

**From:** Jérôme Verstrynge [e-mail redacted]  
**Sent:** Sunday, September 26, 2010 11:51 AM  
**To:** Bilski\_Guidance  
**Subject:** Contribution following press release 10-35

Dear USPTO,

Here is a contribution in attachment following press release 10-35. I believe there is a possibility to create a sub-category of patents, such that the category itself would be unpatentable as an attempt to patent abstract ideas. This would cover issues raised by the Bilski case.

I plan to post this contribution on my blog (<http://adamman71.blogspot.com/>) for general exposure to the Internet community too.

Best regards,

Jérôme Verstrynge

Dear USPTO,

Following your press release 10-35, I would like to make a contribution to help solve the issues raised by the Bilski case. I believe it is possible to define a sub-category of patents *'such that the category itself would be unpatentable as "as an attempt to patent abstract ideas"*. This definition would exclude Bilski-like applications while allowing business method patents, including some software-like patents.

Firstly, I will define a set of issues to cover to create such a sub-category.

There are currently several items which can lead to confusion:

- i.1) the meaning of *'abstract ideas'*
- i.2) the lack of a clear border between *'business method'* and *'abstract ideas'*
- i.3) the lack of a clear border between *'software'* and *'abstract ideas'*
- i.4) the meaning of *'tangible'*

Other thicket items include questions raised by Judge Mayor:

- i.5) *'what constitutes sufficient physical transformation or machine-implementation to render a process patentable?'*
- i.6) *'the extent to which computer software and computer-implemented processes constitute statutory subject matter'*

Other areas of concern are:

- i.7) the potential broadness of interpretation of claims
- i.8) the temptation to reject all software-like patents, since software is abstract by nature
- i.9) the lobbying of some members of the IT industry against patents in general

Secondly, I would like to raise a couple of facts:

- f.1) claim 7 (b) of the Bilski application mentions: *'performing a Monte Carlo simulation across all transactions at all locations for a predetermined plurality of years of weather patterns and establishing the payoffs from each transaction under each historical pattern'*
- f.2) there is no such thing as a unique Monte Carlo method
- f.3) *'Monte Carlo'* is a methodology; it is a set of abstract methods and algorithms helping to deal with uncertainty
- f.4.a) all methodologies require human intervention (i.e., interpretation, definition of meanings) to find out which methods or algorithms should be applied in a given context, when, why and how; we can call this process *'the application of a methodology'*
- f.4.b) the very process of applying a given methodology includes the possibility of rejecting this very methodology for a given context; sometimes it can prove to be inadequate, inapplicable or incomplete
- f.5) no computer is capable of applying a methodology (i.e. performing f.4.a and f.4.b) by itself
- f.6) methodologies, methods and algorithms can be called abstract ideas
- f.7) algorithms and methods can be implemented on various devices using several programming languages; the result is called a software application or implementation

f.8) methods and algorithms are a set of rules to follow to reach an objective or to obtain a result

f.9) the implementation of methods and algorithms is the process of translating a set of rules to follow into a set of code instructions which is understandable and executable by a computer device; it is called programming

f.10) translation is not interpretation

f.11) dictionaries provide multiple definitions for the word '*tangible*': something perceptible by the senses, something capable of being treated as a fact ('*tangible evidence*'), something having a physical substance or a monetary value, something palpable, something comprehensible by the mind

Thirdly, I would like to draw some first conclusions from the above:

c.1) From f.1 and f.3, we can conclude that Bilski's claim 7 (b) is an attempt to patent a methodology

c.2) From f.4.a, f.7, f.9 and f.10, we can conclude that a software application/implementation is not equivalent to a methodology application/implementation

c.3) From f.4.a and f.4.b, we can conclude that the value of a methodology is not immediate and not guaranteed; hence the value of a methodology is not immediately tangible

c.4) From f.4.a, we can conclude that the final product resulting from applying a methodology is not deterministic in that it depends of the context of application; the very nature of this final product is also uncertain; it cannot be computed in advance

c.5) From f.8, f.9 and f.10, we can conclude that the very nature of the final product resulting from programming a method or an algorithm well-identified; it is called a software implementation or application

c.6) From f.8 and c.5, we can conclude that the value of methods and algorithms is immediately tangible

c.7) From f.11, we can conclude that methods, methodologies and algorithms can be called tangibles; regarding methodologies, tangible items are the general principles to apply and the guiding principles for example; methodologies can be written down on paper, they are substantial

Fourthly, I would like to perform further analysis against the set of issues to cover:

- i.1: using the words '*abstract idea*' to identify '*a set of claims which cannot be patented*' has a drawback, since patents often include methods in their claims, and methods can legitimately be called abstract ideas
- i.2: in order to clarify the border between '*business method*' and '*abstract ideas*', one could find out whether the described business method or process is in fact a methodology – or – a concrete of set of rules which does not require any interpretation before implementation or execution
- i.3: regarding the border between '*software*' and '*abstract ideas*', one will not need to clarify this border, in the context of patent application, if one uses a terminology removing the potential confusion induced by the concept behind '*abstract ideas*' (we will call this requirement r.1)
- About i.4, the '*a process [is] patent-eligible if it [produces] "a useful, concrete, and*

*tangible result* test has been proposed and finally considered inadequate by Courts. Fact is, methodologies do not necessarily produce useful, concrete and tangible result as concluded in c.4. Hence, this test could be reinstated as part of creating our sub-category.

- About i.5, the answer could be: 'the absence of an intrinsic requirement necessitating a human intervention to determine the set of steps to perform in the context of application of the method or process'.
- Regarding i.6, if we take c.2 and c.5 into account, we can argue that the presence of a computer-implementable process could constitute substantial and tangible evidence that one is not dealing with an undefined product having an undefined and deferred tangible value.
- Regarding i.7, one should not confuse broadness of application with broadness of interpretation. Methods and algorithms can have a broad application through several translation processes, while methodologies can't without an interpretation step. Hence, rejecting methodology claims will resolve this concern.
- i.8: this issue is similar to i.3 discussed above; there would be no legitimacy in rejecting software patents globally because one is not able to differentiate the methods described in methodologies and the methods used in software implementation

About i.9, Judge Mayer has reported that the "patent system has run amok". The IT industry is suffering from patent trolls and the excessively defensive attitude leading many companies to patent inventions only to make sure they will not be attacked in return. To some extent, the system has gone dysfunctional, but it still helps companies protecting their investment in research, which is an absolute necessity to establish the context allowing a sustainable free market economy. Patents still support the creation of wealth.

Accepting the lobbying of some member of the IT industry to suppress software patents would actually create a context where the value of new inventions could not be protected anymore. Software parts are notoriously easy to copy and to replicate in other software applications. Hence, competitors would automatically benefit from any innovation for free. There would be no incentive to invest in innovation anymore. It would prevent the creation of future wealth.

The small benefit of not having to deal with software patent issues anymore will never match the loss of future potential wealth, which is necessary to create jobs and new products. It is not a valuable option on the long-term. By promoting the suppression of software patents, one does not address the damages this solution would cause to the economy, or the issues raised by the Bilski case. It is off-topic.

Fifthly, I would like to cover some remaining and legitimate concerns.

If one argues that methodologies can prove to be inadequate, inapplicable or incomplete in some situations (f.4.b), then one should be able to provide an example. Well, this example is already available: the current methodology to find out whether a claim is patentable or not. It has shown its limit with the Bilski case and ultimately led to the publication of press release 10-35.

One can argue that if the Bilski method is a methodology (f.1, f.2, f.3) and that if methodologies can be proven to be inadequate, inapplicable or incomplete when applied in some situations (f.4.b), then one must be able to explain how the Bilski method does not necessarily produce an immediate tangible value.

This can be established from two angles. The first angle is that Bilski is assuming one can model patterns based on weather related information to hedge investments. In other words, Bilski is assuming that using Monte Carlo will necessarily produce a valuable pattern in this case.

Unfortunately, large systemic systems (like weather forecasting and finance) do not always adopt a regular or identifiable behavioural pattern. Even if the equations describing the system are established with great scientific accuracy, iterating these equations to predict the behaviour of a given parameter might indicate that this parameter's value is perfectly unpredictable. Hence, the equations are not worth much in this case. We have no guarantee that the Bilski method is not going to study one of those parameters.

From another angle, there is a fundamental issue with the prediction of large systemic systems: the sensitivity to initial conditions also known as the butterfly effect. Any infinitesimal variation on the initial values used to iterate the model has the potential to generate completely different values after a couple of iterations. This explains why computer models cannot predict weather more than a couple of days in advance with useful precision. Since the Bilski method relies on temperature and that a measured temperature is never exactly 81° or 82° (for example), but any value in between, the Bilski modelling method is subject to the butterfly effect. Hence, it has the potential to generate useless values, and therefore unstable patterns.

By assuming that the '*summed payoffs are normally distributed*' in claim 7 (c), Bilski tacitly reveals that he/she does not understand the basics of establishing models. Before making such a claim, he/she would need to establish first that these summed payoffs are indeed normally distributed, which he/she does not. What if this distribution is not normal? How can one hedge investments with faulty or uncertain distribution patterns? What is the value of this invention if it relies on uncertainty?

This explains how the Bilski methodology does not necessarily produce immediate tangible value. If it does not necessarily produce value, then how can it be a valuable innovation? Hence, why should it be patentable if it does not necessarily work in its area of application?

To conclude, I would like to propose a solution to identify a sub-category of patent applications which the USPTO should automatically reject based on the arguments made above.

1. Stopping using the '*abstract idea*' terminology and start talking about '*concrete methods*' versus '*methodologies*' when discussing business methods or processes.
2. Substitute '*abstract idea*' for '*tangible results*' produced by methods or processes (with respect to f.11).
3. Use the following tests to discriminate patent application claims:
  - a. Do the claims explicitly attempt to patent a methodology? If yes, reject the patent application, since the innovation is intrinsically requiring a context and human interpretation before being applicable. How could a third party determine whether it is infringing the claims if it requires an interpretation?
  - b. Do the claims describe a set of rules which does not require human interpretation for application in a given context before execution (as opposed to simple translation of those rules)? If not, reject the patent application since the innovation is not readily usable and the domain of potential application is necessarily subject to interpretation.
  - c. Does the method or process produce a tangible result? If not, reject the patent application since its value cannot be used in practice.
  - d. When dealing with methods or processes related to software or algorithms, is there evidence that these can be implemented and executed on a computer device? If not, reject the patent application since such innovations should necessarily be implementable.

- e. Is the very nature of the product created by the method or the process identifiable in advance (i.e., before the method or process is applied)? If not, reject the patent application since it is not possible to identify the true nature of the innovation itself or its value.
- f. Does the product created by the method or the process have an immediate tangible value (as opposed to unpredictable, deferred or uncertain value)? If not, reject the patent application since the value of the innovation cannot be established.
- g. Is the area of application of claims immediately identifiable or computable regardless of interpretation? If not, reject the patent application since a third party cannot determine whether it is infringing the claims or not.
- h. Is there evidence that the method or process described in the claims will work or operate successfully in its area of application? If not, reject the patent application since it is not an innovation. Allowing the patent would prevent a third party from inventing another solution that covers all area of application.

With this set of guiding principles, business method patents would remain viable.

The Bilski case would be rejected by 3.a, 3.f and 3.h.

Software-related patents could still be accepted while filtering for ambiguities such as:

- abstract or incomplete definition of innovations
- abstract or intangible definition of the value of claimed innovations
- abstract or inapplicable descriptions of methods of implementation of claimed innovations
- abstract or non-deterministic areas of applications of innovations

Requirement r.1 would be covered by 1 and 2.

Best regards,

Jérôme Verstrynge  
(September 26<sup>th</sup>, 2010)