Software and Patent Law: Reverse Contaminations, Hybridizations and Trends, Observed Through the Legal Systems of Italy (and the EPC System), Japan, and the United States

Matteo Dragoni*

Abstract

This article compares how software inventions are protected by patent law in Italy (and the European Patent Convention (EPC) system), United States and Japan. Notwithstanding said legal systems appear to be so distant one from the other, surprising similarities may be found. The article describes such findings and attempts to provide an explanation of the same by looking at the peculiarities of the legal systems involved and how the patent system itself operates internationally.

I. Introduction

The legal systems of Japan and Italy have apparently very little in common. The current Italian legal system, built on 'Roman law' (or, to be more accurate, on a modified and reinterpreted version of the same) and influenced by other Roman-based legal systems, most notably the French and the German legal systems, has developed its own legal theories and doctrines since late XIX Century.

Japan, on the contrary, possessed a completely different legal infrastructure until the second half of the XIX Century, when, in an effort to update its legal framework, also came in contact with, and was partially shaped by, the French and the German legal systems of the times. Japan, however, as comparative law scholars know too well, never really 'copied' any of the mentioned legal systems, but adapted them to its own culture and legal traditions.

Superficially speaking, both Japan and Italy can be classified as 'civil law' systems, and their legal infrastructure has been somehow shaped by the French and the German models, although, unsurprisingly, with two radically different results. The two systems cannot certainly be called 'similar', even if similarities, once again unsurprisingly, may be found.

When it comes to intellectual property rights (IP rights), however, the common thought is that it is very difficult not to find any similarities between

* PhD in Comparative Intellectual Property Law, University of Macerata; Transatlantic Technology Law Forum (TTLF) Fellow, Stanford University; Adjunct Professor of Intellectual Property Law, University of Pavia.

legal systems, even the most distant ones. Given their immaterial character, IP rights can be more easily violated (technically, infringed) than many other kinds of 'property' rights, especially in the era of Internet, computers, smartphones, 3D printing devices and so on. This is why harmonization efforts began very soon in the IP rights' history, especially from the more developed economies of the late XIX Century, and never stopped until today. The aim was, and continues to be, the creation of IP rights which are the same, or at least very similar, worldwide and that can be easily, cost-efficiently and quickly obtained internationally.

As a result of more than one century of harmonization initiatives (and especially after the adherence to the TRIPs Agreement was set as a condition to be part of the World Trade Organization (WTO)), IP rights' basic principles and norms are indeed very similar in most of the world's economies. However, despite the great efforts which have been made to reduce differences between legal systems, when analyzing how IP rights are applied in practice in specific fields, (superficial) differences are immediately apparent, and warn the comparative scholar that the harmonization process might not be at such an advanced stage. At the same time, an in-depth analysis of the said differences from multiple viewpoints (and especially how the law is applied in court/administrative offices, interpreted by lawyers/law practitioners, officers and judges but also taught and studied by scholars) is necessary in order to ascertain whether we are dealing with real differences, or merely cosmetic ones.

In this paper, I take as an example the patent protection of software inventions, which is a very narrow and specific field to which the (allegedly) harmonized IP norms and principles are applied in practice. When dealing with software patents, the first impression is that, notwithstanding a harmonized starting point, we have very different results as to how general standards and requirements are applied. Especially when comparing Japan with Europe (Italy included) and with the United States.

II. A Legal Protection for Computer Programs

In general, patent protection for computer programs has always been a very debated topic.[1] For many years, starting from the sixties, international organizations and some of the most progressive countries of the times debated over which kind of protection software deserved.

While according to some proposals, computer programs deserved (only) copyright or patent protection, some others envisaged a *sui generis* right to

[1] See P. Kirby, 'Industrial Property Protection for Software' 5(2) *IIC – International Review of Industrial Property & Copyright Law*, 169 (1974) and J. Drexl, 'What Is Protected in a Computer Program?' *IIC – International Review of Industrial Property & Copyright Law Studies*, 15 (1994).

address the specificity of software creations: computer programs are abstract and somehow 'literary' in nature, but at the same time characterized by a strong technical side and a myriad of technical and industrial applications. Copyright protection means a protection which is very long-lasting (now in many countries it lasts for seventy years after the death of the author) but focused on the means of expression of the underlying idea and not necessarily requiring any kind of disclosure of how the software program works. In other words, longer and easier to obtain (it is granted automatically from the moment of the protected work's creation) but for some aspects easier to circumvent, although there might be the possibility to keep the important parts of software program (eg the source code) secret or partially secret. Patent protection, on the other hand, means a shorter protection (now maximum twenty years) but a stronger one: a specific product or process is protected, and also their 'equivalents', and in case of product patent the protection works independently of how the product is manufactured. This stronger exclusivity is counterbalanced by the need to file a patent application that at some point becomes public and in which the applicant has to disclose all (or most of) the details about the invention and no protection is granted unless a patent application is made in every State where the protection is sought.

With regard to the *sui generis* proposals, they were mostly a combination between copyright and patent protection, usually with a shorter duration than both (given the quick obsolescence of software programs).

In the US, a proposal to protect software through patents was submitted to the attention of the Congress in the early Sixties but the Presidential commission in charge of its evaluation concluded against such a legislation. Due to the intense lobbying against the patentability of computer programs, other similar proposals failed. On 12 December 1980 the US Parliament decided to pass the Computer Software Copyright Act, which included software amongst the copyrightable works: copyright protection was established even if the courts continued to debate over computer programs' patentability.[2] In Japan, in order to clear the uncertainties, the Ministry of International Trade and Industry (*breviter*, MITI) appointed a special Study Committee on Legal Protection of Software, which issued an interim report in 1972. According to the report, copyright protection was inadequate for software. A year later, the Second Subcommittee of the Copyright Council set up by the Japanese Agency for Cultural Affairs submitted a report stating the opposite, ie that copyright law was the most suitable way to protect software, also because it required minimal changes to the legislation. After several years of debate, in 1983 the MITI was presented with another report in which a *sui generis* legislation to protect computer software was recommended. However, even if this solution had obtained a large consensus, under pressure of the US and Europe, and upon suggestion

[2] See also J.M. Haynes, 'Computer Software: Intellectual Property Protection in the United States and Japan' 13 *John Marshall Journal Computer and Information Law*, 245 (1995).

of the Sixth Subcommittee of the Copyright Council set up by the Agency for Cultural Affairs, Japan decided to grant copyright law protection to software.[3]

With regard to Europe, Italy included, some of the first European decisions conferred protection only to the visual/graphic effects produced by the software, which was initially compared to a cinematographic work. Moreover, the approaches taken *vis-à-vis* computer programs were very different within the States of the European Communities, which tried to harmonize the entire system with the first Software Directive of 1991.[4] Pursuant to the Directive

> 'Member States shall protect computer programs, by copyright, as literary works within the meaning of the Berne Convention for the Protection of Literary and Artistic Works. For the purposes of th(e) Directive, the term "computer programs" shall include their preparatory design material'.[5]

With the clarification that protection shall apply to the expression in any form of a computer program and that 'ideas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected'.[6] The only condition which is required for the computer program to be protected is that 'it is original in the sense that it is the author's own intellectual creation'.[7]

In the end, copyright protection was largely adopted for software programs but their patent-eligibility was not categorically excluded. Some States, however, were indeed more prone than others in recognizing software patent-eligibility.

In Europe, Art 52 of the European Patent Convention, which excludes software *per se* from patent-eligibility, was read by several national courts – and by the European Patent Office itself – as an indication about 'what-not-to-do'. The United States, on the contrary, from the beginning of the Eighties became one of the strongest promoters and supporters of software patent-eligibility, especially if compared to the European Patent Office and Japanese Patent Office (JPO) very cautious approaches. A few years ago the situation in the United States radically changed, as an increasing number of scholars started to criticize such permissive approach,[8] and even the courts indirectly acknowledged a

[3] See, *ex multis*, R. Arancibia, *Intellectual Property Protection for Computer Software – A Comparative Analysis of the United States and Japanese Intellectual Property Regimes,* available at https://tinyurl.com/y6u5h6gr, 2003 (last visited 15 November 2018).

[4] See Council Directive 91/250/EC of 14 May 1991 on the legal protection of computer programs.

[5] See Directive 91/250/EC Art 1, para 1.

[6] See Directive 91/250/EC Art 1, para 2.

[7] See Directive 91/250/EC Art 1, para 3.

[8] See, *ex multis*, C.V. Chien, 'Reforming Software Patents' 50 *Houston Law Review*, 325 (2012); E. Goldman, 'Fixing Software Patents' *Santa Clara University Legal Studies Research*, 1-13 (2013). Some authors have even questioned the patent system as an incentive for innovation: see *ex multis* L. Larrimore Ouellette, 'Patentable Subject Matter and Nonpatent Innovation Incentives' 1115 *UC Irvine Law Review*, 5 (2015); C. Shapiro, 'Navigating the Patent Thicket:

few distortions in the then current system.

On the contrary, the attitude in Europe (meaning the European Patent Convention Countries, Italy included) and in Japan also changed in the last few years, with said legal systems heading towards a still prudent but less conservative approach towards software patents.

III. Software Patents in Japan

Differently from other legal systems, the Japanese Patent Act (JPA), instead of stating what cannot be considered as an invention (eg computer programs *per se* according to EPC Art 52), tries to positively define what an 'invention' is: an invention is 'a creation of a technical idea utilizing a law of nature'[9] As a consequence, the JPA does not contain a list of 'excluded subject matters', ie matters which cannot be the subject of a patent, but such a list is contained – to the benefit of the practitioner – in the JPO Guidelines and Handbook.

As opposed to other national or regional experiences in Japan there were no 'software specific' cases until less than a couple of decades ago. A few interesting cases merely dealt with patent-eligibility in general,[10] clarifying the basic legal requirements for an invention to be considered eligible for a patent and referred to before.[11]

Although (and probably because) there was no specific case law regarding software patents, the JPO took up the task to delineate which kind of protection software inventions deserved, if any.[12]

The JPO Examination Guidelines of 1976 were the first document to contain some guidance about computer program inventions.[13] Such guidelines were quite restrictive: even though they did not preclude patent-eligibility for computer software in general, they stated that programs themselves and recording media containing software were denied patent-eligibility.

Those Guidelines were followed by some improved versions in 1993 and

Cross Licenses, Patent Pools, and Standard Setting' 1 *Innovation Policy and the Economy*, 119-120 (2000).

[9] See Art 2 legge 13 April 1959 no 121.

[10] See Tokyo High Court Judgment 25 December 1956, Gyōshū, 7, no 12, 3157, so called *Utility Pole Advertising Method* case; Tokyo High Court, 22 December 1970, Hanta, no 260, 334, so called *Ionic Toothbrush* case; Tokyo High Court, 12 February 1986, Hanrei Kogyoshoyukenho 2001, at 16, so called *Electric Mirror Stand and Full Length Mirror* case; and Tokyo High Court, May 26, 1999, no 1997 (Gyo-Ke) 206, *Hanji*, no 1682, 118, so called *Video Recording Media* case.

[11] See A. Sako, 'Patentability of Inventions Incorporating Human Mental Acts (Intellectual Property High Court, August 26, 2008)' *Intellectual Property Law and Policy Journal*, 34 (2011).

[12] See H. Sakai, *Historical Transition of Computer Program Protection – A Review of Examination Guidelines over a Quarter Century* (Tokyo: Kobundo Publishing Co, 2015), 154-172.

[13] See Y. Aita, 'Legal Protection of Computer Software', in Id et al eds, *Advanced Science Technology and Intellectual Property Rights* (Tokyo: Hatsumei Kyokai, 2001), 117-119.

1997,[14] but the first significant modification to the JPO Guidelines can be traced back to the year 2000. According to the new Guidelines, an invention expressed as a sequence of computer processes can be considered as a 'statutory invention' regardless of the technology field and even if a computer program per se is claimed.[15]

In 2002, the Japanese Patent Act was amended and its Art 2 was changed to add 'computer programs' and 'any other set of information similar to a program that is designed to be used for computer processing' to the possible subjects of 'product patents'. The reason for this change was to provide a stronger protection for information technology products.[16]

Due to this amendment to the JPA, Japan became one of the few States in the world to explicitly recognize patent protection for computer programs, which can be even the subject of a so called 'product patent' (and not only of a 'process patent').

Starting from the early 2000, a few cases regarding software patents[17] were eventually decided. The JPO guidelines were integrated by the principles set out in the decisions of the Japanese Supreme Court and of the Tokyo High Court and, from its creation in 2005, of the Intellectual Property High Court, specialized in IP disputes.

The first important decision dealing with a software patent is the so called *LSI Simulator* case,[18] where the Tokyo High Court explicitly stated that even an algorithm could be patented[19] as long as the relationship between the algorithm and the physical parts of the invention is specified and there is a concrete interaction between the components.[20]

Another very interesting decision was rendered in the so called *Dental*

[14] See Intellectual Property Committee of the Industrial Structure Council, *Report*, 11 December 2001.

[15] See Intellectual Property Committee of the Industrial Structure Council, *Report*, 12 December 2001.

[16] See Task Force on Industrial Competitiveness and Intellectual Property, *Policy Report*, 5 June 2002, 5; Y. Aita, 'Current State and Remaining Issues of Patent Protection for Computer Programs' *Jurist*, 1303, 138-143 (2005).

[17] Or utility models, as in the decision of the Tokyo District Court Judgment, 20 January 2003, *Hanji*, no 1809, 3/*Hanta*, no 1114, 145 (the so called *Balance Sheet* case). See also N. Nakayama, 'Industrial Property Law' 1 *Patent Law*, 105-106.

[18] See Tokyo High Court 21 December 2004, (Gyo-Ke) 188, (2004) *Hanji* 1891,139.

[19] See R. Hirashima, 'A Note on Patenting Computer Software-related Invention & Assessing the Requirement "Utilizing a Law of Nature" Under the Japanese Patent Law – Something Like "the Suggestion" from LSI Simulator Case' 20 *Intellectual Property Law and Policy Journal*, 65-94 (2008).

[20] See N. Mizutani, *Technical Scope of a Software-Related Invention Described in Functional Claims*, *Commemoration of the Retirement of Prof. Toshiaki Iimura* (Toyko: Japan Institute for Promoting Invention and Innovation, 2015), 517-533. See also Id, 'Determination as to Whether the Laws of Nature Were Used to Make a Software-Related Invention' 20 *Intellectual Property Law and Policy Journal*, 77 (2008).

Treatment case,[21] decided in 2008 by the Tokyo IP High Court and concerning 'an interactive network for dental treatment'.[22] While the High Court conceded that mental activity itself cannot be considered as a statutory invention, it also stated that all technical means have some connections to mental activities of human beings 'as all technical means are created by human beings, assisting, facilitating or replacing human beings' activities including their mental activities'.[23] Therefore, unless

> 'the essential nature of the invention (…) is directed to a human being's mental activities (…) patentability of such invention should not be denied on the grounds of not being considered as an "invention" within the meaning defined in the Patent Act'.[24]

In *Knowledge Base System*,[25] a case decided a few years after *Dental Treatment*, the IP High Court clarifies what 'essential nature of the invention' means. According to the Court, when the 'hardware' components mentioned in a patent application are generic computers or recording media and their specific interaction with the underlying software or database is not clear, the 'technical significance' of the structure described in the patent application is meaningless. In such a case, the invention was considered as merely reciting abstract concepts linked with general computers, and therefore, even when analyzed as a whole, it does not contain any creation of technical ideas using the laws of nature, but mere abstract concepts, as such unpatentable. This also means that there must be a meaningful interaction between hardware and software components which cannot be reduced to the mention of some physical components in the patent in order to quickly pass the patent-eligibility threshold.[26]

After *Dental Treatment*, and before and after *Knowledge Base System*, a few other relevant decisions[27] followed, without however introducing any new or fundamental concepts that required any amendments in the JPO Guidelines.

Finally, it is worth mentioning that the Examination Guidelines also point

[21] See also Intellectual Property High Court 29 February 2008, *Hanji*, no 2012, 97, 2007 (Gyo-Ke) 10239.

[22] See Intellectual Property High Court 24 June 2008, 2007 (Gyo-Ke) 10369.

[23] ibid 25-26.

[24] ibid 26.

[25] See Tokyo IP High Court, 24 September 2014, (Gyo-Ke) 10014 (2014).

[26] See JPO Examination Handbook, Annex B, 14.

[27] See Intellectual Property High Court 26 August 2008, *Hanji*, no 2041, 124/Hanta no 1296, 263, (Gyo-Ke) 10001 (2008); Intellectual Property High Court of 16 June 2009, *Hanji*, no 2064 (so called *Amusement Machine* case) as explained in A. Sako, 'Patentability of Inventions Incorporating Human Mental Acts (Intellectual Property High Court, August 26, 2008)' n 11 above, fns 55 and 56; Tokyo IP High Court 5 December 2012, (Gyo-Ke) 10134 (so called *Energy Saving Action Sheet*) (2012); Tokyo IP High Court 24 February 2016, (Gyo-Ke) 10130 (*Energy Saving Action Sheet II*) (2015).

out, in line with the case law,[28] that in assessing patent-eligibility

> 'the invention should be viewed as a whole, [and that] it is inappropriate to identify the claimed invention separating the aspect of artificial arrangement and that of automation technique'.[29]

As a consequence, the most important part of software patents examination becomes the analysis of the inventive step of the invention,[30] ie whether the invention is obvious or not to the expert in the field of the invention.

In Japan, also at this stage, the invention is considered and assessed 'as a whole', ie without artificially severing the inventive parts from the non-inventive parts and, most importantly, without separating technical from non-technical elements.

The above implies that, for instance, just changing the data that are processed by software is likely to be considered as lacking inventive step, because the underlying program and the interaction between computer program and the machine are exactly the same.

However, if a new mathematical formula implemented by a computer process were to create, as a result, a new and non-obvious process, under the Japanese law this could be a patent-eligible invention and, if as a whole the process were to be found new and inventive, such invention could be considered patentable. Such invention would probably have a harder time in being awarded a patent in the EPO system, Italy included, or in the current US system.

IV. Software Patents in the European Patent Convention System

When dealing with software patenting in Europe,[31] we must start from Art 52 para 1 EPC, according to which patents

> 'shall be granted for any inventions which are susceptible of industrial application, which are new and which involve an inventive step'.

So, first of all, patent-eligibility requires an invention. However, the term 'invention', differently from the Japanese legal system, is not positively described within the EPC, which contains only a 'negative' definition. According to Art 52 para 2 EPC, discoveries, scientific theories, mathematical methods, aesthetic

[28] See *ex multis* Intellectual Property High Court 24 June 2008, (Gyo-Ke) 10369 (2007). See Intellectual Property High Court 26 September 2006, (Gyo-Ke), 10698 (2005).

[29] See also JPO Guidelines for Examination, chapter 2.2.

[30] See for instance one of the last decisions involving software patents, Intellectual Property High Court 6 August 2015, (Gyo-Ke) 10231 (2014), where the claimed invention was found to be lacking an inventive step.

[31] With specific reference to the States which are part to the European Patent Convention (EPC) and the case law of the European Patent Office (EPO) established under the said treaty.

creations, schemes, rules and methods for performing mental acts, playing games or doing business, presentations of information and programs for computers are not regarded as inventions.

The above-mentioned exclusion, however, is not as absolute as it might seem, since Art 52 para 3 EPC clarifies that it is limited only to patent applications directed to computer programs '*as such*'.[32]

Therefore, since a computer program *per se* seemed to be excluded from patentability, the analysis of the EPO Technical Boards and Boards of Appeal (BoAs)[33] was focused on distinguishing a patent-eligible software-related invention from a patent-ineligible one and then on trying to understand when a patent-eligible software invention is then patentable (ie new, inventive and industrially applicable).

The BoAs' case law initially followed two different general approaches with regard to patent-eligibility: the contribution approach and the whole-content approach. Eventually, only one of them was chosen as the most appropriate and an evolution of the same became the standard.[34]

According to the so called 'contribution approach', patent-eligibility is established after a *prima facie* examination of the (alleged) invention's inventiveness: the invention is inherently patentable when the invention's contribution to the art is technical and it is not limited to a subject-matter excluded from patentability (which in the case of the EPC includes software *per se*, algorithms, etc).[35]

According to said approach the invention is not examined 'as a whole' to determine the patent-eligibility, but the 'inventive part' of the (alleged) invention is somehow severed from the initial 'whole' and analyzed separately. If what

[32] See R.M. Hilty and C. Geiger, 'Patenting Software? A Judicial and Socio-Economic Analysis' *IIC – International Review of Intellectual Property and Competition Law*, 623 (2005); E. Arezzo, *Tutela brevettuale e autoriale dei programmi per elaboratore: profili e critica di una dicotomia normativa* (Milano: Giuffrè, 2012), 236.

[33] The EPO not only administers the examination of patent applications (which can be granted or rejected), but it is also equipped with a very functional system of appeal of the examiners' decisions. Both the decisions granting a patent or rejecting a patent application can be opposed in front of a Technical Board and the decision of the Technical Board can be appealed in front of a Board of Appeal.

[34] See P. Van den Berg, 'Patentability of Computer-software-related Inventions', in Members of the Enlarged Board of Appeal of the EPO ed, *The Law and Practice of the Enlarged Board of Appeal of the European Patent Office During Its First Ten Years* (Cologne: Carl Heymanns Verlag, 1996), 33.

[35] See generally J. Pila, 'Dispute over the Meaning of "Invention" in Article 52(2) EPC – The patentability of computer-implemented inventions in Europe' 36 *IIC – International Review of Industrial Property and Copyright Law*, 173-191 (2005). See G.D. Kolle, 'Patentability of Software-Related Inventions in Europe' *IIC – International Review of Industrial Property and Copyright Law*, 660 (1991) and J. Drexl, 'What Is Protected in a Computer Program? Copyright Protection in the United States and Europe' *IIC– International Review of Intellectual Property and Competition Law Studies*, 15 (1994); G. Guglielmetti, 'Brevettabilità delle invenzioni concernenti *software* nella giurisprudenza della Commissione di ricorso dell'Ufficio europeo dei brevetti' *Rivista di Diritto Industriale*, II, 358 (1994); Id, *L'invenzione di software. Brevetto e diritto d'autore* (Milano: Giuffrè, 1996).

seems to be the inventive part is made up only by excluded subject matter (or constituted an advancement in an excluded category), like computer programs, the invention is not patent-eligible.

Such analysis is carried out at the patent-eligibility stage, therefore before analyzing whether the invention is new, inventive and industrially applicable (and sufficiently described).

On the contrary, according to the 'whole content approach', in order to verify whether the invention is patent-eligible or not, the 'whole content' of the invention must be taken into consideration, without splitting it into inventive and non-inventive parts.

If the 'whole' invention is patent-eligible because it does not fall into one of the excluded categories, then the patentability analysis must be carried out. As a consequence of this approach, if the invention is a mix of technical and non-technical elements, the patent-eligibility test is rapidly passed, because the invention could hardly be considered 'a computer program *per se*', 'a mathematical formula as such', etc.

However, the issue of excluded subject matter is not solved with this first assessment, but it revives when considering the criteria of patentability (novelty, inventive step and industrial applicability), and in particular the inventive step. The invention is inventive, and thus patentable, if it is the solution of technical problems or technical means are used to achieve such a solution and the contribution to the art pertains to a field non-excluded from patentability, ie a technical one.

The EPO Technical Boards followed this second approach in several decisions,[36] alternating it to the above-described 'contribution approach'. At some point, the 'contribution approach' was slowly abandoned by the Boards,[37] even though sometimes a revival of the same could be discerned in some decisions.[38]

Starting from 2006, the BoAs began adopting a slightly different approach,[39] by recognizing that some technical means were enough to confer 'technical character' to software-related inventions, therefore making the patent-eligibility phase almost useless. This new line of decisions[40] has been referred to as 'any

[36] See for example T 0208/84 *Vicom/Computer-related Invention*; T 26/86 *Koch & Sterzel/X-Ray Apparatus*; T 115/85 *IBM/Computer-Related Invention*; *KEARNEY/Computer-Related Invention*, T 0042/87; T 236/91 *TEXAS INSTRUMENTS/Language Understanding System*; T 164/92 *Computer Components/Bosch*, etc. See also D. Schiuma, 'TRIPS and Exclusion of Software "as such" from Patentability' *IIC – International Review of Industrial Property and Copyright Law*, 1 (2000).

[37] See for example T 1173/97, *Computer Program Product/IBM*; T 1194/97, *PHILIPS/Record Carrier*; T 931/95, *PBS Partnership/Controlling Pension Benefits System*, etc.

[38] See P. Leith, *Software and Patents in Europe* (Cambridge: Cambridge University Press, 2007).

[39] See T 258/03, *Auction Method/HITACHI*; T 424/03, *Clipboards formats I/Microsoft*; T 154/04, *DUNS LICENSING ASSOCIATES/Estimating Sales Activities*.

[40] But especially T 424/03, *Clipboards formats I/Microsoft*.

hardware approach', since the mentioning of any hardware next to a computer software was enough to make the 'whole' invention patent-eligible and quickly go to the patentability analysis.

The EPO Enlarged Board of Appeal, which is a special composition of the BoA that addresses very specific and delicate matter concerning the application of the EPC,[41] clarified in 2010 that a computer program, in order to be patentable, has to generate 'further technical effects', but those effects do not have to be new or inventive (such analysis will be left to the subsequent inventive step and novelty assessments). In doing such statement, the Enlarged Board of Appeals confirmed that tying the software to 'any hardware' is not enough if in the end the software is claimed as the only invention in the patent. However, if something else containing the software is claimed, for instance a 'storage medium', there is no patent-eligibility issue there. Since 'storage media' *per se* are not excluded from patentability, the analysis will immediately be focused on the patentability requirements.[42]

The Enlarged Board of Appeal clarifies that during the patentability analysis all the elements of the invention have to be taken into account, not only the technical/non-excluded ones. In fact, also matters excluded from patentability may nonetheless contribute to the technical character of the invention.[43]

The EBA's teachings were followed in the subsequent decisions of the Technical Boards and are still followed today.[44] They have also been incorporated in the EPO Guidelines for Examination, whose last version became binding starting from 1 November 2017.[45]

V. Software Patents in the United States

United States' statutory law, unlike the EPC and Italy but similarly to Japan, never contained an explicit list of 'excluded subject matters', ie categories of inventions (or 'non-inventions') that are excluded from patentability. As a consequence, in the US there was no explicit exclusion of 'software patents' from the scope of the Patent Act. However, a less specific (if compared to the EPC) list of 'excluded categories' has been created through a series of judicial precedents.

[41] See G 03/08.

[42] See G 03/08, 38-39 at 10.8.7.

[43] See J. Pila, 'Software Patents, Separation of Powers, and Failed Syllogism: A Cornucopia from the Enlarged Board of Appeal of the European Patent Office' *Cambridge Law Journal*, 70 (2011). See also Id, *The Requirement for an Invention in Patent Law* (Oxford: Oxford University Press, 2010).

[44] See *ex multis* T 0313/10 *Item matching/AMAZON*; T 0573/12, *Automated process flow/SAP*; T 1755/10, *Software structure/Trilogy*; T 1385/12; T 1370/ 11, *On-demand property system/Microsoft*; T 1789/11, *Clipboard formats I/Microsoft*; T 1463/11 *Universal Merchant Platform/Cardinal Commerce*.

[45] See EPO, Guidelines for Examination, 2017.

US courts, through a series of decisions interpreting the US Patent Act and in particular USC 35 para 101, began to exclude 'abstract ideas, natural phenomena or natural laws' from the scope of patentable subject matter.[46]

In order to have some judicial precedents dealing specifically with software patents[47] we have to wait until the Seventies. The first one is *Gottschalk* v *Benson*,[48] which involved a method for converting binary-coded decimals into pure binary numerals. The US Supreme Court found that the invention was not related to any particular machine or apparatus, art or technology, but it was nothing more than a mathematical calculation that could be performed mentally by humans without the help of any machine, and therefore not patent-eligible.

A second relevant case is *Parker* v *Flook*,[49] where the Supreme Court stroke down another software invention because in the opinion of the Supreme Court the algorithm was the only new and useful characteristic of the invention, which per se was not patent-eligible and had to be considered as part of the prior art as a mere mathematical, abstract, discovery.

For a slight change of course from this more restrictive approach, we have to wait another couple of years with *Diamond* v *Chakrabarty*[50] and, with specific regard to software, the Supreme Court showed a more permissive approach towards its patent-eligibility in *Diamond* v *Diehr*.[51] In dealing with the patent at issue, the justices in *Diehr* found that, since the patent at hand involved physical and chemical processes for molding precision synthetic rubber products, it fell within the categories of patentable subject matter[52] and such conclusion could not be altered by the fact that in several steps of the process a mathematical equation and a programmed digital computer were used.[53]

The *Diehr* decision is interesting also from another point of view: While in Europe the contribution approach and the whole content approach were being conceived, the US Supreme Court immediately intervened to clarify a similar matter. The majority of the justices in *Diamond* v *Diehr* stated that

> 'in determining the eligibility of respondents' claimed process for patent protection under para 101, their claims must be considered as a whole. It is

[46] See *O'Reilly* v *Morse* 56 US (15 How) 62, 131 (1853).

[47] See M. Campbell-Kelly, 'Not All Bad: An Historical Perspective on Software Patents' 11 *Michigan Telecommunications and Technology Law Review*, 191 (2005), and E.R. Hyde, 'Legal Protection of Computer Software' 59 *Connecticut Bar Journal*, 298, 302-303 (1985).

[48] See *Gottschalk* v *Benson* 409 US 63 (1972), no 71-485. See also M.A. Duggan, 'Patents on Programs? The Supreme Court Says No' 13 *Jurimetrics Journal*, 135 (1973).

[49] See *Parker* v *Flook* 437 US 584 (1978).

[50] See *Diamond* v *Chakrabarty* 447 US 303 (1980).

[51] See *Diamond* v *Diehr* 450 US 175 (1981). See also, for different points of view, J.E. Cohen, M.A. Lemley, 'Patent Scope and Innovation in the Software Industry' 89 *California Law Review*, 1, 9 (2001) and K.E. Collins, 'Propertizing Thought' 60 *Southern Methodist University Law Review*, 317 (2007).

[52] *Diamond* v *Diehr*, n 51 above, 185.

[53] ibid 188-189.

inappropriate to dissect the claims into old and new elements and then to ignore the presence of the old elements in the analysis (…)'.

After *Diehr*, the Supreme Court did not hear any other patent-eligibility case until 2010. It was the Court of Appeal for the Federal Circuit (CAFC, or 'Federal Circuit') that adopted *Diehr*'s more permissive approach towards patent-eligibility and developed it even further. The landmark case that has been regarded as the first real opening to software patent-eligibility is *State Street Bank & Trust Co* v *Signature Financial Group, Inc.*[54] In this decision, the CAFC clarified that some types of subject matter, standing alone, represent no more than abstract ideas unless they are reduced to a practical application. Even an algorithm, in itself patent-ineligible, can become eligible if applied in a 'useful' way. In *State Street* the Federal Circuit held that the transformation of data, through a machine governed by an algorithm, into other data (amounts of dollar into final share prices) can be seen as the practical application of a mathematical formula, which produces a 'useful, concrete and tangible result'.[55] Such an invention is therefore patent-eligible.

Between 2010 and 2014, a trilogy of Supreme Court decisions changed completely the landscape of patent-eligibility in the United States.[56]

The first decision is *Bilski* v *Kappos* (2010), where the Supreme Court began demolishing the existing system by questioning the validity of both *State Street*'s 'useful, concrete and tangible result' test and the older 'machine-or-transformation' test,[57] however slightly mixing patent-eligibility criteria (natural laws, natural phenomena and abstract ideas) with some of the patentability concepts: the invention is not considered 'as a whole' when determining its patent-eligibility but what seems to be the novel/non-obvious part is severed

[54] See *State Street Bank & Trust Co* v *Signature Financial Group, Inc*, 149 F.3d 1368, 47 U.S.P.Q.2d 1596 (Fed Cir 1998).

[55] See J.R. Allison and S.D. Hunter, 'On the Feasibility of Improving Patent Quality One Technology at a Time: The Case of Business Methods' 21 *Berkeley Technology Law Journal*, 729, 730-31 (2006); J.E. Cohen and M.A. Lemley, 'Patent Scope and Innovation in the Software Industry' 89 *California Law Review*, 46 (2001).

[56] See J.F. Duffy, 'Rules and Standards on the Forefront of Patentability' 51 *William & Mary Law Review*, 609, 612 (2009). See also M.A. Lemley, 'Software Patents and the Return of Functional Claiming' *Wisconsin Law Review*, 905 (2013); A.-Bhattacharayya, 'Implementation, or the Possible Lack Thereof, of the Bilski Supreme Court Decision' 6 *Journal of Business and Technology Law*, 103 (2011); M.A. Lemley, 'Ignoring Patents' *Michigan State Law Review*, 19-21 (2008).

[57] See *Bilski* v *Kappos*, 561 U.S. 593. See also P.S. Menell, 'Forty Years of Wondering in the Wilderness and No Closer to the Promised Land: Bilski's Superficial Textualism and the Missed Opportunity to Return Patent Law to Its Technological Mooring' 63 *Stanford Law Review*, 1289 (2011); M.A. Lemley et al, 'Life After Bilski' 63 *Stanford Law Review*, 1315 (2011); P. Samuelson and J. Schultz, 'Clues for Determining Whether Business Methods and Service Innovations Are Unpatentable Abstract Ideas' 15 *Lewis & Clark Law Review*, 109 (2011); D. Crouch and R.P. Merges, 'Operating Efficiently Post-Bilski by Ordering Patent Doctrine Decision-Making' 25 *Berkeley Technology Law Journal*, 1673 (2010).

from the rest, and evaluated separately.

The second case about patent-eligibility is *Mayo Collaborative Services* v *Prometheus Laboratories*,[58] which was about a method to identify the correct dosage of a medicine to administer to patients that the Supreme Court found to be a mere attempt to secure a patent on a natural law. Interestingly, the Supreme Court also admitted that the para 102 novelty enquiry might sometimes overlap with the para 101 patent-eligibility analysis, but that this is not a good reason to eliminate the para 101 investigation entirely in favor of a 'patentability-oriented' approach.

The third relevant case about patent-eligibility, *Alice Corporation* v *CLS Bank International*,[59] involved four patents regarding an automated platform for mitigating settlement risk. The Supreme Court, attempting to clarify its past approaches, created a new two-step test:

> 'First, we determine whether the claims at issue are directed to one of those patent-ineligible concepts. If so, we then ask, "(w)hat else is there in the claims before us?" To answer that question, we 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. We have described step two of this analysis as a search for an "inventive concept" – ie, 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" '.[60]

Following this test, the Supreme Courts intended to reject any sort of 'any hardware approach', clarifying that the interaction of software with physical components is not enough, by itself, to make an invention patent-eligible.[61]

After *Alice* was decided by the Supreme Court, the United States witnessed a tremendous increase in software patents' invalidations, both at the district court level and at the appeal (Federal Circuit) level.[62] Such a trend started to be seen in a less 'pessimistic' way after the opinion of the CAFC in *DDR Holdings, LLC* v *Hotels.com*[63] and seemed to decrease starting from 2016, when three cases about software-related inventions were decided in favor of the patentee:

[58] See *Mayo Collaborative Services* v *Prometheus Laboratories, Inc*, 566 US 66 (2012). See also R.S. Eisenberg, 'Prometheus Rebound: Diagnostics, Nature, and Mathematical Algorithms' 341 *Yale Law Journal Online*, 122 (2013).

[59] See *Alice Corp. Pty Ltd.* v *CLS Bank Int'l*, 134 S.Ct 2347 (2014).

[60] 134 S.Ct. 2347, 7 (2014), page 7 of the decision.

[61] 134 S.Ct. 2347, 15-16 (2014), pages 15 and 16 in particular.

[62] See generally R.R. Sachs, 'Two Years After Alice: A Survey of the Impact of a "Minor Case" ' I (2016) and, for some statistics, available at https://tinyurl.com/nfbs9ya (last visited 15 November 2018).

[63] *DDR Holdings, LLC* v *Hotels.com, L.P.*, 773 F.3d 1245, 1255 (Fed Cir 2014).

Enfish, LLC v *Microsoft Corporation*,[64] *Bascom Global Internet Services*,[65] *Inc* v *AT&T Mobility LLC* and *McRO Inc* v *Bandai Namco Games America Inc.*[66] Such 'opening', however, was not the beginning of a new golden era for software patens, the majority of which, during the course of 2017, has been declared ineligible,[67] with only two cases[68] where patents concerning computer software managed to pass successfully *Alice*'s two-step test at the appellate level (CAFC), and namely: i) *Trading Technologies International Inc* v *CQG, Inc*[69] and *Visual Memory LLC* v *Nvidia Corp*.[70]

In other words, there seem to be no bright-line rule regarding patent-eligibility of software patents, but it is clear that the patent-eligibility analysis is extensively used by US Courts as a mean to judge (and often invalidate) patents.

Finally, once the patent-eligibility test is passed, the next steps are novelty and non-obviousness, which are not analyzed very differently from what the patent law and the United States Patent and Trademark Office (USPTO) guidelines instruct. The real hurdle for software patents is now the patent-eligibility phase.

VI. Software Patents in Italy

Having now briefly summarized how the EPC system, Japan and United States address software patent-eligibility and patentability (with only some references here and there to the Italian legal system), it is interesting to see whether and how Italy has dealt with such legal issues.

Since Italy is an EPC member State, it is not surprising that the Country has been strongly influenced by the EPO's case law and examination practices, notwithstanding the Italian judiciary remains independent and has to follow the EPC as far as it has been incorporated into Italian law.

[64] *Enfish, LLC* v *Microsoft Corp*, 2016 US App LEXIS 8699, 2016 WL 2756255 (Fed Cir 12 May 2016).

[65] *BASCOM Global Internet Services, Inc.* v *AT&T Mobility LLC*, no 15-1763 (Fed Cir 27 June 2016).

[66] *McRO, Inc.* v *Bandai Namco Games America, Inc.*, no 15-1080 (Fed Cir 13 September 2016).

[67] See for example *Intellectual Ventures I* LLC v *Erie Indemnity Co* CAFC Appeals nos 2016-1128 and 2016-1132 (7 March 2017); *Intellectual Ventures I LLC* v *Capital One Financial Corp*, CAFC Appeal no 2016-1077; *Return Mail, Inc.* v *United States Postal Service*, CAFC Appeal no 16-1502 (28 August 2017); *Secured Mail Solutions LLC* v *Universal Wilde, Inc*, CAFC Appeal no 2016-1728 (16 October 2017); *Smart Systems Innovations, LLC* v *Chicago Transit Authority*, CAFC Appeal no 2016-1233 (18 October 2017). See also, although non-precedential and amongst many non-precedential cases, *Clarilogic, Inc* v *FormFree Holdings Corp*, no 2016-1781 (5 March 2017).

[68] Or three, if we also consider the case *Thales Visionix Inc.* v *US*, 850 F.3d 1343, 121 U.S.P.Q.2d 1898 (Fed Cir 2017).

[69] *Trading Technologies International Inc* v *CQG, Inc*, CAFC Appeal no 2016-1616 (18 January 2017).

[70] *Visual Memory LLC* v *Nvidia Corp,* CAFC Appeal no 16-2254 (15 August 2017).

One would expect that, similarly to the UK, where software patent-eligibility has been the subject of debate multiple times in national courts (especially before the referral to the EBA of the EPO aimed at clarifying the matter and mentioned above), also in the Italian case law software's patent-eligibility and patentability have been highly debated. By analyzing said Italian case law, however, one soon discovers that after the EPO became operational, no relevant case regarding software patents can be found in Italy.[71] When software patents cases have been decided, no issues specifically relating to the peculiarities of software inventions' patentability and patent-eligibility have arisen or anyway no position is taken regarding them by the judges deciding the case.

It is not so easy to give an explanation to this phenomenon. One might be that since the Italian Patent Office did not carry out any kind of substantive patent examination until 2010, Italian patents had very little value, and patent litigation was mostly done in other jurisdictions (Germany, UK, etc) or directly at the EPO level through its opposition and appeal mechanisms. Another explanation might be that the Italian market is less strategical than others (independently of the strength of the national patents), so that big software litigations were started in other more crucial countries (once again Germany, UK, France, etc) but not in Italy.

Another possible explanation might be that in Italy, during court proceedings, much of the work, when it comes down to patents, is demanded to the court-appointed expert which interacts with the party-appointed experts. As a result, with specific concern to technical matters, it is not infrequent for the Italian judge (which has no technical background) to rely on the court-appointed expert's determination. Therefore, the judge (and the resulting decision) might not dive deep into the details of patent-eligibility and patentability of a software invention even when those issues arise because: 1) sometimes technical issues arise and are solved during the technical investigation phase; 2) connected to number 1), and given that anyway the lawyers know that there will be a technical 'phase' of the judgment where technical matters will be discussed, it is also common practice not to use and explain in detail all the arguments in the initial briefs and leave the matter (and some arguments) to the technical experts. As a consequence, it might not be so surprising to find out that decisions do not make reference to software's peculiar aspects.

One other possible explanation of the above paucity of Italian case law regarding software patents may also derive from the fact that Italian judges are very reluctant to decide differently from what the EPO has decided (especially after opposition proceedings and subsequent appeal have taken place). This

[71] In 1981, when the EPO was just becoming operational, the Italian Supreme Court decided a case regarding broadly-speaking, software patents, but without dealing with the specificities of software inventions highlighted before: Corte di Cassazione 14 May 1981 no 3169, available at www.dejure.it.

also means that once the matter has been settled at the EPO level, in case of European patent validated in Italy, the case can be considered almost decided also for Italy, except when the EPO decision was dependent on procedural aspects which might differ between the two systems (eg which kind of evidence is considered is not always the same at the EPO and in Italian courts).

Finally, another possible explanation might derive from EPO's 'any hardware approach', which has made crucial also for software inventions the inventive step examination phase, where the prior art as a whole comes into play to assess whether the invention was obvious or not. As a result, it might not be surprising that also Italian courts have dealt with the issue as a matter of inventiveness, and it is therefore more likely that software's specific problems and peculiarities remain hidden amongst all the other considerations regarding the prior art which is relevant to assess the presence of an inventive step.

On the practical level, and similarly to Japan, the absence of national guidance to national software patents cannot be considered, and was never, an obstacle to software patenting. It is true that in Italy there is some sort of reluctance, among many practitioners, to talk about software patents, but this has not prevented software patents from being issued in Italy. In some cases it might be true the opposite: patents that would not have survived an EPO examination could survive in Italy thanks to the absence of a substantive examination at the Italian Patent Office.

Even now that the EPO is entrusted with the substantive examination of Italian patents for the Italian Patent Office, the EPO carries out only one examination, and then sends back a report to the Italian Patent Office, stating whether according to the EPO the patent should be granted or rejected or how the patent should be modified in order to be granted. The EPO sticks to the report in its answer to the patent applicant. The applicant, however, can reply to the Italian Patent Office (through a patent attorney) and explain why the EPO was wrong in its report on the patent application and also make some amendments, preferably following at least some of the suggestions given by the EPO. If that happens, since the answer of the patent applicant to the EPO report does not go back to the EPO but stays with the Italian Patent Office and the Italian Patent Office is unequipped to respond and evaluate the matter, the patent is usually granted as proposed by the patent applicant anyway. Hence, the absence of case law specifically dealing with software patents does not have a negative impact of patent grants or on the patent examination in Italy. The EPO carries out the examination according to its own guidelines and case law, and absent in Italy a substantive examination phase, even if there were some national case law about software patenting, it would be anyway disregarded during the Italian quasi-examination phase.

VII. Conclusion

Despite the highlighted differences between the legal systems that are here considered, the result of the investigation is that there are also more similarities than one could imagine. Whether such, sometimes surprising, similarities are the product of a fruitful dialogue between patent offices and judges of different jurisdictions or merely the result of casuistry, is hard to tell. It is certainly true that patent offices around the world, and especially JPO, EPO and USPTO (along with the Korean and Chinese patent offices), are always following very carefully what happens in other crucial jurisdictions, so that some contaminations and influences can be expected. Patent grants, and how strict a patent office's examination is, are also matters of policy, and Countries can be more or less interested in having weaker, stronger, broader or narrower patents than other jurisdictions, and they constantly monitor what other patent offices and other national courts do in order to compete for the most attractive legal system (for investments, etc).

Japan, similarly to Italy, had almost no case law regarding software patents and their patent-eligibility and patentability criteria. Nonetheless both countries seem to be venues where software inventions can obtain more easily a patent, even if for different reasons. A very liberal, pro-software, patenting approach of the Japanese legal system, opposed to the absence of a real substantive examination in Italy.

The 'essence of the invention' analysis that is performed by the JPO and by Japanese courts, on the other hand, is very similar to what happens in the EPC system (Italy included, for the reasons stated above) during the inventive step examination phase. The technological problem is found and the means to solve such a problem are isolated: if such means do not make use of any laws of nature, then the invention is not patentable. This is not very different from what happens in the EPC inventive step analysis: the problem-solution approach (which is the main, although not unique, method for assessing the inventive step) is similarly applied, with the difference that in Japan novelty and non-obviousness considerations are usually not made in conjunction with this step, although patent practitioners confirm that there is no clear guidance and they might sometimes be performed.

With regard to a Europe-US comparison, when the US courts mention the 'inventive concept', they are reviving some sort of 'contribution approach' (long abandoned in the US since the eighties), that tries not only to find what the 'core' invention is (the solution to the problem, Japanese style) but also what is really 'inventive' in the invention. In doing so, inventiveness is necessarily taken into account, mixing patent-eligibility with patentability. The *Mayo-Alice* approach looks at the 'essence' of the invention, but in a slightly deeper way than the Japanese system does.

Once again, however, the US practice seems very distant from the European

one but it is not. Applying *Mayo-Alice* test is not so different from what the EPO and its boards do when assessing the patentability of the invention, but without artificially severing the inventive part of the invention before assessing the patent-eligibility of the same. If some 'further technical effects' are produced or, alternatively, some kind of hardware is mentioned, the invention is patent-eligible, but if its 'inventive'/'non-obvious' part resides in a non-patentable element, then the invention cannot be considered to be 'non-obvious', because excluded subject matter as per Art 52 para 2 EPC would dictate so. And the inventive part is evaluated by taking into account only those elements that contribute to the technical character of the invention.

In other words, in Europe this separation between 'inventive' and 'non-inventive' occurs at a later stage, but the general criterion used to deal with 'excluded subject matter' does not seem so different.

In the end, the examination of software inventions is a composite evaluation of what is 'inventive' and what patent policy wants to be 'excluded subject matter'. Such evaluations take place at different stages: patent-eligibility for US and Japan; inventive step for the EPC system, including Italy.

Despite the apparently radically different approaches, by knowing differences and similarities of the systems that have been analyzed, a good patent practitioner is certainly able to draft a patent application in such a way as to maximize the chances of survival of the same patent application in multiple jurisdictions, and at least in Japan, Europe and the United States.