. It is less concerned that ruling out patents on abstractions will hinder innovation, because it does not believe that will happen. Past experience suggests that innovation has not been discouraged by the failure of the patent system to give out patent monopolies for new ideas. We got $E = mc^2$ and the Laplace transform without giving Einstein or Laplace patents for them.

One consequence, as already remarked, is that many business methods will be deemed insignificantly different from their underlying abstract ideas and therefore patent ineligible. Another consequence is to tilt doctrinal evolution of patent-eligibility law away from any focus on what is in the useful arts, as the Constitution uses that term. Because the *Alice* Court has made it easier to find an underlying abstract idea, it has become largely unnecessary to develop a doctrine that business methods are categorically excluded from the patent system because they are not within the useful arts. In large part, the abstract idea doctrine has made a useful-arts-based doctrine superfluous, because the abstract-idea doctrine now suffices to cover the same ground—indeed, to eliminate many or most business-method patents.

Nevertheless, there are unusual cases—outliers—in which, despite the intuitive categorical inappropriateness of any patent, the abstract-idea doctrine is ineffective to curb a bizarre patent. For example, the machine implementation may not necessarily be trivial or conventional. Or if conventional, the implementation may clearly be a new use. The machine for exercising cats and the notional machine for making a punishment fit the crime (discussed in Part I of this column) may illustrate the point. No such case has yet come to the Court.

The Supreme Court's post-*Alice* business-method rulings

On 30 June 2014, the last day of the October 2013 term, about a week after the *Alice* decision, the Court disposed of three certiorari petitions that had been pending the *Alice* decision. In one case, the Federal Circuit had upheld the patent eligibility of a scheme for compensating Internet users for viewing advertising material by allowing them to view copyrighted films, books, and songs, where the advertiser paid the royalty due for viewing the media products. The Federal Circuit had said that the scheme required elaborate programming, although the programming was not said to be unconventional, and certainly not inventive. The facts otherwise seemed similar to those in *Alice*. The Supreme Court vacated the judgment and remanded the case to the Federal Circuit for reconsideration in light of the *Alice* decision.

In a second case, the Federal Circuit had held patent ineligible a scheme for generating tasks to be performed in an insurance organization, which required a computer system to update a database when an insurance-related event occurred. The Federal Circuit decided that case under a methodology very similar to that prescribed in the *Alice* opinion—generally following the *Mayo*
paradigm. Significantly, the Federal Circuit had said in this second case that "simply implementing an abstract concept on a computer, without meaningful limitations to that concept, does not transform a patent-ineligible claim into a patent-eligible one." The Supreme Court denied certiorari, leaving the judgment of patent ineligibility undisturbed.

In a third case, the Court declined to review a Federal Circuit ruling that a scheme for "managing a stable value protected life-insurance policy," again involving storage and data manipulation by computer, was patent ineligible. The Federal Circuit agreed with the trial court's determination that the abstract idea of the patent was drawn to "concepts for organizing data rather than to specific devices or systems." The claims did not distinguish what the patent claimed from the abstract idea itself, so that the patent was one on the underlying idea. Again the court decided the case using a methodology very similar to the one the Supreme Court used in the Alice case. The patent owner tried to support patent eligibility by pointing to complex implementing details in the specification of the patent, but the court pointed to their total absence in the claims. It said, "In other words, the complexity of the implementing software or the level of detail in the specification does not transform a claim reciting only an abstract concept into a patent-eligible system or method." Again, the Supreme Court denied certiorari, leaving the judgment of patent ineligibility undisturbed.

Apparentely, the Supreme Court is unimpressed by a claim that a computer program is complex, unless it also displays an inventive step. That is likely to be a difficult criterion to satisfy. Probably, a claim (as made in the first of these two cases) that the scheme and the computer program make available new functions will not succeed in conferring patent eligibility if, as is likely, the added functionality will either be found to be what computers conventionally do or else be attributed to the supposedly abstract idea and not to the otherwise routine programming that implemented provision of the asserted functionality.

Software

One of the dissenting opinions in the Federal Circuit in the Alice case had predicted that the decision, if affirmed, would be "the death of hundreds of thousands of business method, financial system, and software patents." This prediction alarmed many stakeholders in such patents. But in the Alice case, the Supreme Court avoided any pronouncements on software in general, and just called the computer programming for Bilski's hedging patent and Alice's settlement risk mitigation patent conventional and routine, and therefore ineligible for patenting. During oral argument, Justice Sotomayor repeatedly asked the following questions:

- "Why do we need to reach-software patents at all in this case?"
- "What's the necessity for us to announce a general rule with respect to software? There is no software being patented in this case."
- "Do you think we have to reach the patentability of software to answer this case?"

The opinion of the Court reflects that caution, which calmed the fears of many of the restless natives. To the extent that there is a division of opinion about Alice, it appears to reflect a difference in interest among stakeholders. Those with a stake in business-method patents, many of whom do not operate businesses other than asserting and licensing patents, are disturbed and disappointed. Those whose major interest is selling products and services (typically, electronic or Internet-based), and who are concerned about being sued by the first group, are pleased.

The Flook-Mayo-Alice rule would not seem to threaten the patent eligibility of those software patents that are not mere routine computerizations of preexisting business or financial expedients. By implication at least, the Alice opinion leaves room for patents on software that improves technological and industrial processes. Software on the internal functioning of computers would also appear patent eligible. Although the opinion did not say so, patents on software for encryption or data compression, if not deemed simply mathematics (which begs the question), would probably remain patent eligible. (In a future Micro Law column, I will address encryption and discuss whether patents like that on RSA encryption are just patents on performing mathematical calculations, and how that affects their patent eligibility.) Such a device is not "a method of organizing human activity," which both opinions deprecated in regard to patent eligibility. The Alice opinion made a point of adding, though, "There is no dispute that...many computer-implemented claims are [patent-eligible]." The Court offered no specifics, however. This issue too must therefore await further litigation and further incremental clarification.

Aftermath in the Federal Circuit

In its first patent-eligibility decision after Alice, on 11 July 2014, the Federal Circuit invalidated a patent on processing digital graphics data. The Court stated that the patent "claims an abstract idea because it describes a process of organizing information through mathematical correlations and is not tied to a specific structure or machine." The claimed method steps were generating two sets of data that described color information for a digital image, and then combining the two datasets according to an algorithm. The court stated that the "claim thus recites an ineligible abstract process of gathering and combining data" by using "mathematical algorithms to manipulate existing information to generate additional information," and that "is not patent eligible." The proper legal principle, the court said, is: "A claim may be eligible if it includes additional inventive features such that the claim scope does not solely capture the abstract idea." The opinion was quite terse, but it indicated that the
What is Stare Decisis?

Stare decisis is the doctrine or policy that courts should follow precedents set by previous court decisions in cases with similar issues. This policy is followed in common law systems because it creates predictable outcomes and therefore greater certainty and consequent security of business and personal expectation. The phrase is an abbreviated form of "stare decisis et non quieta movere," which can be translated roughly as "to stand by previous decisions and not make waves." Sometimes, this rule is followed even when a decision later seems incorrect, on the ground that it is better to have things settled and stable than to fine-tune their correctness. The US Supreme Court has said that "we will not depart from the doctrine of stare decisis without some compelling justification." Interestingly, a policy of following prior decisions and making few and only incremental changes can make the development of the law highly path-dependent, so that the order in which cases (that is, disputes with particular fact patterns) reach the highest court affects the ultimate rule or at least the state of the law at any point.

Reference

The Alice decision is incremental. It confirmed the line of decisions since Bilski, but did not go appreciably beyond what the cases immediately preceding it held or implied. Its application of the rationale of Mayo to computer-implemented business methods should not have been surprising; it should have been expected. Nonetheless, it was important to spell that out.

Beyond that, the fact that the Alice opinion provides no detailed guidance on when computer implementation is effective to confer patent eligibility is consistent with the recent course of Supreme Court decisions in this area, and should therefore be no surprise, at least to careful observers. For example, Professor Duffy said that "the Supreme Court has been remarkably resistant to providing clear guidance in this area, and this case continues that trend." But that reticence has disappointed many who had hoped, first in Bilski and then in Alice, for clear, categorical statements. For example, Professor Merges said, "To say we did not get an answer is to miss the depth of the non-answer we did get." Indeed, Mayo's relatively more expansive discussion (in comparison with the other opinions) whetted their appetites and stirred their aspirations. But now, some commentators deplore the Court's series of narrow rulings and fear that they will never see a clear ruling on software or business methods from the Court. They fear that the Court will just keep on issuing narrow rulings on a case-specific basis out of anxiety lest a broad, informative statement from the Court have unintended, negative economic consequences, dash investor expectations, or even stifle innovation. I suggest elsewhere that such fears caused a flip in the Bilski case and turned Justice Stevens's five-Justice majority opinion condemning all business-method patents into a four-Judge separate opinion disagreeing with the majority's rationale.

Some critics were particularly exasperated by Justice Thomas's characteristic refusal to engage in development of theoretical infrastructure for the judgment or Socratic exploration of hypothetical cases to test the limits or soundness of the legal ruling. To them, the high point (or perhaps low point) of all that in Alice was his statement—"we not labor" to spell out what is or isn't too abstract for patent eligibility;

In any event, we need not labor to delimit the precise contours of the "abstract ideas" category in this case. It is enough to recognize that there is no meaningful distinction between the concept of risk hedging in Bilski and the concept of intermediated settlement at issue here. Both are squarely within the realm of "abstract ideas" as we have used that term.

One commentator responded bitterly to this that Justice Thomas had "balked at the messy, challenging issues surrounding software" and had provided "not a particularly useful" decision about patenting software. The comment featured a cartoon in which the protagonist (depicted as Lewis Carroll’s Alice) wagged a finger at Justice Thomas for saying "we need not labor to delimit" and replied to him "yes you do!" The cartoon argued that the Court did not explain in Bilski what made something too abstract and now just said that Alice's patent is as abstract as the patent in Bilski "where we never defined abstract—SIMPLE!"

Several responses to that are appropriate. First, the narrow opinions in these cases are the expectable price of unanimity in a nine-member tribunal. The costs of uncertainty over specific case patterns may well be outbalanced by greater sensed legitimacy and precedential stability—the benefits of stare decisis (see the "What is Stare Decisis?" sidebar for more information).

Second, there is an observable trend line in these cases. Using an appropriate algorithm (pace Flook) to smooth the data points, we can discern a clear trend line in the patent eligibility cases from the 1970s to 2014. Based on that trend line, we can predict that few if any present business-method patents will survive an abstract-idea analysis.

As for dashing settled business expectations, building business expectations on legal theories that run counter to a long-term trend of Supreme Court
In copyright law, the "abstractions" or "patterns" or "concentric circles" analysis is often used for distinguishing idea from expression in an infringement analysis. When using this analysis, the zone of expression (say, the center of the bull's-eye in Figure A or the circles near it) and the zone of idea (say, the outer circles of the target) in a work are differentiated by the magnitude of the body of limitations present, in terms (for example) of plot and characterisation. Thus, a play simply about a rotund knight and a lippish steward will not infringe Twelfth Night, because the shared pattern is too abstract. But as further limitations are added—for example, the knight likes practical jokes, the steward is persuaded to write love notes and wear crossed garters, the steward is thought mad and is locked up in a dark room, and so on—the characters more closely approach Sir Toby Belch and Malvolio. Now the second play, more closely approaches Twelfth Night, and at a certain radius from the bull's-eye, the second play becomes infringing.

Consider Nichols v. Universal Pictures Corp.1 and Sheldon v. Metro-Goldwyn Pictures Corp.2 In the Nichols case, Judge Hand explained the abstractions principle in these terms:

Upon any work...a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the play is about, and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the playwright could prevent the use of his "ideas," to which, apart from their expression, his property is never extended.

But moving from theory to application is problematic. There is no general theory of how to locate the correct radius. In Peter Pan Fabrics v. Martin Weiner Corp.3 Judge Hand's last copyright opinion, written shortly before his death, he opined:

Obviously, no principle can be stated as to when an imitator has gone beyond copying the "idea," and has borrowed its "expression."...We cannot say how far an imitator must depart from an undeviating reproduction to escape infringement.

decisions does not create a vested interest. It would be absurd to subordinate the public interest in order to bail out those who adopt unrealistic and misguided investment strategies.

Finally, it is sensible to make narrow, incremental rulings as to software patent eligibility, because at present we are not so well-informed that we can speak with confidence in very broad terms. Which software patents, if any, are consistent with both encouraging software progress and avoiding its preemption remains to be seen. We might feel assured that software patents, and certainly all business-method patents, are a plague that ought to be extirpated, but we should remember Cromwell's entreaty to "think it possible that you may be mistaken."4 Therefore, recourse to the laboratory of case-by-case development is a prudent expedient.

Providing clear, precise, categorical demarcation for the concept of abstract ideas might be a hopeless task. There is a similar problem in copyright law—distinguishing between ideas and expressions of ideas (see the "Idea and Expression in Copyright Law" sidebar). Just as, in the words of Judge Hand, "no principle can be stated as to when an imitator has gone beyond copying the 'idea,' and has borrowed its 'expression,'" it could well be true that in patent law no principle can be stated as to where the frontier of abstract idea is. In any case, a
satisfactory such principle has yet to be articulated.

References
3. Alice, 134 S. Ct., at 2355.
4. Alice, 134 S. Ct., at 2355, quoting Mayo.
5. Alice, 134 S. Ct., at 2356.
6. Alice, 134 S. Ct., at 2356-57.
7. Alice, 134 S. Ct., at 2357.
8. Alice, 134 S. Ct. at 2358.
11. Alice, 134 S. Ct. at 2359.
12. Alice, 134 S. Ct. at 2359-60.
13. Alice, 134 S. Ct. at 2360.
20. Accenture, 728 F.3d at 1346.

Richard H. Stern is an attorney at Kellogg Huber Hansen Todd Evans & Figel and a law professor at George Washington University Law School. Contact him at rstern@khhte.com or rstern@law.gwu.edu.

Selected CS articles and columns are also available for free at http://ComputingNow.com.