14   Computer-Implemented Inventions as Patentable Subject Matter [(*)]

Invited Researcher: Pessi Juho Antti Honkasalo [(**)]

In the past fifty odd years, software development has shown steady growth. It is now generally accepted that computer programs can attract patent protection, but examination of software inventions is difficult. Questions lacking a simple answer include, but are not limited to, when a contrivance that is implemented on the computer constitutes a patentable invention, when such subject matter can be deemed to be novel, what can be regarded as an inventive step in case of software and what is the eventual scope of the protection conferred by patent claims that are realised by a computer program.

This research addresses issues mentioned above by reviewing laws, examination guidelines and judicial precedents in the light of technically literate commentary. The territorial scope of the research encompasses the patent laws of Japan and Europe, both of which have been subject to international harmonisation in the form of multilateral agreements and treaties. Through the methods of doctrinal and comparative research, this study contemplates Japanese and European practices in this field and endeavours to analyse whether there are lessons to be learnt from the other jurisdiction. The Report concludes with a list of intra- and extra-systemic implications that need to be discussed or solved in the future.

Ⅰ Introduction

The patentability of computer programs has been the subject of considerable debate in post-industrial societies around the world. On the basis of previous studies as well as developments in case law and administrative guidelines it is possible to offer a comparative look into the present legal state with regard to computer-implemented inventions as patentable subject matter.

Ⅱ Patentability of Computer Programs

1   Patentable Inventions: Primer

To qualify as a patentable invention, there must first of all be patent-eligible subject matter. In Japan, patent-eligible inventions have been demarcated by a statutory definition: they must, pursuant to Section 2(1) of the Patent Act, be "creation[s] of technical ideas utilizing the laws of nature." Under the European Patent Convention (EPC), the legislature has chosen an opposite approach: Article 52(1) EPC provides that European patents are to be granted for inventions in all fields of technology without further defining the term "invention," but paragraph 2 of the same article then stipulates as an exception to the principal rule that certain subject matter and activities are not to be regarded as inventions in the aforementioned sense. At all events, irrespective of the structuring of the applicable regulatory framework, the element of invention is critical for any award of a patent. It is a treaty-based requirement under Article 27(1) of the Agreement on Trade-Related Aspects of Intellectual Property Rights (TRIPS) that signatories take care of it.

2   European Disinclination

The statutory prohibition against the grant of a patent for programs for computers is in accordance with Article 52(3) EPC restricted to cases where the application relates to subject matter falling under this category "as such." Accordingly, distinction has been drawn between a program for a computer, which *per se* cannot be regarded as an invention within the meaning of Article 52(1), and a computer programmed to operate in a particular manner, which as a technical apparatus satisfies the provision. From the vantage point of intellectual property law as a whole, this makes sense insofar as patent right is to protect the program's operation, in other words the behaviour that the software and hardware components bring about. The actual manner in

which this is done, in terms of literary expression, is protectable by copyright.

The *Vicom* decision (T 208/84) is the pivotal figure in the European journey towards the patentability of computer programs. It was held that "[g]enerally speaking, an invention which would be patentable in accordance with conventional patentability criteria should not be excluded from protection by the mere fact that for its implementation modern technical means in the form of a computer program are used. Decisive is what *technical contribution* the invention as defined in the claim when considered as a whole makes to the known art." [1] Subsequently the Technical Board of Appeal 3.5.01, which primarily deals with the relevant classifications, has in different compositions adopted divergent criteria in determining applications before them. Still in *IBM* (T 1173/97), the Board in question was asking whether the invention is machine-oriented and thereby makes such a technical contribution; if not, it was to be barred from patentability under Article 52(2)(c).

But in later cases, notably that of *Microsoft* (T 424/03), the Board has carved out a rule of simply asking whether there is a claim to something "concrete," for example a storage medium. If there is a reference to an apparatus, the *per se* exclusion does not apply. The President of the European Patent Office (EPO) tried to intervene under Article 112(1)(b) EPC. The Enlarged Board of Appeal held referral G 3/08 inadmissible but noted that, in consequence of a "legitimate development of . . . case law," a claim to a program on a computer-readable storage medium should now necessarily avoid exclusion from patentability on this point.

Ⅲ Novelty and Technical Effect

1 Legal Fiction of "Technical Contribution"

The Guidelines for Examination in the European Patent Office (hereinafter "EPO Guidelines") state that technical character as a fundamental requirement of patent law is implied in the term "invention." The Guidelines start by stating that any claimed subject matter, which defines or uses technical means, is able to pass this basic hurdle and be an invention within the meaning of Article 52(1) EPC. Even though the execution of a computer program always involves certain physical effects in the form of electrical currents and the like, such normal physical effects are, the Guidelines continue, not in

themselves sufficient to lend a computer program technical character. But if a computer program is capable of bringing about a further technical effect, which goes beyond the normal physical interactions between the program and the computer, it is not excluded from patentability.

It is not only the further technical effect, however, that can lend to technical character. The requirement of technical character may also be satisfied if technical considerations, which must be reflected in the claimed subject matter, are required to carry out the invention. If the claimed subject matter passes this prima facie test for technicality, the patent examiner ought then to proceed to questions of novelty and inventive step. In assessing whether there is an inventive step, the examiner must formulate an objective technical problem that has been overcome. The solution of that problem constitutes the invention's technical contribution to the art. The presence of such a technical contribution establishes that the claimed subject matter has a technical character. If no objective technical problem is found, the Guidelines explain, there can be no technical contribution to the art, and the claim is to be rejected on that ground. Thus, the enquiry recognises no less than the following parameters: technical character, technical means, technical effect, technical considerations, technical problem and technical contribution.

One can essay to map these variables onto a flowchart so as to piece together their systemic interactions. Figure III.1 below represents the process of examining a European patent application when it relates to a computer-implemented invention. The illogical fork that the process makes after the question of technical means signifies the difference between the technical-effect approach, laid down in *Vicom* and *IBM*, and the subsequently adopted "any hardware" approach, advocated for example in *Microsoft*, which removes much of the need to debate the existence of technical character and focuses on novelty and inventive step instead.
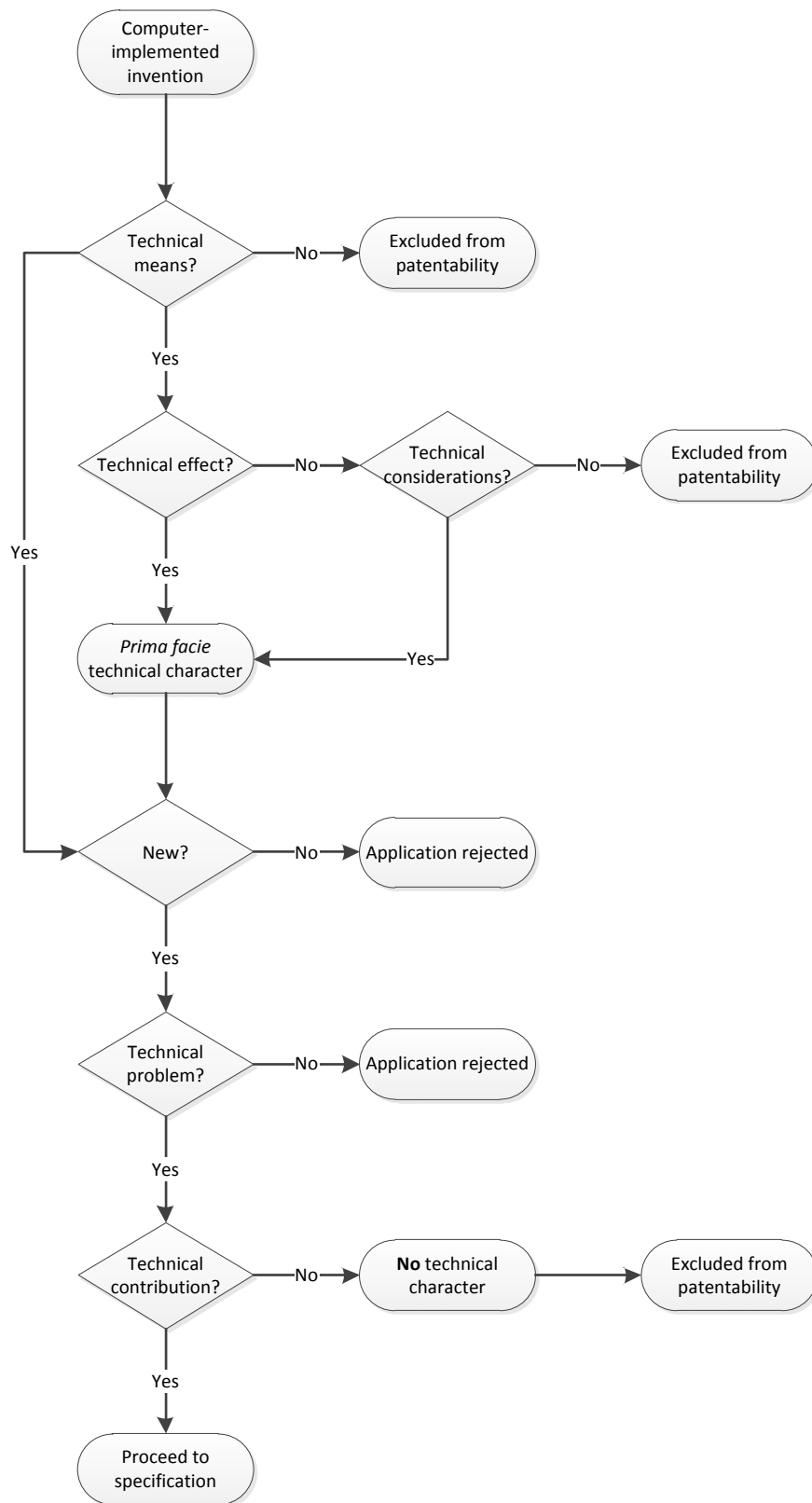
FIGURE III.1. EXAMINATION OF PATENTABILITY BEFORE THE EPO

2　Japanese Guidelines

In Japan, the statutory doctrine is interpreted so that where information processing by software is concretely realised by using hardware resources, said software is deemed to meet the statutory requirement of being an invention within the meaning of Section 2(1) of the Patent Act. Software inventions will be considered patentable subject matter if the claims

describe how the computer reads the software and uses it in combination with hardware in order to perform arithmetic operations or other manipulation of data. Practical utility is not enough, but a "technical feature" is the key element: the claimed software-related invention must be concrete enough to accomplish a certain purpose. To be eligible for a patent, the invention needs to utilise a computer so as to provide a *concrete means in cooperation with* the program or programs. The procedure, described in the Examination Guidelines for Patent and Utility Model in Japan (hereinafter "JPO Guidelines"), for judging whether or not an invention in this domain is capable of constituting a prima facie protectable invention has been imaged in Figure III.2 below.
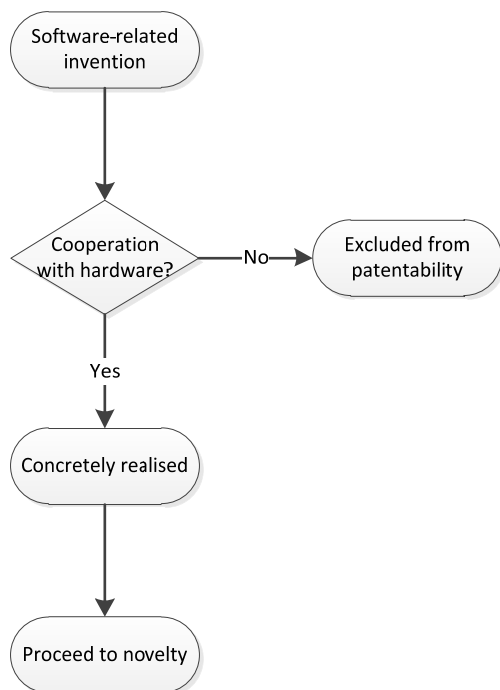

FIGURE III.2.
EXAMINATION OF PATENTABILITY BEFORE THE JPO

Not unnaturally, all enquiries like these are, in the ultimate analysis, open to various interpretations and require discretionary judgment on the part of the examiner. But as compared to Europe and the system constructed by the EPO Boards of Appeal, which requires the examiner subjectively to determine whether the contribution of the claim to the prior art lies in a technical area or an area excluded by Article 52 EPC, the JPO Guidelines provide a more concrete and unambiguous treatment of patent claims related to software.

3 Examination of Prior Art

It has been said that information technology is an area in which it is particularly cumbersome for the examiner to locate prior art. In addition to patent documentation, unpatented research and commercial developments – in the form of journal and other written materials as well as proprietary and open-source software on offer – are of utmost importance in these fields. In Japan, the National Center for Industrial Property Information and Training (INPIT) operates a database called IPDL, which incorporates the "Computer Software Database." The latter contains non-patent literature related to software, indexed and cross-referenced, labelled with keywords and archived with summaries, but only in Japanese.

The problem of locating prior art however is not the only, and perhaps not even the most difficult, challenge in assessing novelty in case of computer-implemented inventions. The structure of computer programs exists only in terms of their operational environment and their operational dynamics. It follows that it is possible to delineate a program on multiple levels of abstraction in many ways, all of which are equally true. Owing to this complexity, it is far from trivial to conclude whether a software-related invention is really novel or just another way of expressing the same idea that many a software developer has utilised before. Comparing source code with patent claims can be well nigh an insurmountable exercise.

It has been expounded that "crowd sourcing," harnessing the input of so-called "citizen experts" in obtaining prior art material, might offer a solution to some of these problems. The Japan Patent Office (JPO) entrusted the IIP with carrying out a pilot project in community patent review in 2008 to test the efficacy of open review in the patent review process. The pilot elicited a positive response from reviewers and reviewer contributions exceeded expectations, thereby proving the potential of the peer-to-patent model, and so a new project, Peer to Patent Japan (at http://peertopatent.jp/), was launched in the beginning of 2011. The system is still under construction but is expected to go operational in 2012. The EPO has yet to follow suit by implementing a similar initiative, but the United Kingdom Intellectual Property Office allowed online review of patent applications from 1 June to 31 December 2011. 174 U.K. cases were selected for participation. Programmes like these provide an additional resource for patent

examiners to obtain pre-grant observations from third parties.

Ⅳ Inventive Step Case of Software

1 Hurdle of Inventive Step

How is inventive step as a condition for patentability to be applied in case of software? The JPO Guidelines state that combining technologies used in different fields or applying a technology used in one field to another is usually considered to belong to the realm of ordinary creative activity of a person skilled in the art. Consequently, when there is no technical difficulty for such combination or application, such an invention does not involve an inventive step save where special circumstances such as remarkably advantageous synergistic effects exist. Furthermore, software-related inventions may be considered obvious if they involve (i) an operation known in other fields, (ii) the addition of a commonly known means or the replacement of a means by an equivalent, (iii) the implementation in software of functions previously performed by hardware, (iv) routine-like systematisation of pre-existing human transactions, (v) reproduction of known events in a computerised virtual space, or (vi) a design modification on the basis of known facts or customs.

2 Problem-and-Solution Approach

In order to assess inventive step in an objective and predictable manner, instruct the EPO Guidelines, the so-called "problem-and-solution approach" should be applied. Under such an approach, there are three main stages: (i) determining the "closest prior art," (ii) establishing the "objective technical problem" to be solved and (iii) considering whether or not the claimed invention, starting from the closest prior art and the objective technical problem, would have been obvious to a person skilled in the art. In this approach, inventive step is in large part a product of postulating a problem and analysing the technical contribution thereto. All this utilises the closest prior art, which is why the significance of identified pre-existing material is fairly important. The involvement of an inventive step is judged by comparing the claimed invention with the closest prior art, so that if there is an absence or a misidentification of the latter, it is possible to get the inventive step considerably wrong. If precious

few relevant prior art has been found or noticed, an invention is given protection for an inventive step that is only apparent, not substantive, and the resulting patent can be said to be more or less frivolous.

Pursuant to Section 24bis of the Regulations under the Japanese Patent Act, the statement as provided for in an ordinance of the Ministry of Economy, Trade and Industry referred in Section 36(4)(i) of the Act, concerning the detailed explanation of the invention, must contain not only matters necessary for a person with ordinary skills in the art to understand the technical meaning of the patent, but also the problem to be solved through the invention and the means to solve that problem. The applicant needs to state the technical field to which the invention pertains and, in the section dealing with the solution to the problem, how the method or procedure has been embodied. This can be done using flowcharts and suchlike. In accordance with the JPO Guidelines, it is a violation of the requirement set forth in the Ministerial Ordinance if a person skilled in the art cannot understand the problem to be solved by the invention and its solution on the basis of the detailed description of the invention, drawings attached to the application or the common general knowledge at the time of filing.

Ⅴ Protection Conferred by Claims

1 Disclosure of the Invention

Proper construction of the claims is a basic step for any patent application. It requires identification of the nature of the subject matter of the patent and the scope of the protection that is sought. Claims must be supported by a sufficient disclosure of the invention so as to be acceptable. In Japan, Section 36(4)(i) of the Patent Act provides that the detailed explanation of the invention must enable any person ordinarily skilled in the art to which the invention pertains "to work the invention." Under Article 83 EPC the requirement is that an application be so "complete for it to be carried out by a person skilled in the art."

However, applicants need not disclose any source code to receive a patent. With inventions that utilise computer programs, the statutory requirements related to the disclosure of the invention have been interpreted in a way that it suffices to specify the functional description of the software. After all, a patent confers on the patentee an exclusive right to prevent

unauthorised third parties from using the patented product, the patented process or a product obtained directly by the patented process, in the meaning of certain claimed functionality, not just from reproducing the underlying programmed code.

The effect is that the invention in a software-related application can reside to a large extent in diagrammatic representations of the model. However, as regards computer programs, it is possible to describe the same underlying invention in a way that it appears to be an entirely different entity and, to some extent, vice versa. The sufficiency requirement, if applied vigorously, would appear to be one key to that dilemma. The disclosure of the invention must contain a description of how the claimed operational procedure can be realised on the computer – in other words, how the invention can be carried out –, and the claims should be construed within the frame of this explanation. The explanation must be sufficient for the hypothetical person skilled in the art to understand how the software and hardware are to be structured or else the application does not meet the statutory requirements.

2 Frivolous Patents?

The breadth and scope of protection conferred by a patent upon its right holder is determined by the claims. Should the language used therein be ambiguous, it results in difficulties of deciding just what is protected. This naturally applies to all technologies and is not specific to computer programs, but the adaptable nature of software can make the problem more difficult in this field of computer-implemented inventions. In accordance with Article 84 EPC, the claims must be clear and concise and be supported by the description. This is echoed by Section 36(6)(ii) of the Japanese Patent Act, pursuant to which "the invention for which a patent is sought is clear" is one of the items that that the statement of the scope of claims as provided in paragraph 2 of the same article must comply with. However, as has been explained, it is possible to delineate a computer program on multiple, equally true levels of abstraction.

Normally, once a product has been designed, if a review of existing patents reveals a blocking invention, it is possible to modify the product so that it does not infringe the patent. But the levels of abstraction as related to software may create difficult barriers to designing around granted patents, depending upon what has been allowed for protection. If the technical scope of a patent covers only one possible solution to a problem, it is most likely that there are alternative options available for competitors. In contrast, should protection have been given effectively to the problem itself, workarounds are of much less utility. In instances where a new technology area has received very broadly scoped protection at the outset, there is no need to build a patent cluster around the invention; a single patent is a blanket strategy in itself.

The legislative, administrative and judicial standards applied both in Europe and in Japan should limit the scope of the patent to a particular implementation in hardware and software. By that token, non-specific claims for the use of a general-purpose computer or a mobile computing platform to "translate documents," for example, do not meet the requirements of a patent monopoly even if the abstract idea was novel. Employing confusing claim language should not be of much assistance either, inasmuch as both the EPC and the Japanese Patent Act require that the patent application disclose the invention in a clear manner. Therefore, if the claims and the description of the invention are inconsistent, it is arguable that the application is in violation with the requirement of clarity.

VI Conclusion

1 Intra-Systemic Implications

(1) Problematic Nature of Classification

The classification of "software patents" is problematic. Even though protection may be less with that form of claim, a patent attorney equipped with relevant understanding of technology can turn a software invention into a hardware one. Industries have replaced analogue control with digital means and in many instances software has taken over the machine tasks. Owing to this ubiquity, it is nearly impossible to prevent patent applications that relate, more or less directly, to computer programs. And yet the EPC states that programs for computer are not patentable inventions as such. It might be better to accept the current position, which protects software in its own right in all but name, and come to grips with adapting the patent system to cope with the special demands of unorthodox subject matter.

(2) Invalidation Proceedings

Opposition, whether in the form of Part V EPC or Section 123 of the Japanese Patent Act, is a cheaper option than litigation. But the nine-month time limit set forth in Article 99(1) EPC is a double-edged sword. On the one hand it lays down a deadline after which patentees need not fear EPO-wide invalidation proceedings through a one-stop shop. However, it may also attract opponents, who feel bound to take action within the time limit, simply in the hope that some prior art arises. It may be that there ought also to be a centralised system of re-examination should prior art be found outside the opposition period. In any event, the existing European process could be stepped up, either by following the Japanese example of giving preference in the office to invalidation proceedings or perhaps by adopting the recent U.S. approach of a legislative fiat requiring disposition within a given period of time.[2]

(3) Affirmative Defence of Prior Use

Software artefacts are seldom static entities. It is normally necessary to fix problems that have manifested themselves in the course of the use of a program. Is the affirmative defence of prior use still available once security vulnerabilities or other bugs have been fixed? What about improving the usability or performance, or both, of the computer program; does it still maintain the general nature of such prior use? What of the replacement of software with a newer or better version, which offers the same functionality, but brings the system up to date and slightly improves its characteristics; is it approved as within the scope of prior use? For the time being, no unequivocal answer can be given.

(4) Concerns over Interoperability

It would be quite out of the ordinary if the technical scope of a patent equalled, one-to-one without remainders, its embodiment in the form of a computer program. Ordinarily one or more features of software are covered by a patent right, but the remaining components are unprotectable under this heading of intellectual property. However, because a patent confers on its owner the exclusive right to prevent third parties not having the owner's consent from the act of "using" and, where the subject matter of a is a product, "making" the patented invention, any execution of a program on a computer is, prima facie, an act restricted by patent if one or more such rights are involved. This may cause problems if one wishes to peruse the parts of the program that provide for interconnection and interaction between elements of software and hardware, in other words its "interfaces."

It is unclear whether patent law's current exceptions for experimental and research purposes permit reverse engineering so as to compile interoperability information if the idea is to bring a new computer program – either an add-on for or a competitor of the original software – commercially into the market. In an industry characterised by strong network externalities that privilege interoperable components and easily lead to vendor lock-in, where innovation is sequential and cumulative, this may create a problem. The function of a computer program is to communicate and work together with other components of a computer system and with users. Reverse engineering is the tool of last resort to make this interoperability possible and if patent law rules out such a possibility, it means that it will be possible to leverage patents to protect elements of software that are essentially unprotectable.

2 Extra-Systemic Implications: Interface with Copyright

Computer programs as such are protected as literary works within the meaning of Article 2 of the Berne Convention. Inventions relating to software can also be protected by patent. Both of these intellectual property rights can co-exist simultaneously in the same piece of software. The more patent and copyright law overlap, the more important it becomes that they take account of one another.

Copyright protects the program code and its particular implementation against reproduction, alteration, distribution and communication to the public, but not the implementation of a similar idea. Because of the availability of a patent following a substantial examination, copyright protection of software should be retained to instances of literal copying, whether in original or altered form, and temptations to broaden the scope of protection ought to be rebutted.

---

[1]   Emphasis added.

[2]   *See* Leahy-Smith America Invents Act of 2011, Pub. L. No. 112-29, § 316(a)(11), 125 Stat. 284, 302 (2011) (. . . the Director [of the USPTO] may, for good cause shown, extend the 1-year period by not more than 6 months . . .").