EUROPEAN PARLIAMENT

Directorate-General for Research

WORKING PAPER

The patentability of computer programmes

Discussion of European-level legislation in the field of patents for software

Legal Affairs Series

JURI 1xx EN

The document is available in

EN (original) only.

An abridged edition, JURI 1xx AXX, will be published in EN (OR), ...

Publisher: **EUROPEAN PARLIAMENT**
L-2929 Luxembourg

Authors:
Reinier BAKELS, project researcher at the Institute for Information Law of the University of Amsterdam (IViR), under the supervision of **P. Bernt HUGENHOLTZ**, Professor of Intellectual Property Law at the University of Amsterdam and Co-Director of IViR

This study was commissioned at the request of the Committee on Legal Affairs and the Internal Market of the European Parliament.

Editor: **Klaus H. OFFERMANN** Directorate-General for Research
Division for Social and Legal Affairs
Tel.: (00352) 4300-23709
Fax.: (00353) 4300-27723
E-mail: koffermann@europarl.eu.int

Manuscript completed in April 2002.

EUROPEAN PARLIAMENT

Directorate-General for Research

WORKING PAPER

The patentability of computer programmes

Discussion of European-level legislation in the field of patents for software

Legal Affairs Series

JURI 1xx EN

04-2002

Table of contents

1. Preface 3

2. Introduction 3

3. Legal framework 5
3.1. Legal protection of computer programs 5
3.2. The European Patent Convention 6
3.2.1. Overview 6
3.2.2. Procedural aspects 7
3.2.3. Case law 7
3.2.4. Examination Guidelines 9
3.3. The United States 9
3.3.1. Statutory regulations 9
3.3.2. Case law 10
3.4. Japan 11
3.4.1. Statutory law 11
3.4.2. Examination Guidelines 12
3.5. International framework: the TRIPs Agreement 12

4. Observations and projections 14
4.1. Methodology 14
4.2. Software and the software industry 16
4.2.1. "Computer program related inventions" 16
4.2.2. The computer software industry 17
4.3. The functioning of the patent system in the economy 18
4.3.1. Patents and innovation 18
4.3.2. Patents, monopolies and "network effects" 19
4.3.3. Strategic use of patents 20
4.3.4. Patenting "Business Methods" 21
4.4. Small and Medium-sized Enterprises 23
4.5. Open Source Software developers 23
4.6. Patent offices 25
4.7. Conclusion 26

5. The proposed European software patent directive 26
5.1. Summary of the proposed directive 27
5.2. Comments 27

6. Further issues 29
6.1. The requirement of "technical character" 29
6.2. Triviality and quality 31
6.3. A European Patent Observatory 32
6.4. Other solutions 33
6.4.1. Protection of software inventions by special legal means 33
6.4.2. Shorter protection term for patents 34
6.4.3. Compulsory licences 34
6.4.4. Measures to contain litigation costs 35
6.5. Towards a community patent 36

7. Summary and conclusions 36

8. References 38
8.1. Previous studies 38
8.2. Other references 39

1. Preface

This report is the result of a short-term study commissioned by the European Parliament on the desirability of EC level legislation in the area of software patents. It is based on a comparative analysis of the present state of the law, and the advantages and disadvantages appearing from current practice in the EC Member States, the United States and Japan. While its principal focus is on software patents, the report also includes commentary on the patenting of "business methods", as patents in this area are closely related to software patents.

While not explicitly stated as an objective, the study has also considered the proposal for a "Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions", that was published on February 20, 2002, at a time when this report was being finalised. Because of this timing, the reactions that the proposal has elicited could be taken into account only to a very limited extent.

This is certainly not the first study to examine the pro's and con's of software patenting. Several government sponsored studies have been undertaken before, most of them fairly recently, commissioned by either the European Commission or the Member States. Typically, these studies have included extensive consultation exercises. In view of that fact, and the very limited time frame allowed for producing the present study (three months), we have refrained from conducting yet another consultation. Instead, we have made ample use of the results of previous studies.

Since the purpose of this report is to inform the European Parliament, it aims at discussing both sides of the equation whenever possible, rather than advocating some particular thesis. It is intended to serve primarily as a position paper for policy makers, discussing both legal and factual aspects. To serve its purpose we have kept the volume of this study within limits, and have refrained from extensive quotations and footnoting.

This study was written by Reinier Bakels, project researcher at the Institute for Information Law of the University of Amsterdam (IViR), under the supervision of P. Bernt Hugenholtz, Professor of Intellectual Property Law at the University of Amsterdam and Co-Director of IViR.[1] All views and opinions expressed in this study are the sole responsibility of IViR, and can in no way be attributed to the European Parliament.

Amsterdam, 14 April 2002

2. Introduction

A high patent activity is generally considered a sign of economic health. Governments tend to promote patents: patent application must be made easy, and public patent awareness must grow. Indeed, patents can provide economic incentives to invest in inventive activities. And patents can be useful because patents are only granted if the patented inventions are disclosed. Disclosed inventions may serve as a source of inspiration and information for subsequent inventors working on new inventions.

However, patents can also be harmful. Patents create exclusive rights that may cause economic monopolies with negative effects for potential competitors, and for society as a whole. Ideally, a

[1] <http://www.ivir.nl>.

patent system should maximise the net positive effects. It is important to distinguish this objective from a moral view of property rights. While patents are a form of intellectual "property", it is generally accepted that the mere efforts of an inventor do not by themselves justify a "natural" right in their inventions. Patent law essentially is economic law, and its merits must be judged primarily in economical terms.

As new technologies and new types of inventions emerge, the question must be asked again whether patent protection is an appropriate and efficient system. The present study attempts to answer this question in respect of computer program related inventions. Is there a fundamental difference between traditional nuts-and-bolts technology and software technology? Patents are granted for inventions in a wide variety of industries. Are software related inventions really different?

Should "Europe" act proactively, and follow the American example? The United States' tremendous success in the software business, and the high patenting activity in this area in the U.S. appear to suggest a positive correlation between patentability and industry success. In reality, however, patenting activity might be caused by innovative activity, rather than the reverse.

On the other hand, there are serious concerns that the negative effects of these new types of patents may be stronger than those of "traditional" patents. Some even claim that software patents may have no less than a disastrous paralysing effect on the software industry. Small and Medium Sized Enterprises, who are sometimes supposed to benefit most from patents, may actually suffer worst from software patents. Software is increasingly provided on a non-commercial basis as "Open Source Software". Generally, Open Source Software is considered very valuable for most players in the IT industry. But there are fears that Open Source Software developments may be frustrated by unjust patent claims.

The debate over whether the benefits from patents really offset the disadvantages is not new. Patents on software related inventions have been granted in one form or another for some decades now, with certain restrictions. The present debate focuses on the question whether these restrictions are still valid.

Several studies have been undertaken in recent times in an attempt to answer this question.[1] There is a considerable degree of consensus that the rules that currently exist in Europe provide insufficient legal security, and need to be amended for clarification. On the other hand, it is highly controversial whether there is any reason for a substantial change of the law.

None of these studies have really arrived at concrete answers to this question. There are no factual data providing solid proof that software patents provide any benefits to society; neither is there firm evidence of the opposite. Extensive consultation exercises conducted as part of most of those studies have revealed widely varying opinions from experts and interested parties on what *might* happen and on what one would *like* to happen. The constant factor in all these consultations appears to be the diversity of the software industry, the diversity of the subject matter "software" itself, and the diversity of the opinions on patentability in this environment.

These observations constitute an important starting point for the present study. We will in particular investigate *why* the various studies in the recent past did not arrive at consistent conclusions, in order to come to a synthesis that will help the European Parliament to arrive at an informed opinion.

In the next chapter of this study (Chapter 3) we will sketch the existing legal framework, both at the European and the international level. In Chapter 4 we will critically assess the patent system; what are the actual advantages and disadvantages of patent systems in general, and applied to computer software and business methods in particular? In Chapter 5 we will comment upon the

[1] These studies will be further introduced in the "Observations and Projections" chapter. For reference information see Annex 8.2.

proposal of a European Software Directive in the light of the legal and economic framework presented in the previous chapters. In Chapter 6 we will discuss possible improvements of the patent system. Finally, in Chapter 7 we will summarise our conclusions and recommendations.

3. Legal framework

In this chapter, we will give a brief overview of existing legal regimes for the protection of computer programs. Next we will describe in some detail the way computer programs are currently treated under patent law in the major industrial regions in the world: Europe, the United States and Japan. Finally, we will elaborate on the existing international framework provided by the TRIPs agreement.

3.1. Legal protection of computer programs

Patent law is certainly not the only legal regime available for protecting computer programs. Ever since the adoption of the European Directive on the legal protection of computer programs in 1991[1] and its subsequent transposition into national laws, computer programs have been protected under copyright as 'literary works'. In the United States copyright protection for computer programs has been codified as early as 1980.

Copyright and patent protection of computer software are complementary regimes. Copyright protects 'original expression' – in the case of computer software: the originally coded program – against direct copying. Patents protect inventive ideas – in the case of a software related invention: the exclusive right to apply the idea. Patents can be invoked even against independent inventors of the same idea; in this sense they create monopolies that may be hard to circumvent. Copyright, on the other hand, cannot prevent independent creators from recreating the same or similar work. Usually, it is possible to implement a program's functionality in different ways, i.e. in a different "expression", so the exclusive right provided by copyright law is unlikely to create a monopoly.

Given the cost and burden of the patent application process, software inventors often refrain from patenting.[2] In particular for those inventions that can be hidden in software internals, copyright protection may actually provide sufficient legal protection for underlying inventions as well. If the program "source code" is kept secret, such inventions may be hard to uncover from the "object code".[3] Arguably, copyright protection of the "object code" layer may constitute an impediment to establishing proof of patent infringement in some cases.[4] Statutory exemptions allowing "decompilation" (i.e. reverse engineering), as provided for pursuant to Art. 6 of the European Software Directive, apply only if a program is reproduced with the purpose of achieving interoperability.

Trade secrets enjoy legal protection as well by various regimes such as tort law. If secrecy is technically an option, it has many advantages. But the weakness of a trade secret is that it is lost once it is broken. The risk of a trade secret being lost is particularly high if it is licensed.

Patent law and copyright are complementary legal regimes. Depending on the type of invention, software copyright can effectively protect software inventions as well.

---

[1] *Council Directive 91/250/EEC of 14 May 1991 on the legal protection of computer programs*, <http://europa.eu.int/eur-lex/en/lif/dat/1991/en_391L0250.html>.

[2] As we will elaborate later, SMEs in particular tend to rely on copyright protection for software.

[3] Programs are written in "source code" and subsequently compiled into "object code" that can be executed on a cumputer. Source code is readable for programmers, object code is readable for computers only.

[4] Xerox Corporation has created two laboratories with the task of analyzing competitors' products for infringements; see 'Xerox sues rivals over patents', *Patent World* (1998) 104 (Aug.), p. 12.

3.2. The European Patent Convention

3.2.1. Overview

European patents are governed by the European Patent Convention,[1] a treaty between all member states of the European Union and several other European countries. Co-existing with the European patent system, national patent laws in Europe offer additional opportunities for patent application at the national level. Unlike other areas of intellectual property such as copyright and trademark law, patent law has remained largely unaffected by harmonisation at the EU level. The only existing directive in this area is the Biotechnology Directive adopted in 1998.[2]

National patent laws and the European Patent Convention are similar in structure. Patents are granted for *inventions* on certain *subject matter*, if certain – substantive and formal – *requirements* are met.

The law usually does not define explicitly what constitutes an invention. According to Art. 52 (1) of the European Patent Convention, "European patents shall be granted for any inventions which are *susceptible of industrial application*, which are *new* and which are *not obvious*." Subsequent subsections of Art. 52 EPC list subject matter that cannot be the object of an invention. Some subject matter is inherently unpatentable, such as discoveries and scientific theories (Art. 52(2)(a) EPC). There is also subject matter that is excluded for reasons of social policy, such as medical methods in Art. 52(4) EPC.

Computer programs are explicitly mentioned on the exclusion list of Subsection (2). Subsection (3) however specifies that subject matter listed in Subsection (2) is only excluded from patentability "as such". Chemical theories for instance cannot be patented "as such", but a chemical theory leading to a new medicine can indeed be patented in the context of a pharmaceutical patent claim. Similarly, a computer program can be patented if it is part of diagnostic equipment patent claims.[3]

From Art. 52 it follows, that a statutory invention must meet other requirements in order to qualify for a patent. Firstly, the invention must involve what is called an *inventive step* - which makes it *non-obvious*. The requirements for an invention to be novel are further set out in Article 54. The requirement of *inventive step* is elaborated in Article 56, which states that an invention must not be "obvious to a person skilled in the art". "Industrial application" is clarified in Article 57, which requires that an invention must be "*made or used in any kind of industry*".

In addition, it is generally assumed that an invention must also be *technical* in order to qualify for a patent. This requirement is not mentioned explicitly in the EPC, but rather derived from European Patent Convention Rule 27,[4] which explains that *the description of the invention shall specify the technical field to which the invention relates.* There is no generally agreed legal definition of the word "technical" within this context. In particular, there is considerable debate to what extent computer software is to be considered "technical" for the purpose of the patent law.

There is some confusion over whether the requirement that an invention be "susceptible for industrial application" (Art. 57 EPC) actually implies technical character. This confusion may be the result of differences in meaning of the term "industry" in various European languages. In French and Dutch the word "industrie" is used only for the manufacturing industry, which has a more narrowly defined technical character. But the English word "industry" really refers to any

---

[1] *European Patent Convention*, <http://www3.european-patent-office.org/dwld/epc/epc_2000.pdf>.
[2] *Directive concerning the harmonisation of the patentability of biotechnological inventions*, 98/44/EC.
[3] T26/86, *Official Journal of the EPO* 1988,19 (*Koch & Sterzel*). More case law is presented in the next section.
[4] According to Article 164(1) EPC, the rules are to be seen as part of the Convention.

kind of industry even including the "government industry". Similarly, the German requirement that an invention be "gewerblich anwendbar" (commercially applicable) has a broader scope than the (technical) manufacturing industry, but not quite as wide as the English word "industry".
Either way, it is generally assumed that the decisive factor for patentability is whether a computer program has a "technical character". The criteria developed by Technical Board of Appeal of the European Patent Office to assess whether a computer program has such a technical character will be discussed below in the case law section of § 3.2.3. But first we will discuss a few procedural aspects of patent law in order to identify the possible sources of case law.

3.2.2. Procedural aspects

European Patents are granted by the European Patent Office (EPO). Once a European patent has been granted, it confers to its proprietor in each of the contracting states of the EPC the same rights as would be conferred by a national patent.[1] As a consequence, the adjudication of patent disputes is divided over European and national fora as follows.
Appeal against EPO decisions, in particular in the grant phase, is possible before the EPO "Board of Appeal" or the "Enlarged Board of Appeal".[2] While these appeal bodies are formally part of the EPO system, they are in a position to decide independently from the original EPO decision.[3] At this stage, no further appeal is possible.
The EPC leaves it to the national courts of the contracting states to deal with infringement of European patents.[4] At that stage, appeal is possible only within the country itself, up to the highest-level national court. National courts may interpret the EPC differently. For instance, a European patent may be revoked in one country, while it is still considered valid in another. This may be problematic in those areas where there is debate over whether a certain category of subject matter is patentable, e.g. computer software. In borderline cases such as these national courts will not decide consistently.

3.2.3. Case law[5]

From the above it follows that there are two sources of case law for European patents: decisions of the courts of the contracting states and decisions of the Boards of Appeal of the European Patent Office.[6] In this paragraph will focus on the latter category of case law, as it applies to all contracting states.
The distinction between (unpatentable) inventions involving computer programs "as such" and (patentable) computer program related inventions is at the heart of a series of decisions of the Technical Boards of Appeal.[7] The criterion of technical character has consistently played a crucial role in these decisions, in one form or another.
In the *VICOM* case[8] image processing was considered to lead to a result sufficiently technical to qualify for patentability, even though it is based on a mathematical method. In the *Koch &*

---

[1] Art. 64(1) EPC.
[2] Art. 106–112 EPC.
[3] See Art. 23 EPC: "Independence of the members of the Boards".
[4] Art. 64 (3) EPC.
[5] For a comprehensive overview of the current possibilities to obtain patents for software under the European Patent Convention see Beresford 2000. This monograph is written primarily from the perspective of the practising patent attorney and its clients: it promotes software and business method patents without reservations. Not much attention is paid to public policy matters.
[6] *Case Law of the Boards of Appeal*, <http://www.european-patent-office.org/legal/case_law/e/index.htm>.
[7] <http://www.european-patent-office.org/legal/case_law/e/I_A_1-1.htm>.
[8] T208/84, *Official Journal of the EPO* 1987,14.

Sterzel decision[1] likewise a computer program was considered to be used for a technical purpose, in this case controlling X-ray equipment. Software that was used for the co-ordination and exchange of data between interconnected processors was also considered to have a sufficiently technical character.[2] Even a business problem solution requiring "technical considerations" was held to be technical in the *SOHEI* case.[3] In particular, it was emphasised that an otherwise statutory invention would not become non-statutory because of the fact that a business method is involved. Software to be used in conjunction with the display of special (e.g. Arabic) characters was, however, not considered sufficiently technical.[4] As a last example, we mention a decision holding that a method for interactive rotation of displayed graphical objects was, once again, sufficiently technical.[5] This case demonstrates an important point: the way a patent claim is worded is often decisive. Initially the application was rejected, but after redefinition of the claims it was accepted.[6]

In several of these cases, the technical character was derived in a straightforward manner from the involvement of traditional "hardware" technology. Other decisions are less convincing. Is there really a relevant difference between the display of graphical objects and the display of special characters? Isn't systems control software really a computer program "as such" and "technical" software at the same time? The delimitation between patentable and non-patentable inventions as shown by these decisions seems rather arbitrary.

Two recent decisions of the EPO Technical Board of Appeal, both involving IBM patent applications,[7] mark a change in policy of the EPO. In these decisions the Board expressed the following opinion:

> *"In the view of the Board, a computer program claimed by itself is not excluded from patentability if the program, when running on a computer or loaded into a computer, brings about, or is capable of bringing about, a technical effect which goes beyond the "normal" physical interactions between the program (software) and the computer (hardware) on which it is run."*
>
> "*Furthermore, the Board is of the opinion that with regard to the exclusions under Article 52(2) and (3) EPC, it does not make any difference whether a computer program is claimed by itself or as a record on a carrier.*"

The Board explains that every computer program "as such" causes a "technical effect" when run on a computer in the form of electrical currents in the electronic circuits of the computer's processor. As the EPC excludes computer programs "as such" from patentability, this technical effect by itself is apparently insufficient for patentability. Consequently, some other, "further" technical effect is required. Such an effect would obviously be present if the computer program controls, for instance, a conventional ("technical") piece of machinery, such as x-ray equipment in the case mentioned above.[8] In the present cases, the *further technical effect* was found in the control of computer system resources, distinguishing systems control software from application programs.

The Board's decisions in the IBM cases indicate that patents may be obtained for computer program *products*. The Board considers that "it would seem illogical to grant protection for a technical process controlled by a suitably programmed computer but not for the computer itself when set up to execute the control". The practical importance of the possibility of such

---

[1] T26/86, *Official Journal of the EPO* 1988,19.
[2] T6/83, *Official Journal of the EPO* 1990,5.
[3] T769/92, *Official Journal of the EPO* 1995, 525.
[4] T158/88, *Official Journal of the EPO* 1991, 566.
[5] T59/93.
[6] Beresford 2000, Chapter 3.
[7] T935/97 and T1173/97, *Official Journal of the EPO* 1999, 609.
[8] *Koch & Sterzel*, T26/86, *Official Journal of the EPO* 1988,19.

computer program product patents is that the unauthorised sale of such a program amounts to direct patent infringement. Otherwise, such a sale would at best constitute "indirect"(or contributory) infringement, which usually requires knowledge or negligence on the part of the (indirect) infringer, and therefore might be more difficult to prove.
To be sure, the notion of a "computer program by itself" is not to be confused with the notion of a "computer program as such". The former notion relates to the packaging and distribution of software, while the latter relates to its functionality. Understandably, the Technical Boards of Appeal only considers the functionality a relevant criterion for patentability.
Finally, a few words on business method patents. Contrary to popular belief, under certain conditions business methods can be patented in Europe already today, as becomes apparent from the case law of the Technical Boards of Appeal.[1] The involvement of a machine by itself is not sufficient to give a business method invention a technical character.[2] But a (business) method to unload ships by technical means such as bagging machines was deemed technical indeed.[3] And the *SOHEI* decision quoted above shows that the technical contribution, if only present, could be only minor. A system for waiting line (queue) management was also considered technical, because it involved a special device.[4]
The recent *Pension Benefits System* decision[5] involved a business method executed by a computer system. The Board found that "the improvement envisaged by the invention is essentially an economic one i.e. lies in the field of economy, which, therefore, cannot contribute to the inventive step", and ruled that a patent could not be granted. In this case the technical *contribution* rather than the technical *character* of the invention was considered decisive.

3.2.4. Examination Guidelines

The European patent Office has published detailed "Examination Guidelines",[6] that give directions on how to interpret the rules of the EPC on such matters as the assessment of patentability, novelty, and inventive step. It must be emphasised, however, that the Guidelines do not constitute "law" – they can be, and have been overturned by the Boards of Appeal.

Computer program related inventions are patentable in Europe if they produce a technical effect beyond the electrical currents in the computer circuitry.

A computer program product can be patented, either on a carrier such as a CD-ROM, or by itself (e.g. if the program is distributed over the Internet).

Business methods are patentable in Europe if the invention provides a technical contribution.

3.3. The United States

3.3.1. Statutory regulations

Under U.S. patent law "*Whoever invents or discovers any new and useful process, machine, manufacture of composition of matter [...] may obtain a patent therefor [...]*" (35 U.S.C. § 101). Section 102 sets out the *novelty* requirement, whereas Section 103 clarifies that patents are

[1] <http://www.european-patent-office.org/legal/case_law/e/I_A_1-4.htm>
[2] T854/90, *Official Journal of the EPO* 1993, 669.
[3] T636/88.
[4] T1002/92, *Official Journal of the EPO* 1995, 605.
[5] T931/95.
[6] *Guidelines for Examination in the European Patent Office* <http://www.european-patent-office.org/legal/gui_lines/pdf/gui_e_full.pdf>. The current version comprises 466 pages.

granted only for *non-obvious* subject matter. Unlike the EPC, the U.S. Patent Act does not comprise a list of subject matter that is excluded from patentability. Also, under U.S. law there is no statutory requirement of "technical character".

3.3.2. Case law

The U.S. Supreme Court has held that Congress intended to allow patents for "anything under the sun made by man".[1] The courts have clarified that "laws of nature, natural phenomena and abstract ideas" are not patentable.[2] Such subject matter should be "free to all men and reserved exclusively to none". Business methods originally were considered not patentable, as they would represent "abstract ideas".

The first U.S. Supreme Court decision in the area of computer-program related inventions was *Gottschalk v. Benson*.[3] The Court found that the computer program in fact implemented a mathematical algorithm. By granting a patent the algorithm would effectively be monopolised.

In the case of *Diamond v. Diehr*[4] the Supreme Court for the first decided time that under certain conditions patents could indeed be granted for computer programs. The mere use of a mathematical algorithm or computer program would not prevent an invention from being patented. The decisive test is whether the invention involves the "transformation and reduction of an article into a different state or thing". That criterion was met by the invention at hand; it involved a computer-controlled process to cure synthetic rubber.

In a series of later cases[5] the so-called *Freeman-Walter-Abele* test was developed, involving two steps:

1. Is a mathematical algorithm recited directly or indirectly in the claim?

2. If so: is the claimed invention as a whole no more than the algorithm itself, i.e. is the algorithm not applied to physical elements or process steps?

If the answer to both these questions is affirmative, then no patent can be granted. Recently, this test has been criticised by the courts.[6] In particular, the requirement of "physicality" was found to be inappropriate. *In re Alappat*[7] the invention involved a software program for the transformation of numerical values in a manner that creates a smooth display of data on an oscilloscope screen, so there are no "physical elements or process steps". According to the Court of Appeal for the Federal Circuit the decisive test is whether an invention as a whole concerns a "disembodied mathematical concept". If so, the invention is not patentable. If on the other hand the invention produces a "useful, concrete and tangible" result, there is no objection against patentability. In other words the dependency of software-related invention on a mathematical algorithm by itself is no longer an impediment to patentability.

Another long-standing rule in U.S. patent law was the judicially created "business method exception". Both this exception and the "mathematical algorithm exception" previously discussed were addressed in the *State Street Bank and Trust Co. v. Signature Financial Group Inc.* case. [8]

---

[1] S.Rep. No. 1979, 82d Cong., 2d Sess., 5 (1952); H.R. Rep. No. 1923, 82d Cong. 2d Sess., 6 (1952). U.S.C.C.A.N. 1952, pp. 2394, 2399.
[2] *Diamond v. Diehr* 450 U.S. 175, 186 (1981).
[3] 409 U.S. 63 (1972).
[4] *Diamond v. Diehr* 450 U.S. 175, 186 (1981).
[5] *Freeman*: 573 F. 2. 1237 (C.C.P.A, 1978), *Walter*: 618 F. 2. 758 (C.C.P.A. 1980) *Abele*: 684 F. 2. 902 (C.C.P.A 1982).
[6] Fellas 1999, p. 331.
[7] 33 F. 3d 1526 (Fed. Cir. 1994).
[8] *State Street Bank and Trust Co. v. Signature Financial Group Inc* (149 F. 3d 1368, 1375 (Fed. Circ. 1998).

The *State Street* case involved a patent for a computer-based system for the pooling of mutual funds into a common investment fund, yielding economies of scale and tax benefits. A computer program implemented a number of complicated algorithms for the assignment of assets and expenses to the individual participants.
The District Court found that the invention could not be patented, as both the mathematical algorithm and the business method exceptions would apply.[1] With regard to the former exception the Court of Appeals for the Federal Circuit again emphasized that the Freeman-Walter-Abele test no longer applies. Instead, the decisive criterion is whether a "useful, concrete and tangible" result is produced.
With regard to the business method exception the Court noted that this is not really an exception by itself. Business methods may be more likely not to pass the tests of novelty or non-obviousness.[2] But really "since the 1952 Patent Act, business methods have been, and should have been, subject to the same legal requirements for patentability as applied to any other process or method". In addition, the Court noted that neither itself nor its predecessor[3] had ever recognized the business method exception. Whether certain subject matter is patentable is to be decided on the basis of § 101,[4] which does not provide for any exception for business methods.
In *AT&T v. Excel Communications Inc.*[5] the Court confirmed that in the case of process claims the scope of § 101 is to be interpreted in the same way as in *State Street*, which involved only apparatus claims.

Unlike in Europe, under U.S. patent law no requirement for "technical character" exists. U.S. law merely requires that the invention produce a "concrete, useful and tangible" result.

The U.S. Court of Appeals for the Federal Circuit has refused to recognize a "business method exception". Any business method that produces a "concrete, useful and tangible" result and meets the other requirements of patentability (i.e. novelty and non-obviousness) can be patented.

3.4. Japan

3.4.1. Statutory law

Art. 2 of the Japanese Patent Act[6] defines an invention as follows:

> "Invention" in this Law means the highly advanced creation of technical ideas by which a law of nature is utilized.

For computer programs, this effectively leads to a requirement similar to the European "further technical effect" test. In practice, however, it is possible to apply for patents for software in Japan, and even for business method patents.
As the patentability, Art. 29 reads as follows:

> (1) Any person who has made an invention which is industrially applicable may obtain a patent therefor, except in the case of the following inventions:

[1] 927 F. Supp. 502 (D. Mass. 1996).
[2] 35 U.S.C. § 103.
[3] The Court of Appeals for the Federal Circuit (CAFC) replaced the Court of Customs and Patent Appeals (CCPA) in 1982 as the intermediate court of appeals for patents.
[4] 35 U.S.C.§ 101 (U.S. Patent Statute).
[5] 172 F.3d 1352 (Fed. Circ. 1999).
[6] Law No. 121 of April 13, 1959 as amended by Law No. 220 of December 22, 1999. Entry info force: January 6, 2001, with updates entering into force on October 1, 2001. English text obtained from <http://www.jpo.jo.jp>.

> (i) inventions which were publicly known in Japan or elsewhere prior to the filing of the patent application;
>
> (ii) inventions which were publicly worked in Japan or elsewhere prior to the filing of the patent application;
>
> (iii) inventions which were described in a distributed publication or made available to the public through electric telecommunication lines in Japan or elsewhere prior to the filing of the patent application.
>
> (2) Where an invention could easily have been made, prior to the filing of the patent application, by a person with ordinary skill in the art to which the invention pertains, on the basis of an invention or inventions referred to in any of the paragraphs of Subsection (1), a patent shall not be granted for such an invention notwithstanding Subsection (1).

Subsection (1) comprises a requirement similar to Art. 57 EPC. Subsections (1)(i), (1)(ii) and (1)(iii) specify the requirements for novelty, similar to Art. 54 EPC. Finally, Subsection (2) lists the requirements for the inventive step, similar to Art. 56 EPC.

3.4.2. Examination Guidelines

Traditionally, the role of the courts in patent matters in Japan is fairly limited. In practice, the interpretation of the patent law is mostly up to the Japanese Patent Office (JPO). Thus the patentability of software patents in Japan is to be inferred in large part from the JPO's Examination Guidelines.

On December 28, 2000, the JPO released a new version of the "Examination Guidelines for Computer Software-related Inventions".[1] This document includes examination standards for business-method related inventions. Computer software can be patented under these guidelines under the following condition:

> Where "information processing by software is concretely realized by using hardware resources," the said software is deemed to be "a creation of technical ideas utilizing a law of nature."[2]

This refers to the basic requirement for patentability under the Japanese Patent Act.[3] It is different from the corresponding European requirement in that no "further" technical effect is required.

Similar requirements exist for business methods, which becomes apparent from the following - negative - example in the Guidelines:[4]

> When an information processing system to execute mathematical solutions, business methods or game rules is stated in a claim, there is no description in the detailed description of the invention on how to realize such methods or rules on a computer, so that the invention cannot be carried out.

There appear to remain few obstacles to software and business method patenting in Japan, as long as the claims are properly drafted and include references to hardware resources. For software patents, this requirement would hardly constitute an impediment. Business methods must be computer-based in order to be patentable.

3.5. International framework: the TRIPs Agreement

Among the various attempts that have been made for substantial harmonization of patent law on a global level, the Agreement on Trade Related Aspects of Intellectual Property Rights,[1] (TRIPs

[1] <www.jpo.go.jp/infoe/Guidelines/PartVII-1.pdf>.
[2] Guidelines p. 11.
[3] Available under <http://www.jpo.go.jp>
[4] Guidelines p. 8

agreement), is by far the most important. This agreement is part of the 1994 Marrakech agreement to found the World Trade Organisation, a treaty that was concluded on the basis of the 1947 General Agreement on Tariffs and Trade, GATT. The TRIPs Agreement sets global minimum standards for protection of intellectual property rights in a variety of legal areas, including copyright law and patent law.

What makes the substantive rules of the TRIPs Agreement particularly important, as compared to existing international norms in the field of industrial property, such as those incorporated in the Paris Convention of 1883, is that TRIPs compliance is subject to the dispute settlement mechanism provided for under the GATT agreement.[2] Non-compliance may eventually lead to retaliation by other WTO member states.

The TRIPs agreement is binding upon the European Community (a WTO member since 1 January 1995) and its Member States. Any legislative initiative in the field of intellectual property will, therefore, have to comply with TRIPs minimum standards.

Section II.5 of the TRIPs agreement is dedicated to patent law. Article 27(1) requires that "patents shall be available for any inventions, whether products or processes, in all fields of technology, provided that they are new, involve an inventive step and are capable of industrial application. (...)."

Proponents of software patenting have argued that Article 27(1) does not allow software from being excluded from patentability, since computer software is to be considered a "field of technology".[3] The discussions preceding adoption of the TRIPs agreement, however, do not confirm such a reading. In the absence of a legal definition of "invention", the agreement arguably leaves it to the member states to determine what constitutes a patentable invention, and whether or not that includes computer software as such.[4] In any case, it seems unlikely to assume that business methods as such are (part of) a "field of technology", so the TRIPs agreement does not appear to impose an obligation to offer patent protection for business methods.

Article 31 of the TRIPs agreement allows for compulsory licensing under a variety of detailed circumstances. As a means of mitigating some of the negative effects of software patents on competition, it has been suggested to resort to the instrument of compulsory licensing more regularly.[5] If and when this will happen, the limitations of Article 31 should be taken into account.

Finally, it is worth noting that Article 33 of the TRIPs agreement provides for a minimum term of protection for patents of twenty years counted from the filing date. Assuming that any future European regime with regard to software-related inventions will fall under the scope of TRIPs, this minimum term is a given. A shorter patent term for software-related inventions, as sometimes suggested in view of the supposed particular characteristics of software inventions, is not permitted. On the other hand, TRIPs does not prohibit the establishment of a sui generis protection regime outside the patent paradigm.

The slow progress of the work on the new Patent Law Treaty (PLT)[6] that was created under the auspices of the *World Intellectual Property Organisation* (WIPO) suggests that no new developments in substantive patent law harmonisation are to be expected soon. The PLT, as adopted in June 2000, contains no substantive rules at all. Earlier PLT proposals had included

---

[1] The text of the TRIPs agreement is available at <http://www.wto.org/english/docs_e/legal_e/27-trips.pdf>.

[2] See Gervais 1998, p. 241-252.

[3] Schiuma 1998, p. 852-858 (in German) , Schiuma 2000, p. 36-51 (in English). See also EPO Technical Board of Appeal T1173/97, *Official Journal of the EPO* 1999, 609, at § 2.3.

[4] Joseph Strauss, 'Bedeutung des TRIPS für das Patentrecht', *GRUR Int.* 1996, p. 179-205, at 191.

[5] Blind and others 2001, p. 231.

[6] <http://clea.wipo.int/PDFFILES/English/WO/WO38EN.PDF>.

plans for substantive harmonisation, but despite considerable effort the contracting states failed to reach agreement.[1]

In discussing the future of European patent law it is often all too easily suggested that Europe must inevitably follow in the footsteps of the United States. An argument in favour of following the U.S. model is that the market for much computer software comprises both Europe and the United States. If software-related inventions can be patented only in the U.S., such inventions would be in the public domain only in Europe. However, since a viable software product must be marketable both in Europe and the U.S., European inventors would still be forced to shop for patents in the U.S.[2]

Historically, the patent laws of many countries have indeed often followed the developments in the U.S., sometimes with considerable delay. However, in our opinion it would be wrong to suggest that Europe should automatically follow the U.S. once again with regard to software or business method patents. As we have noted before, the U.S. patent system is fundamentally different from existing European systems in that it is not, and presumably never has been, limited to inventions bearing technical character. Interestingly, as illustrated by recent developments elsewhere in the field of intellectual property, the United States have in some cases demonstrated a willingness to adapt its intellectual property regimes to European norms.[3]

The TRIPs agreement requires patent protection to be available in "all fields of technology". Arguably, this does not mandate patent protection of computer programs per se. Surely, TRIPs does not prescribe patentability of business methods.

The generally perceived need for world wide patent law harmonisation should not lead to the conclusion that the European patent system must follow U.S. developments automatically.

4. Observations and projections

In this chapter, we shift our focus from the legal framework to the realities of the patent system and its role in the area of software and business methods, in order to assess the possible needs for changes.

First, we will elaborate on our methodology. In particular, we will focus on the various consultations on software patents that were conducted recently. Next, we will give a brief sketch of "software" and the software industry in its endless variety. Then we will discuss the functioning of the patent system in the economy. Does the patent system promote innovation or does it create undesirable monopolies? Subsequently, we will elaborate on the role of the primary users of the patent system, the inventors. We will in particular pay attention to Small and Medium sized Enterprises and developers of Open Source Software. Finally, we will devote some words to the administration of the patent system by the patent offices.

4.1. Methodology

It is tempting to ask: is there any proof that the alleged inventor-friendly American patent policy leads to more innovation, and to more social welfare in general, as compared to the more conservative approach in Europe? It is just as tempting to answer this question right away by pointing out that the U.S. are hugely successful in the realm of computer software, and at the same time show a high patent activity.

---

[1] <http://www.wipo.org/about-ip/en/index.html?wipo_content_frame=/about-ip/en/patents.html>.

[2] And they may like it, because patent application is less expensive in the U.S. than in Europe.

[3] Recent examples are the extension of the copyright term by twinty years pursuant to the 1993 EC Term Directive, and the database right which was created in Europe as part of the 1996 EC Database Directive.

While the two are probably related, it might well be that the prosperity of the American software business has been the cause of this patent activity, rather than its result. Empirical data do not support any conclusions on the effects of changes to the patent system on innovation.[1] As we will elaborate below, the patent system brings about a complicated system of causes and effects, and by and large, there is insufficient information available to match individual effects to individual causes. In addition, there are transient effects due to the recent changes in patent policy in the U.S., which may take years to fade out. It would be premature to draw conclusions right now from data related to the present state – a state that most likely still has not stabilised.

As Jaffe points out, there are methodological and practical problems on several levels.[2] Clearly, coincidences appearing in statistics do not constitute any proof or even indication for causal relationships. At a more basic level the question has to be asked: what is a proper measure for "benefit to society"? Even if this is equated to "innovation", the question remains what a proper measure for innovation would be. And what changes in the patent system should be taken into account? Even if only the effects of "software patents" would be studied, the question arises: what is really a software patent? As we have seen, patents have been granted for computer program related inventions for several decades in one form or another. But as those inventions "as such" were not considered patentable, they do not show up as such in statistics, which complicates historical statistical research.

Whether software related inventions really represent a different category or not, the current discussions on software patentability typically refer (implicitly) to some sort of "regular" invention, of which patentability is beyond debate. It should be noted, however, that the role of the patent system varies considerably from one segment of industry to the next.[3] In fact, patents in all fields of industry are controversial to some extent. Indeed, the debate on software patents reflects discussions on the validity of the patent paradigm in general, in that age-old arguments for and against the patent system are repeated again and again.

Several recent studies[4] have concluded that it is virtually impossible to arrive at explicit conclusions on the desirability of software patents in general. In fact, it was found consistently that the notion of software itself, software industry players and their opinions are very diverse.

PbT Consultants[5] found rather polarised opinions in their analysis of a consultation in all EU member states. Proponents of a "liberal" (pro-patent) approach were found predominantly in circles of lawyers, established industry players, and government agencies. Proponents of a restrictive approach were found among students, academics, engineers, and start-up companies.

Similar results were found in a consultation conducted in Germany by Blind and others.[6] Again, independent developers were found not to be in favour of software patents. A distinction was made between a "primary branch" consisting of enterprises whose main aim is the development of software, and a "secondary branch" consisting of enterprises from manufacturing industry that develop software as well. However, opinions on software patenting appeared to be divided in both groups.

A French interdepartmental working group[7] distinguished four types of players: large suppliers of generic and infrastructure software together with suppliers of integrated software, suppliers of specialised application software and software components, software houses, and universities and researchers together. All these players again were found to express widely divergent opinions.

---

[1] Jaffe 1999 p. 20.
[2] See note 60.
[3] Hart, Holmes & Reid 2000, p. 31-32.
[4] See Annex 8.1 on page 38 for an overview of sources.
[5] PbT Consultants 2001.
[6] Blind and others 2001, p. 99-103.
[7] French interdepartmental working group 2001.

In the Netherlands, two consultation studies were performed within a few years. In 2000, Verkade, Visser and Bruining reported that both for large and small enterprises, the opinions were very divided: both might benefit, or suffer from software patent liberalisation, according to the respondents.[1] In 2001, the Dutch Ministry of Economic Affairs found that no decisive answers could be given on questions about the financial interest, the effects on competition, the value of patents for innovation or the incremental value of patent protection in addition to other means of legal protection.[2]
Similarly, the UK Government concluded in 2001 from a large consultation exercise that there was no consensus among respondents on the extent to which computer software ought to be patentable.[3]
Some consultations carefully selected their respondents, and explicitly claim representativity,[4] while other consultations were open to everyone over the Internet. In our view, any claim for numerical representativity must be reviewed critically, including corrections for alleged shortcomings in representativity. Apparent majorities may or may not be meaningful. Typically, most respondents were players in the software industry. Other stakeholders such as the buyers of software may not be adequately represented.
Still, the opinions expressed in the consultations are valuable. There is a remarkable consensus on the issues at stake, even if there is little agreement on the direction to go. Together, the consultations offer a comprehensive catalogue of key issues. Moreover, the divergence in opinions seems to be related largely to the diversity of software, software patents and the software industry. This observation by itself is important, and calls for further investigation and discussion, as we will elaborate below.
Finally, the lack of data on the way the patent system works in practice is by itself a conclusion that calls for action. The abundance of opinions, promises, theories, fears and concerns expressed about the patent system in the course of literally centuries is by no means matched by factual data. As early as 1958 Machlup observed: "*If we did not have a patent system, it would be irresponsible, on the basis of our present knowledge of its economic consequences, to recommend instituting one. But since we have had a patent system for a long time, it would be irresponsible, on the basis of our present knowledge, to recommend abolishing it*".[5] And we still do not know.
Of course, it would be difficult, if not impossible, hard to perform experiments with patents. But it certainly would be advisable to collect much more data on the actual operation of the patent system, preferably on a routine basis. We will come back to this later when we present our recommendations in the final chapter.

No hard empirical data exist today to support conclusions in favour or against software and business method patenting. Consultations consistently show widely varying opinions, but do provide valuable insights in the issues at stake.

4.2. Software and the software industry

4.2.1. "Computer program related inventions"

The colloquial term "software patent" actually refers to patents on "*computer program related inventions*". Just as the concept of "invention" in the sense of the patent law,[6] this is a rather abstract notion, which often leads to some confusion. In particular, software patents are

[1] Verkade, Visser & Bruining 2000, p. 26, no. 8 and 9.
[2] Dutch Ministry of Economic Affairs 2001, p. 70.
[3] UK Government 2001, no 11.
[4] Blind and others 2001 p. 49-51; PbT Consultants 2001, p. 4.
[5] Machlup 1958, p. 80.
[6] See Chapter 3: Legal Framework.

sometimes equated to patents on software products, rather than to patents on the inventions incorporated in those products. Software patents may relate to multiple products. Or software patents may not be related to any specific product at all, as long as the associated invention is "susceptible for industrial application" - which is required for patentability.

For example, in an analysis of German court decisions in the area of software patents[1] the following categories of software patents were found: measurement and control technology, Computer Aided Design/Computer Aided Manufacturing (CAD/CAM), digital signal processing, operating systems, auxiliary programs, data compression, customer management, text processing, table calculations, data encryption, programming tools, authentication, and time series analysis. Inventions in some of these areas were found to be patentable, while others were not. We will not discuss those decisions here; rather we want to stress that some of these inventions relate to products, while others typically do not represent products by themselves.

Computer program related inventions are not just diverse, they are also difficult to define and delimit. Machines are increasingly controlled by software-controlled processors (so-called *embedded processors*) rather than by hard-wired electronic circuitry; as has been decided by the EPO a long time ago,[2] patents should not be denied in such cases. From a functional point of view, such machines are similar to conventional machines. Only if a machine with an embedded processor is disassembled, it becomes apparent that there is actually a microprocessor chip inside. But from a technical point of view, such a chip is a computer controlled by a computer program not essentially different from any other computer program.

Due to the restrictions on the patentability of software, in order to obtain patent protection, software patents claims must be drafted in the form of hardware claims patents, at least in Europe. But this is not just a matter of words; there is also technology that is hard to categorise as either software of hardware. In modern electronics digital techniques abound, but probably not all of those techniques can be considered software related.

On the other hand, it is also argued that computer software is something fundamentally different due to its rather abstract character. As a consequence, software patents are deemed inappropriate, and would destabilise the patent system.[3]

In the controversy between those who argue that software is nothing special, and those that insist that software is something very special, we feel it would be inappropriate to take a position. There seems to be some truth in either statement. But the debate demonstrates that there is one firm conclusion to draw: it is dangerous to make general statements on "computer program related inventions". The current controversy on software patents in our view can be attributed at least partly to confusion about the subject matter.

Software is very diverse. Generalisations should be critically assessed.

4.2.2. The computer software industry

Most of the studies undertaken show that *the* software industry actually does not exist. Rather it is a very diverse industry, ranging from one-man shops to multinational conglomerates. While many think of Microsoft as *the* example of a software company, in fact a single properly equipped PC is enough for developing software, and independent programmers often make important contributions to software technology.

Consequently, any generalisations about "the" software industry must be carefully reviewed. For instance, the assertion that the software industry is characterised by comparatively short

[1] Blind and others 2001, p. 140
[2] T26/86, *Official Journal of the EPO* 1988, p. 19 (*Koch & Sterzel*).
[3] Lutterbeck, Gehring & Horns 2000.

development cycles[1] is based on a particular, limited view of that industry. While a new version of Microsoft Windows is released just about every year, fundamental techniques may live much longer. The "DES" Data Encryption Standard was developed in 1977, and is still in use today. It has also been claimed that "incremental development" is more common in the software industry than in other industries.[2] This may be true for some parts of the software industry. But there are no signs that this is a feature unique to computer program related inventions.

Even so, some observations seem to apply to most types of computer program related inventions. Firstly, software is rarely used in isolation, but mostly in combination with other software or hardware. Interoperability, therefore, is essential. Communications networks such as the Internet as well as stand-alone computers call for interoperability, not merely for data exchange, but also for software and (peripheral) hardware to work together. Interoperability is particularly important because operating systems easily become de facto standards, making application software (e.g. word processors) dependent on (often proprietary) interfaces.

Secondly, the software market is generally perceived as a global market. While there are some obvious exceptions,[3] software, and in particular the underlying inventions that would qualify for patent protection, typically are applicable on a world wide scale.

Previous studies have made various distinctions in the software industry. In Germany, a "primary" and a "secondary" branch were distinguished, depending on whether software development was the main business activity of an enterprise.[4] In France, a distinction was made between producers of infrastructure and generic software, producers of embedded software, producers of specialised software and software components, software houses and the academic world (in its quality of potential patent applicant).[5] In the Netherlands, it was emphasised that *the* software sector simply does not exist.[6] The UK government report also underlines the "enormous diversity" of the software industry.[7]

The diversity of the software industry adds to the diversity of "computer program related inventions" that we have observed before. All in all it comes as no surprise that such diverse players have widely divergent opinions on the merits of patenting; even a single player can have conflicting interests, depending on its role in the software industry.

The software industry is very diverse. Again, generalisations should be critically examined.

4.3. The functioning of the patent system in the economy

4.3.1. Patents and innovation

Patents are supposed to stimulate innovative activity, and innovation is assumed to increase competitiveness and economic welfare in general. So the first question to ask ourselves is whether patents indeed increase innovative activity to such an extent that the – potentially harmful – patent monopoly is justified, and whether such an increase can not be achieved by alternative means.

Undoubtedly, in some sectors of industry it would be nearly impossible to stimulate investments in research and development without the possibility of patent protection. This is true in particular for industries that require large investments in R&D, and where competitors can

[1] Blind and others 2001, p. 59.
[2] Blind and others 2001, p. 60.
[3] E.g. Dutch speech recognition software will not be widely used outside a few European countries.
[4] Blind and others 2001, p. 44.
[5] French interdepartmental working group 2001, p. 9-12.
[6] Dutch Ministry of Economic Affairs 2001, p.12
[7] UK Government 2001, nr. 14.

easily reproduce inventions once they are applied in marketed products. A prime example is the pharmaceuticals industry; it may require substantial efforts to discover a chemical compound having a certain medicinal effect, but once it is found, it might be relatively easy to manufacture.

Would software development and innovation be commercially unattractive if no patent protection were available? The answer is clearly negative; software development has thrived for many years absent patent protection. Even today, whether legally possible or not, most software innovation is not actually being protected by patents.

But even if patent protection is not necessary to stimulate investment in R&D, the patent system can supposedly promote innovation indirectly by means of its "information function". If a patent is granted, it is disclosed (Art. 83 EPC). Other inventors may use the disclosed invention to build on while working on new inventions. The information function of the patent system might be a reason in its own right to justify patent protection. Of course, this function is only relevant in those cases where it is technically or practically possible to keep the invention secret. This may or may not be the case for software inventions. Inventions involving programming techniques may indeed be hidden in "object code" – the form in which computer programs are shipped to users.

It is, however, doubtful whether the information function actually plays an important role in the software industry. German research[1] shows that patent archives are primarily valued as a source of information for legal purposes, and only to a lesser extent as a source of technical information. Whether this conclusion applies to all kinds of software patents has not been investigated. Proponents of the patent system often argue that patents as a source of knowledge need more promotion. That the patent archives might indeed be useful for a wider circle of potential users, in particular for Small and Medium Sized enterprises, is, however, an assumption that is made without proof.

Patents may actually have adverse effects on innovation. [2] Inventions often build on each other; second inventors will therefore need to seek licences for patented prior inventions. The burden or even the impossibility of obtaining such licenses may impede further innovation. Patent protection may even stifle innovation altogether by creating monopolies in core inventions that are difficult, or even impossible, to design around. Absent legal protection competitors are free to imitate inventions; the original inventor must keep on innovating in order to stay ahead of the competition. This type of "arms race" is in fact the rule in business competition, where it is generally perceived as beneficial to the economy.

It has not been demonstrated that software patents contribute to innovation. The opposite may be true as well.

4.3.2. Patents, monopolies and "network effects"

The exclusive rights granted by a patent diminish competition, and interfere with market mechanisms. A patent owner enjoys a monopoly that enables him to ask a higher price. To a certain extent, this is an unavoidable and accepted consequence of the patent system. Specifically for software it is a well-known fact that even without patents there is a natural tendency towards monopolies due to a need for standardisation. As we have previously observed, software is often used in conjunction with other software. A word processing program will run only on a given operating system; word processor files are exchanged between users, by e-mail, on diskette or otherwise. Such software will therefore become more attractive if it is

[1] Blind and others 2001, p. 94-95.
[2] Bessen & Maskin 2000.

used on a larger scale. In economic terms,[1] goods that are valued more highly as the user group increases are said to have *network effects*. Network effects have been known in the IT industry for a long time. Typically, those effects show "*positive feedback*",[2] causing goods that are slightly more attractive than other products (by virtue of their quality, or for other reasons) to become increasingly attractive. On the other hand, slightly less attractive goods may become very unattractive over time.

Patents may amplify these network effects. If the document file format used by Word for Windows would be protected by a patent, it would not merely be difficult but legally impossible (absent a license) to exchange documents with other word processing programs. Conversely, it has been argued that inventors may refrain from patenting in such situations, as patents might hinder the development of such lucrative "networks".[3]

4.3.3. Strategic use of patents

Patents may be used in an aggressive manner to fight competition by means of patents rather than by performance. Again, while this phenomenon is to a certain extent inherent to any patent system, undesirable excesses may occur. Patents are said to be used in a "strategic" way if the owner employs his patents merely to prevent competitors from using the invention, rather than to exploit the invention himself. In a broader sense, strategic use of patents could also be considered to include other actions specifically targeted at the obstruction of competitors.[4]

The patent system may also be abused to fight competitors by filing unjustified infringement lawsuits.[5] Even the threat of such a lawsuit may deter competitors.[6] A lawsuit that is eventually lost by the plaintiff still imposes a heavy burden of time and cost upon the defendant. There is reason to believe that the risk of frivolous patent infringement claims may be relatively high in the area of computer software, as compared to other segments of industry, because many existing software patents may actually lack novelty or inventive step. We will revisit this problem of "patent quality" later in this report.

Aggressive use of patents calls for defensive measures. Again, this is normal in any patent system, and there is no reason for concern, as long as it does not happen in a disproportionate way. To a certain extent, all patents serve a defensive purpose; a patent owner can prevent others from applying the technology he has developed. A purely defensive use of patents may be the filing of patents with the sole objective of creating an exchange object that can be used to obtain a licence for another patent from a competitor who would otherwise be reluctant to grant a licence. In view of the very limited possibilities of compulsory licensing under most patent laws, this may in fact be the only way to obtain a licence. Patents are said to be used in such a defensive manner on a large scale, [7] forcing enterprises to build patent portfolios with no other purpose than to serve as exchange objects.

Another defensive measure sometimes employed is to pre-empt patent claims by disclosing inventions.[8] In either case, the disclosure of technical knowledge can still be valued as a positive result. However, the amount of work involved, even in the case of defensive publishing, can lead to huge costs that cause the overall balance to be negative. This work will consume the

---

[1] Shapiro & Varian 1999.

[2] Rather confusingly, *positive* feedback denotes an undesirable form of feedback. While regular *negative* feedback provides a correction, positive feedback amplifies effects to the extreme.

[3] Blind and others 2001, p. 21; Hart, Holmes & Reid 2000, p. 31.

[4] Actual strategies may vary from a single "blocking" patent to more complicated constructions such as "thickets". See for instance Rivette & Kline 2000.

[5] Hart, Holmes & Reid 2000, p. 37.

[6] French interdepartmental working group 2001, p. 15.

[7] Hart, Holmes & Reid 2000, p. 32.

[8] The U.S. patent law allows the registration of defensive publications (35 U.S.C. § 157).

time of highly skilled people, that would otherwise be devoted to development work leading to new inventions. In particular, this could be a serious problem for SME's.
All in all the defensive use of patents might result in a very expensive arms race which eventually has little practical effect for those who are able to play the game, but which may present a formidable obstacle to new entrants to the market. Thus patenting may lead to a concentration of the software industry, and a shakeout of SMEs.[1] Needless to say, such a result is utterly undesirable.

4.3.4. Patenting "Business Methods"

To many European observers business method patents represent a horrific prospect – yet another example of unwanted 'Americanisation'. Even those who are in favour of software patenting usually are vehemently opposed to patenting business methods.[2]
There is some confusion over what these patents really are. Some equate business method patents to "e-business" or "Internet" patents. Others perceive business method patents as a variety of software patents; indeed, references to software in a business method patent claim may enhance patentability.
While some Internet-related patents have received a lot of publicity,[3] the notion of a business method patent covers a much wider area. The current debate over business method patents was triggered by the *State Street* decision in the United States discussed above.[4] Business method patents have become common practice in the U.S. ever since. In the 2001 fiscal year around 10.000 applications were expected in class 705 "Modern Business Data Processing", which represents an important (but not the only) category of business method patents; nearly 50% of applications is actually granted a patent.[5] The growth of business method patent applications in the U.S. appears to have stabilised somewhat in recent times. This might be attributed to the consolidation in the Internet industry.[6]
There is a tendency to view the surge of business method patents as a transient fashion, as a mistake that will be corrected soon. But there are signs that this may not be true. [7] Admittedly, inadequate examining has lead to business method patents of questionable quality; many such patents relate to inventions that are trivial at best. But the U.S.P.T.O. is currently trying hard to improve business method patent quality.[8]
Still, there are important reasons to be very sceptical of these American developments. If really "*anything under the sun made by man*" would be eligible for patent, as long as it leads to something "*concrete, useful and tangible*", the patent system would be opened to areas of human activity traditionally considered way beyond the realm of the patent system. This is not just theory; in the U.S. patents have been granted in areas such as architecture,[9] auctions,[10]

---

[1] Blind and others 2001, p. 105.
[2] French interdepartmental working group 2001, p. 14; Blind and others 2001, p. 224, UK Government 2001, nr. 21.
[3] The prime example is the infamous "one click" amazon.com patent. See for instance Parker 2001, p. 12, <http://www.amazon.com/exec/obidos/subst/misc/patents.html/002-0977844-7404807> and <www.noamazon.com>.
[4] See § 3.3.2 on page 10.
[5] U.S.P.T.O., *Business Methods Still Experiencing Substantial Growth - Report of Fiscal year 2001 Statistics*, <http://www.uspto.gov/web/menu/pbmethod/fy2001strport.html>.
[6] Sterne & Lee 2002, p. 8.
[7] Lang 2000, Maier & Mattson 2001, Parker 2001, Toren 2000, Zoltick 2000.
[8] *A USPTO White Paper. Automated Financial or Management Data Processing Methods (Business Methods)*, <http://www.uspto.gov/web/menu/busmethp/whitepaper.pdf>.
[9] For instance U.S. Patent No. 5,761,857. U.S. patents can be found by number on <http://patft.uspto.gov/netahtml/srchnum.htm>.
[10] For instance U.S. Patent No. 5,897,620.

investment,[1] marketing,[2] personal instruction,[3] psychological analysis,[4] sports,[5] language,[6] and even theology.[7] More (sometimes exotic) examples can be found in literature.[8]

The question can and should be asked, whether patent protection is justified in those areas. For a proper assessment, one should revert to the basic objectives of the patent system. A patent is not a "natural" right, but an instrument of economic policy. As such, it must be judged on its economic merits. Wouldn't inventive business methods be developed without the incentive of a patent? Would business methods otherwise be kept secret absent patent protection? Does the 'enabling disclosure' of business method inventions serve any useful social purpose? And if there are indeed such benefits, are they so great that the inevitable negative effects on competition would be justified?

Many argue that "business methods" really are the core of business competition. Sound competition typically involves the continuous development of new ideas, to attain and maintain a competitive advantage. Businesses have developed over centuries without the incentives of the patent system. Granting patents on business methods would effectively eliminate business competition at a very basic level. Also the beneficial effects of disclosure must be called into doubt; business method "inventions" can hardly be kept secret while applying them in the market place.

In the U.S. a substantial number of business method patents has been granted or applied for in the four years since the *State Street* decision. Not surprisingly, there is little empirical data available to assess the positive or negative impact of business method patenting. In most countries, including the United States, examination of prior art and inventiveness take place before patents are granted; in practice this may require a considerable period of time. For new categories of subject matter, such as business methods, the lead time may be even longer than usual. Another reason for this lack of empirical data are the start-up problems the U.S.P.T.O. is currently experiencing. Moreover, the economic effects of business method patenting may become apparent only after several more years.

Early experiences with business method patenting do not seem to suggest a positive effect on the economy. A number of business methods patent grants have received a huge amount of negative publicity.[9] This may not prove that *all* business method patents are bad, but does illustrate that things *can* go very wrong.

Having said this, it must be noted that business method patents are already being granted in Europe today, albeit on a more limited scale than in the U.S. As we have seen before, if business method claims are drafted in such a way as to include a "technical effect", such inventions may indeed qualify for patents.[10]

Business method patents constitute a departure from basic European patent system principles. There is strong opposition against such patents in Europe, and there are doubts as well in the U.S. However, business method patents today are a fact that cannot be ignored.

[1] For instance U.S. Patent No. 5,809,484.
[2] For instance U.S. Patent No. 5,668,736.
[3] For instance U.S. Patent No. 5,851,117.
[4] For instance U.S. Patent No. 5,190,458.
[5] For instance U.S. Patent No. 5,616,089.
[6] For instance U.S. Patent No. 4,864,503.
[7] For instance U.S. Patent No. 5,734,795.
[8] Thomas 1999, p. 33.
[9] For a summary, see <http://www.noamazon.com/>.
[10] For examples see § 3.2.3.

4.4. Small and Medium-sized Enterprises

Small and medium-sized enterprises are generally supposed to be more innovative than large companies. Whether this is true is beyond the scope of this study. But certainly in the software development industry SMEs, and even one-man shops, are in a good position to develop innovative products, because investments needed for software development are relatively modest, compared to e.g. the pharmaceuticals industry. So it is particularly important to assess the potential effects of a liberalised patent regime on these institutions.

Opinions vary whether SMEs will benefit or suffer from such changes. On the positive side, patents may help SMEs to capitalise their inventions into assets that can be sold or licensed. While SMEs may be just as innovative as larger businesses, or even more innovative, they have fewer resources to exploit their inventions. In those cases, it is important that inventions can be protected by patent.[1] Patents can also help to raise (venture) capital; patents represent assets that can be included on the company's balance sheet and thereby increase its creditworthiness. Moreover, SMEs might benefit from information that is disclosed through patents granted to others, even if they do not apply for patents themselves.

A commonly expressed belief is that SMEs might benefit more from the patent system, both from its protection and its information function, if only they would be better informed of its merits. SMEs, it is often suggested by proponents of the patent system, should be educated on the benefits of patenting.[2] In our opinion SMEs' relative lack of interest in the patent system might have a more rational explanation. Possibly, SMEs make conscious decisions to stay away from patents, for a variety of reasons –costs and effort being important factors. Typically, patenting (or defensive action against patent claims) will take time from the same few people that are the creative brains behind the inventions. In practice, many SMEs tend to rely on copyright protection for software, which is readily available and may provide all the protection SMEs really need.

On the down side, a proliferation of patents has severe drawbacks for SMEs. Development activity by SMEs may be blocked by patents held by major companies. Unlike larger enterprises, most SMEs will not own patent portfolios for the purpose of cross-licensing. Defensive actions against patents and defensive patenting may be prohibitively expensive or time-consuming. Finally, SMEs have to be careful not to infringe inadvertently on the patents of others. This risk is particularly high in cases of "trivial" patents. While such patents are likely to be invalidated by the courts, most SMEs simply do not have the resources to pursue a case, even if it would eventually be won.

Proponents of the patent system all too easily assume that patents are good for SMEs. In fact, patents can have serious negative consequences for SMEs. For many software-developing SMEs software copyright protection is probably sufficient, obviating any need for patent protection.

4.5. Open Source Software developers

Objections against software patents have been particularly strongly voiced by the developers of Open Source Software and their supporters.[3] Open Source Software is generally considered very important for innovation in the computer software industry.[4] But there is a fear that this valuable phenomenon may suffer disproportionally from inappropriate use of the patent system.

---

[1] The alternative of trade secret is much less attractive, as we discussed in § 3.1.

[2] For a study specifically on the interests of SMEs see Tang, Adams & Paré 2001.
See also Hart, Holmes & Reid 2000 p. 5,6; French interdepartmental working group 2001, p. 7; Blind and others, p. 229; Dutch Ministry of Economic Affairs 2001, p. 35.

[3] PbT Consultants 2001, p. 3.

[4] Blind and others, p. 227.

The prime example of Open Source Software is Linux, a computer operating system that was initiated by Linus Thorvalds, at the time (1991) a student from Finland. Since then Linux was further developed by an ever-larger "virtual community" on the Internet.[1] Linux and its components are made available free of charge, and in "source code", allowing the programs to be enhanced, extended, and modified by other programmers. Remarkably, the apparently unorganised "virtual" community of Linux developers, without any commercial intentions, has managed to create a system that is to be considered a viable and attractive alternative for commercial, proprietary operating systems even by such major players as IBM.[2] And there is much more software that has been developed following the Open Source Software model.[3]

Open Source Software is earning much sympathy not just because it is free of charge, but also because of its good quality and its flexibility. Open Source operating systems are seen by many as the only viable alternatives to the omnipresent proprietary Microsoft systems. Responses from consultations indicate that the IT industry depends on Open Source Software to a considerable extent.[4]

The striking success of Open Source Software has lead to the Open Source Initiative and other movements that endorse and promote the "Open Source model".[5] On many occasions, proponents of fears Open Source Software have expressed their fears of the potentially detrimental effects of a proliferation of software patents. Because of its "openness", any (inadvertent) patent infringement by Open Source Software developers will be relatively easy to prove, whereas commercial software developers are sheltered by the practice of distributing software in "object code".

Also, the problem of "trivial" patents mentioned before might be more of a threat to Open Source Software developers than to commercial software producers. Many members of the Open Source community will not be in a position to routinely consult patent archives, so the risk of accidental infringement is relatively high. Even if trivial patents can be invalidated by the courts, the problem of litigation costs and efforts remains. If trivial patents are considered valid, Open Source Software developers will inevitably become dependent on patent licences on a massive scale. Commercial software developers may not be willing to grant such licenses to Open Source Software developers, assuming the latter could afford them. It has even been suggested that the Open Source Software community should play the same game, and build patent portfolios for cross-licensing purposes.[6] That would, however, be completely alien to the "culture" of Open Source Software.

For producers of commercial software the Open Source Software movement presents a direct threat – a new form of competition. Some vendors of commercial software may find it difficult to compete against these newcomers using conventional commercial means. This is not the place to speculate about possible strategies of commercial software vendors, but if they would be able to compete using legal means such as the patent system, there is no guarantee they would not do that.

---

[1] *What is Linux?* <http://www.linux.org/info/>.

[2] <http://www.ibm.com/linux/>.

[3] Examples include the "Apache" web server software package, which is used for more than 50% of the worlds websites (see <http://www.securityspace.com/s_survey/data/index.html>), the "BIND" software that is commonly used for Internet domain name servers, and the "SENDMAIL" software that relays most electronic mail that is sent over the Internet (see <http://www.opensource.org/docs/products.html>).

[4] Blind and others 2001, p. 31.

[5] Such as: *Free Software Foundation*, <http://www.fsf.org>; *League for Programming Freedom*, <http://lpf.ai.mit.edu>; *Förderverein für eine Freie Informationelle Infrastruktur* (Germany), <http://www.ffii.org>; *Association Francophone des Utilisateurs de Linux et des Logiciels Libres* (France), <http://www.aful.org>.

[6] Hart, Holmes & Reid 2000, p. 3-4.

Because of its vulnerability to patent claims, it has been suggested to introduce special legal provisions to protect developers of Open Source Software.[1] In our opinion, solving the problem of "trivial" software patents, that constitutes the most serious threat, might obviate such legislative intervention.

Developers of Open Source Software are relatively vulnerable to patent infringement claims, particularly in respect of "trivial" patents. Rooting out such patents may obviate the need for special protection of Open Source Software developers.

4.6. Patent offices

As patents proliferate into new areas, and the number of patent applications increases, *patent offices* are faced with the problem of keeping their capacity and skills up to date. If there is a sudden surge in demand, problems may occur due to a lack of skills, knowledge or capacity. These problems can not be solved overnight; new examiners have to be hired and trained, and knowledge bases have to be built containing information on the state of the art in these new areas.

In the United States the *State Street* decision has resulted in an enormous increase in the number of business method patent applications. Because the U.S.P.T.O. had not anticipated this decision and was not properly prepared for this influx of new applications, a proper examination for novelty and non-obviousness of these new claims could not be carried out for some time. As businesses typically do not record their methods in publicly available documents, the collection of prior art information in this area is very difficult. As a consequence, the patent system in the U.S. was said to be in crisis.[2]

Faced with increasing backlogs, patent offices may be inclined to become less critical in examining statutory patent requirements. Appeal procedures are time consuming, and can be avoided by accepting dubious or even unjustified patent claims. A mitigating factor in Europe may be that, unlike the U.S. patent system, the European Patent Convention[3] provides for the correction mechanism of an opposition procedure.

Courts and Boards of Appeal are essential to counterbalance patent offices decisions. It is not always easy for a court to judge patent infringement. While the skills and the knowledge of patent offices is already a scarce resource (as we have noted above), the skills and knowledge of courts in these areas may be even more so. Patent offices may be called in for advice, with the risk of jeopardising the courts' independence. In practice, relatively few patent cases are tried in court, and the consistency of court decisions is questioned.[4]

Proper patent examination for novelty and non-obviousness is crucial, but is problematic, in particular regarding new subject matter such as software and business matters. For a variety of reasons – lack of records on prior art; constraints of time and resource – patent quality may suffer.

The role of the courts in critically reviewing patent office decisions is essential.

[1] Blind and others 2001, p. 227.
[2] Merges 1999, p. 577-615.
[3] Art. 99 EPC.
[4] Blind and others 2001, p. 136-139.

4.7. Conclusion

In the end the question remains: what makes computer software so special, as compared, as compared to subject-matter traditionally protected by patents? Are existing objections against software patenting simply expressions of a desire not to allow the ever-controversial patent system to "contaminate" the relatively new software industry?

The special nature of the software industry may turn the scale into the negative. Patents may not really be needed to stimulate investments in software development, while, on the other hand, software might be more susceptible to the potential negative effects of patents. Software development really is a matter of "ideas", and it may not be appropriate to grant exclusive rights on many of those ideas, even if they are novel and not obvious. Still, there do not seem to exist compelling reasons to deny patentability to all software-related inventions in principle.

But how to distinguish computer software that is eligible for patents from software that is not? Making such a distinction is not an easy task. If the rules are improperly applied, there is a risk that the distinction becomes part of the problem rather than part of the solution. It is doubtful whether a proper line can be drawn by applying a requirement of technical character, effect or contribution. As we have seen, software-related inventions can be *made* "technical" by drafting patent claims in such a way as to meet this requirement. We will explore this question further in § 6.1 of this report.

But it is not simply a practical problem to make a proper distinction. There is also the more fundamental question what criteria would best serve the basic objectives of the patent system. The concerns commonly heard about software patents predominantly relate to trivial software patents rather than to non-technical software patents. In the end it may be more important to apply a proper standard for non-obviousness than to define some imaginary, precise criterion of "technicality".

In conclusion, it seems inappropriate to ask for the benefits and disadvantages of software patents in general. Computer software is such a diverse phenomenon that any attempt to arrive at conclusions claiming general validity are bound to fail, as has been demonstrated by a wealth of studies and consultations. In our view, future efforts should be directed at defining and enforcing a proper distinction between patentable and non-patentable software. For that purpose, it might be more important to prevent "trivial" software patents than to prevent "non-technical" software patents.

> The crucial question is not *whether* software patents should be allowed, but *what* software patents should be permitted. Applying or restating a requirement of technical character may not solve this problem entirely. Future solutions should be aimed at increasing patent quality, i.e. by preventing the granting of "trivial" software patents.

5. The proposed European software patent directive

On 20 February 2002, the European Commission launched a proposal for a directive on the patentability of computer-implemented inventions.[1] In this section we will first summarize the proposed directive, and, next, provide commentary.

[1] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>.

5.1. Summary of the proposed directive

The objectives of the proposed directive are to resolve the current legal uncertainty in the area of software patents, and to approximate existing national patent law in the EC Member States. Rather sensibly, the proposal does not envisage any major changes in the existing requirements for patentability for software-related inventions in the European Union.

In particular, the proposed directive confirms and emphasises that a "technical contribution" is required for software inventions to be patentable. This would exclude computer programs "as such", and business methods "as such" – in line with current legal requirements. No further definition or explanation is given of the notion "technical". According to the European Commission, this notion may develop over time in case law as technology progresses.[1]

The proposed Directive will oblige the Member States to amend and apply national patent law in compliance with its provisions. Therefore, the Directive will affect both national and European patents. Even though the Directive leaves the provisions of the European Patent Convention (which is not part of community law) unchanged, Member States' courts will be bound to interpret national and European patents in accordance with the Directive, as eventually transposed into national law. Note that the EPC only governs the *granting* of European patents; national laws govern European patents *after* the grant (Art. 64 EPC).

5.2. Comments

The explanatory memorandum accompanying the proposal[2] starts with a lengthy exposé of the economic importance of computer software and the alleged benefits of software patents. In this context, the consultation by the Commission various economic studies are mentioned.[3] It is pointed out that none of these sources arrive at an unambiguous positive conclusion on the benefits of software patenting. The Commission correctly observes that "any move to strengthen intellectual property protection in the software industry cannot claim to rest on solid economic evidence". The proposal identifies concerns about software patents, such as the grant of "clearly invalid patents" in the U.S., the risk that software patents may benefit big players and hurt small players on the market, and the increase of economic cost because of the increased need for patent searches and licence negotiations. None of these objections is countered with convincing arguments.

Typically, the Commission invokes the lack of conclusive evidence as an argument favouring software patenting. It is said that "the study acknowledges that it has not been shown that that these reservations outweigh the positive effects of the patentability of computer-implemented inventions in the U.S.".[4] Elsewhere it is observed that "the study[5] finds no evidence that

---

[1] *Proposal for a Directive on the patentability of computer-implemented inventions - frequently asked questions*, <http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm>.

[2] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>.

[3] "*The Economic Impact of Patentability of Computer Programs*", Hart, Holmes & Reid 2000, Wesley M. Cohen, Richard R. Nelson,, and John P. Walsh, *Protecting their Intellectual Assets*: *Appropriability Conditions and why U.S. Manufacturing Firms Patent (or not)*, Working Paper 7552, National Bureau of Economic Research, February 2000; James Bessen, and Eric Maskin, , *Sequential Innovation, Patents, and Imitation, Working Paper, Department of Economics*, Massachusetts Institute of Technology, January 2000; Adam B. Jaffe, *The U.S. Patent System in Transition: Policy Innovation and the Innovation Process*, Working Paper 7280, National Bureau of Economic Research, August 1999.

[4] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>, p 6.

[5] "*The Economic Impact of Patentability of Computer Programs*", Hart, Holmes & Reid 2000.

European independent software developers have been unduly affected by the patent positions of large companies or indeed of other software developers".[1]
The Commission notes, that "the individual responses to the consultation were dominated numerically by supporters of open software", who are opposed against software patents. However, it is concluded that "there seems little doubt that the balance of economic weight taking into account total jobs and investment involved is in favour of harmonisation along the lines suggested".[2] It might be concluded as well that there are sincere, strong and justified concerns among large numbers of Open Source Software users. While Open Source Software may affect the income of some commercial software vendors, its value for the IT industry as a whole should not be underestimated.[3]
The ever-repeated observation, also in the explanatory memorandum,[4] that SMEs should be educated on the benefits of software patenting raises the question whether the lack of popularity of patents in these circles really is caused by ignorance. Perhaps SME's deliberately stay away from patents based on a conscious trade-off.
In line with existing European case law, the starting point for the Commission's legal position is that a *technical* contribution is required for software inventions to be patentable. An extension to computer programs "as such" is to be avoided.[5] An explicit codification of the technical contribution requirement is expected to reconfirm and clarify this requirement. The Explanatory Memorandum, however, offers little guidance on how to define this notion of "technicality". According to the Commission it would be impossible to give a proper definition in the context of a directive, as the meaning of "technical" is likely to change as technology develops. Case law would allow flexible adaptation to changing needs.[6]
As we have seen before, distinguishing between patentable and non-patentable subject matter is a complicated endeavour. In particular, some computer software is considered "technical", such as graphics manipulation software, while other apparently similar software is not, such as software for the manipulation or Arabic characters. Most importantly perhaps, the technical contribution criterion does not prevent all business methods to be patented, as some business methods may be considered to include a technical contribution, e.g. if a conventional machine is involved which shows an inventive step. This is admitted in the explanatory memorandum.[7] In our opinion the "technical contribution" requirement as laid down in the proposed directive may actually fail to improve legal certainty,[8] and it may not prevent the granting of undesirable patents, such as business method patents.
Contrary to current case law of the EPO Board of Appeal, the proposed directive will no longer allow patents on "computer program products" (e.g. software on a CD-ROM), even if the

---

[1] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>, p 6.
[2] *Proposal for a Directive on the patentability of computer-implemented inventions - frequently asked questions*, <http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm>, p. 5.
[3] See for instance Blind and others, p. 227.
[4] *Proposal for a Directive on the patentability of computer-implemented inventions - frequently asked questions*, <http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm>, p . 4.
[5] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>, p. 11.
[6] *Proposal for a Directive on the patentability of computer-implemented inventions - frequently asked questions*, <http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm>, p . 5.
[7] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>, p 11.
[8] Basinski 2002, p. 4.

invention concerned would involve a technical contribution. Only patents on programmed apparatus or processes carried out by such apparatus through the execution of software will be permitted. This will no longer permit apparatus claims for a computer program products "either on a carrier or by itself". In practice, this might complicate proof of infringement. It will no longer be possible to sue a manufacturer or reseller of patented software for direct patent infringement.[1] Apparently the Commission is concerned that any patentability of computer programs "by themselves" might be construed as patentability of computer programs "as such".[2]

Since our report was finalised shortly after the Commission published its proposal, only a few early reactions to the proposal could be taken into account. Most of these originate from the Open Source Software movement and its supporters. In these circles there is great anxiety that the proposed directive will, one way or another, enable patents on *all* software and business method patents.[3] While these concerns may not be fully justified, experience has indeed demonstrated that "technical" patents may include (potentially undesirable) software and business method claims.

The proposed directive devotes little attention to the problem of "trivial" patents. Objections raised against software and business method patenting often relate to the large numbers of allegedly trivial patents that have been (or are being) granted in these areas. As we have learned from the U.S. experience, such patent quality problems may actually destabilise the patent system. In our opinion, the proposal should have squarely addressed this issue, e.g. by way of a recital recognizing the need to preserve patent quality, and stressing the importance of properly applying the test of inventive step in respect of software-related patent claims.

We will discuss the problem of patent quality and a few related issues not directly addressed in the directive in greater detail in the following chapter.

The proposed directive is unlikely to meet its objective of improving legal security. It may not prevent undesirable software or business method patents. Also, the proposal fails to address the problem of patent quality, i.e. the granting of "trivial" software patents.

6. Further issues

The proposed directive concentrates on distinguishing patentable from non-patentable software by requiring "technical contribution". In this chapter we will examine the merits of this criterion. Subsequently, we will address the important problem of "triviality". Finally, we will spend some words on the idea of establishing a "Patent Observatory".

6.1. The requirement of "technical character"

Even though it is not immediately apparent from the text of the EPC, it has always been assumed that the European patent system is confined to *technical* inventions. In view of widespread concerns against business method patents it seems logical to reconfirm that software patents are available only for *technical* software inventions. Experience has shown, however, that applying the test of "technical character" does not always produce satisfactory results. The reasons for this can be categorised as legal security issues on the one hand, and questions regarding the relevance of this criterion on the other hand.

---

[1] Basinski 2002, p. 4.

[2] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions* (presented by the Commission) <http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf>, p. 15, explanation or Article 5.

[3] For a summary, with links to other material, see *The European Commission wants Unlimited Patentability! Proposal Written by BSA*, on <http://www.eurolinux.org>.

As we have seen in Chapter 3, patent applications can be easily drafted as to claim a "technical character" in respect of apparently non-technical subject-matter – not just for software, but even for business methods. Various publications provide comprehensive guidelines for patent applicants and attorneys on how to draft patent claims in order to pass the hurdle of the requirement of "technical character".[1] Some commentators have even argued that the protection for "e-commerce" patents is just as powerful in Europe as it is in the United States, and in some cases even stronger.[2]

Another question is whether it is really relevant to require a "technical character" in the first place. More than two decades of software patent case law have demonstrated that this criterion draws a rather arbitrary line. Many "technical" business method patents already exist today in Europe. Are we satisfied with such patents because simply they are "technical"?

In order to assess technicality, complicated tests have been devised, such as a requirement of "change in physical nature",[3] "planned use of controllable natural forces in order to achieve predictable result",[4] or similar notions from the realm of physics. Such rules tend to obscure the fact that in the end patent law is there to serve a purpose; it is a means to an end, but the means sometimes seem very remote from the end. The primary reason to stick to a requirement of "technical character" so rigidly appears to be that such a requirement has always existed, at least in Europe.

On the other hand, there is a widespread consensus that the domain of the European patent system should remain restricted in *some* way. However, attempts to define a criterion that could serve as an alternative requirement have so far been unsuccessful.[5] In Europe many still believe patents should be reserved, in one way or another, for *technical* inventions, whatever "technical" may mean. There is little sympathy for the approach in the United States, where all that is required is that an invention lead to a *concrete, tangible, and useful* result.

It has been suggested that the notion of "technical character" is *inherently* hard to define as patents deal with new technology by definition.[6] In our view however such a statement ignores the basic premises of the patent system. Patent law is economic law, and its virtues should be assessed essentially by its economic merits. If "technical character" is the decisive factor, its definition must be based on the role that technology plays in society, rather than on antiquated notions derived from physics or other exact sciences.

In the system of Art. 52 the European Patent Convention, business methods are listed among the categories of subject matter that are excluded "as such" (Subsections (2) and (3)). Perhaps it would be more appropriate to classify business methods in the same category as methods for medical treatment under Subsection (4), which is not subject to an "as such" limitation. This approach may reflect more properly the widespread desire to preserve certain patent-free domains.[7]

Specifically for software, it has been proposed to define categories that may or may not be patentable.[8] A problem to be solved here is how to keep such listings up to date, as technology is rapidly changing.

---

[1] Beresford 2000. See also the discussion of EPO TBA decisions in § 3.2.3.
[2] Lang 2000.
[3] Dutch Supreme Court, 20 January 1950, *NJ* 1950, 274 (*Rooilijnen*).
[4] German Supreme Court, 27 March 1969, GRUR 1969, 672 (*Rote Taube*).
[5] See for instance Dutch Ministry of Economic Affairs 2001, p. 38-43.
[6] See under the heading 'What is meant by "technical contribution"?' in: *Proposal for a Directive on the patentability of computer-implemented inventions - frequently asked questions,* <http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm>.
[7] Thomas 1999, p 47.
[8] See for instance Blind and others 2001, p. 225.

Intuitively, there is a desire in Europe to limit patents to "technical" inventions. In practice, this criterion has proven to lack selectivity. Also, it may not adequately reflect the economic rationales of the patent system. If certain patent-free domains are to be preserved, a categorical approach, e.g. excluding business methods unconditionally, is to be preferred.

6.2. Triviality and quality

As we have seen, a commonly heard objection against software and business method patents is that many of those patents have been granted to "trivial" inventions. Triviality in this sense usually implies that an invention lacks "inventive step", in other words it is "obvious to a person a skilled in the art".[1] Triviality might also mean that an invention is not novel at all.[2]

Often these problems are perceived merely as "quality problems", caused by patent offices not applying the rules properly. But there are other issues here; the rules may not be sufficiently clear, and the rules may even be inappropriate.

Testing for novelty requires comprehensive knowledge of the state of the art in the subject area. In new areas, such as software and in particular business methods, it is difficult and time consuming for examining offices to acquire such knowledge. Techniques are not always documented even if they are common knowledge – perhaps because they are common knowledge. Business methods in particular are rarely recorded in publicly available documents.

Apart from the problems of documenting prior art, applying the rule of novelty is relatively straightforward. Assessing whether or not an alleged invention is obvious, however, leaves more room for interpretation. The Examination Guidelines of the European Patent Office[3] leave no doubt that the test of non-obviousness will present an obstacle to patentability only in exceptional cases. For instance, a new but trivial combination of known techniques would be considered "obvious".

It has sometimes been argued that next to the novelty test a requirement of non-obviousness would be redundant. Novel inventions in areas of high research activity, such as computer software, would be non-obvious almost by definition; otherwise, they would have been invented before.

In theory non-obviousness is not enough; inventions must meet a certain standard of inventiveness. In German this is called "Erfindungshöhe" (literally: "inventive height"), in French "degré d'inventivité" ("degree of inventiveness"). These wordings indicate that the requirement of non-obviousness is not minimal; a substantial level of inventiveness is required.

The requirement of inventive step distinguishes patents from copyrights. Patent protection being much stronger and further-reaching, it is only logical to impose (considerably) stricter prerequisites. If novelty would suffice, patents would be available on conditions similar to the low-level originality requirement of copyright. In our view, this clearly demonstrates that in addition to novelty, a substantial level of non-obviousness must be required.

In several countries, "lightweight" patents are granted in the form of "utility models", requiring a "smaller" inventive step.[4] If the inventive step requirements for a regular patent would really be just minimal, there would be no place for such patents.

In practice however, a novel invention is rarely denied a patent, unless it is completely obvious to "a person skilled in the art". To be sure, this is by no means unique to software or business

[1] Art. 56 EPC (or 35 U.S.C § 103 (a)).

[2] Art. 52 EPC, 35 U.S.C. §102.

[3] *Guidelines for Examination In The European Patent Office* <http://www.european-patent-office.org/legal/gui_lines/pdf/gui_e_full.pdf>.

[4] An example is the German "Gebrauchsmusterrecht". Other European countries that currently have a utility model patent in one form or another include Belgium, Spain, Greece, and Portugal.

method patents. Unfortunately, courts may not be able to raise this standard, as it is firmly rooted in the tradition of the patent system. A major departure from current practice in this respect would be beyond the authority of the courts.

The "patent inflation" caused by granting "trivial" patents on a routine basis in our view calls for a fundamental debate, and eventually legislative action. Discussions concerning the proper domain of patent law have overshadowed the more mundane, but possibly more serious problem of trivial patents. In our view much of the controversy currently surrounding the patent system can be attributed to problems of "patent inflation" due to the routine granting of trivial patents. If patents are controversial because they do not live up to their promises, the main problem might well be patent "inflation", rather than the precise demarcation of the domain of patentable subject matter.

In sum, it would be too easy to conclude that the problem of trivial patents is simply an operational "quality" problem, to be addressed by administrative measures. The existing statutory requirement that a patentable invention not be "obvious to a person skilled in the art" merits serious reconsideration. Possibly, it should be replaced by a stricter requirement.

The fundamental question to be answered is how to define this requirement in such a way that the patent system is optimised for overall economic value. It will not be easy to find a proper criterion, also because we do not have a comprehensive insight in the way the patent system actually works.

Even so, preserving patent quality certainly is an important goal. The lesson from the U.S. experience is that sudden changes in patent law may confront patent offices with nearly insurmountable practical problems. It takes time to hire examiners skilled in new areas, and to build knowledge bases of "prior art" necessary for a proper assessment of the novelty of inventions in new areas. Apparently, the European Commission is aware of the risks of sudden changes in the patent system.[1]

In the United States a "Business Method Patent Improvement Act" has been proposed.[2] The proposal is entirely devoted to issues of patent quality. It provides interesting clues as to how trivial patents might be prevented. The solution offered is simply to lower the defendant's burden of proof of obviousness in infringement cases.[3] In conjunction with the existing European opposition procedure, this might be an interesting proposition.

Trivial patents represent a serious problem that is more than simply a matter of operational "quality". It is crucial to change the long-standing tradition of patents being granted for relatively simple inventions. Lowering the burden of proof of obviousness in case of infringement or opposition procedures might be part of a solution.

6.3. A European Patent Observatory

According to the proposed directive the Commission shall monitor the impact of computer-implemented inventions on innovation and competition. We would strongly recommend not just conducting such an evaluation on a periodical basis, but to go a step further by creating a "management system" that would keep track of the workings of the patent system on a routine basis. Statistics currently collected by the European Patent Office and other institutions such as WIPO and OECD, and the numerous consultations discussed in this report do not provide conclusive evidence on the merits of the patent system as it functions in practice. In our opinion,

---

[1] *Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions*, p. 11.

[2] *A bill to amend title 35, United States Code, to provide forimprovements in the quality of patents on certain inventions*, <http://www.uspto.gov/web/offices/com/doc/ipnii/lawpat.pdf>.

[3] Taffet & Hanish 2001.

collecting such data is essential for any informed policy decision concerning the ever-controversial patent system.
Therefore, we would recommend the establishment of a "European Patent Observatory", an agency dedicated to collecting all the data necessary to provide a solid empirical basis for future patent law making. [1]

Given the far-reaching effects of the patent system, we know surprisingly little about how it actually works. Therefore, we would recommend the establishment of a "European Patent Observatory" that would collect the empirical data needed to justify future changes of the patent system.

6.4. Other solutions

6.4.1. Protection of software inventions by special legal means

Over the years, several proposals have been tabled to introduce a *sui generis* right for the protection of computer software, as an alternative to patent protection.[2] The reason to create such a special right would be the special nature of computer program related inventions.[3] There is some merit in those ideas, as becomes apparent from the debate over software patents.[4] On the other hand, we have found that it is virtually impossible to draw a dividing line between software-related inventions and "regular" inventions. If the legal protection for software inventions would be weaker than for other inventions, history may repeat itself and patent attorneys may find ways to avoid their patents from being classified as "software patents".
The argument that computer program related inventions are "different" is actually just a repetition of the old debate that patents work out different in different industries.[5] Ideally, the goals of the patent system would be better met if it would be "tailored" in order to match the different needs of the various industries.[6] But it does not seem realistic to strive for such a differentiated patent system.
From a legal perspective, a major drawback of any sui generis regime is the absence of an existing international protection framework. There are provisions for patents and for copyright in the TRIPS agreement, but *sui generis* legal protection may not be covered by the agreement.

Even if good arguments exist for establishing a special legal regime for computer software, for practical reasons, as well as for the lack of an existing international framework, we would not recommend it.

---

[1] Other European Observatories existing today are: The Observatory of European SMEs, <http://europa.eu.int/comm/enterprise/enterprise_policy/analysis/observatory.htm>, The European Observatory on the Social Situation, Demography and Family, <http://europa.eu.int/comm/employment_social/family/observatory/who.html>, The European Industrial Relations Observatory (EIRO), <http://www.eiro.eurofound.ie>, The European Employment Observatory, <http://www.eu-employment-observatory.net/>, The European Observatory for Textile and Clothing, <http://www.oeth.com>, etc.
[2] Tauchert 1998, 1999.
[3] Lutterbeck, Gehring & Horns 2000.
[4] Before the time copyright became available for software, there has been a lengthy and fundamental debate about the appropriateness of copyright, patent or sui generis legal protection for software, e.g. within the framework of the *World Intellectual Property Organization* (WIPO).
[5] Granstrand 1999.
[6] Hart, Holmes & Reid 2000, p. 29.

6.4.2. Shorter protection term for patents

It is often said that in the software industry development cycles are shorter than in other industries. Therefore, it is sometimes questioned whether the normal maximum term of patent protection (twenty years from application) is appropriate for patents on computer program related inventions.

Again, the problem arises how to distinguish computer program related inventions from other inventions. Moreover, it must be acknowledged that such inventions should not be equated to e.g. pieces of programming code, or to software products. Some certain software-related inventions will remain relevant for several decades.[1] Conversely, many "regular" inventions will loose their market value well before the full patent term of twenty years.

In countries such as France and The Netherlands, low-threshold patent regimes exist providing for a shorter protection term (e.g. six years) under "easier" conditions. Patents of this type are registered without prior examination; for this reason they are known as "registration patents". The substantive validity of such patents can only be established by the courts in the course of an infringement suit. A somewhat comparable regime of "light-weight" patents is the "utility model" system that exists in certain European countries, which is also being considered at the European level.[2]

Existing short-term patent systems all depart from the objective, that patent application should be made easier for "simple" inventions. In both systems legal security is sacrificed for the purpose of providing a speedy and low-cost alternative to the patent system. Registration patents are available for inventions in all fields of technology. While software patent applicants may value an easy patent application procedure, because of its inherent lack of legal security a registration patent system does not appear to be a suitable model for software patent protection. Interestingly, it was not mentioned by any interested party during the French consultations.[3]

Shortening the term of protection might be appropriate for some software patents. However, not all software inventions are short-lived. Existing regimes providing short-term patent protection fail to provide legal security, and are therefore unsuitable for protecting software-related inventions.

6.4.3. Compulsory licences

Most patent laws contain provisions for compulsory licensing. If a patent owner is reluctant to grant a licence voluntarily, in exceptional cases a licence can be obtained in court or from an administrative body. Compulsory licenses can for instance be obtained if the patented invention is not used in any product of process (*non-usus*), or in the case of *dependency* of an invention on a pre-existing patent.

At first sight, compulsory licensing appears to be the appropriate remedy against "strategic" abuses of the patent system. "Blocking" patents might be licensed involuntarily by invoking a rule of *non-usus*. Compulsory license provisions, however, apply only under strict conditions; in practice, *non-usus* rules are rarely applied.

It has been suggested to interpret compulsory licensing provisions for reasons of dependency, which would apply only in case of "considerable economic interest",[4] to take into account "macro-economic interest" rather than just the "public interest".[1]

---

[1] We mentioned the Data Encryption Standard before. Another example is the widely used Lempel-Ziv data compression method which dates back to 1977 as well.

[2] *Commission presents a proposal for a Directive on the protection of technical inventions (utility models)*, <http://europa.eu.int/comm/internal_market/en/indprop/1127.htm>.

[3] French interdepartmental working group 2001, p. 12-13.

[4] As expressed e.g. by Art. 31(l)(i) of the TRIPS Agreement.

As a last resort, one might consider taking recourse to general rules of competition law. However, it must be emphasised that competition law can only serve as a "safety valve" under exceptional circumstances, not as a general cure to ailments of the patent system.[2]
To be sure, in practice voluntary patent licences are granted on a large scale, including non-exclusive licences. Patent auctions, "market places" and patent brokers have become a common phenomenon in the U.S.,[3] and one day the trade in patent licences might become just a common as the trade in shares.[4]

Compulsory licensing is generally perceived as an exception, to be invoked only under extraordinary circumstances. The widespread "strategic" use of the patent system might be a reason to revisit this assumption.

6.4.4. Measures to contain litigation costs

As software patents proliferate, the number of software patent lawsuits will inevitably increase. However, as we have seen, litigation may be prohibitively expensive, particularly for SMEs. As a solution to the problem of litigation costs, an insurance system or patent litigation fund has been proposed.[5]
Patent insurance is nothing new.[6] Commercial patent litigation insurance systems, however, have rarely been successful. Because of the risk of very high damages, premiums may become prohibitively high, or coverage is limited. An alternative might be the creation of a subsidised patent litigation fund. Patent applicants might be required to contribute to this fund on a compulsory basis.
Attractive as a patent litigation might seem to market players, it would of course not solve the fundamental problems of the patent systems, and probably only lead up to further patent proliferation.
Another suggestion is to introduce a form of Alternate Dispute Resolution (ADR), perhaps even on a compulsory basis.[7] The assumption underlying such an approach would be that patent disputes can be settled at lower cost and/or more quickly than by regular courts, yet still providing a similar or possibly even better level of legal security. This assumption may or may not be true, depending on how the courts operate in individual countries. Another argument in favour of ADR is specialisation, an argument less relevant for those countries that already have specialised patent courts.

The idea of the creating a (subsidised) patent litigation fund deserves further study. However, such a measure cannot replace more fundamental reforms of the patent system.

---

[1] Blind and others 2001, p. 231: "Das Kriterium des "öffentlichen Interesses" sollte dabei im Sinne des *gesamtwirtschaftlichen* Interesses verstanden werden." (The criterion of „public interest" should be understood in the sense of interest for the *economy as a whole*.)
[2] Hart, Holmes & Reid 2000, p. 36-38.
[3] See for example <http://www.patentauction.com>, <http://www.yet2.com> and <http://www.inventorsipo.com/patent_brokers.htm>.
[4] Rivette & Kline 2000.
[5] French interdepartmental working group 2001, p. 9, p. 20.
[6] Conference On Patent Insurance, April 25, 2000, Brussels, <http://europa.eu.int/comm/internal_market/en/indprop/litigation.htm>
[7] Kingston 2000, p. 154-158.

6.5. Towards a community patent

Since the proposed directive does not the European Patent Convention, it will not change the present situation that a European patent application does not lead to a Europe-wide patent, but merely to a "bundle" of national patents. Patent application will therefore remain much more expensive in Europe than in the U.S. A Community Patent, a single patent for the entire European Community, would solve this problem.[1] Also there is a long-standing desire to create a single European patent court.[2]
Given these forthcoming, more fundamental changes of the European patent system, the proposed directive would constitute at best a short-term measure.

> The proposed directive represents only a modest step towards unification of the European patent system.

7. Summary and conclusions

Should software and "business method" patents be allowed in Europe to the same extent as in the United States? Can we expect a favourable effect on the European economy if the patent system in Europe would be liberalised in a manner similar to the U.S. system? In our study we have found that despite extensive research there are no conclusive answers to these questions. Economists have examined the economics of the patent system in general for many years, but have never been able to demonstrate conclusively that the advantages of a patent regime outweigh its drawbacks. Even the basic assumption that patents promote innovation to such an extent that the costs are justified and the disadvantages are offset by the advantages, is questionable; according to some scholars, the opposite might be true as well.
Conclusive evidence supporting a liberalisation of existing European patent law and practice in respect of software and business method patents, on the basis of the U.S. experience, does not exist. The liberalisation of U.S. patent law as a result of the 1998 *State Street* decision, that has opened up the domain of patents to all sorts of business method claims, does appear to have lead to considerable administrative problems. But it would be too early to draw conclusions with regard to longer term economic effects.
While some claim that software is something radically new, the current debate over software patentability is actually as old as the patent system itself. Of course, there are R&D intensive industries that could hardly function without patents, such as the pharmaceuticals industry. However, there are many other sectors of industry where the contribution of patents is questionable. Unless this fundamental lack of knowledge is addressed in a more structured manner, any proposal to optimise the patent system in respect of software-related inventions is based on nothing more than wild guesses or wishful thinking.
Several previous studies have tried to find answers by means of consultation exercises. These studies show a wide diversity of opinions from a wide variety of stakeholders. The constant factor seems to be the observation that "software", the software industry and its players are very diverse, and their opinions are very divergent. Relatively few consulted parties seem to be in favour of liberalising the patent regime currently existing in Europe for software patents, and "business method" patents are generally considered to be undesirable. SMEs apparently do not like patents at all, despite extensive promotion efforts by proponents of the patent system.

---

[1] *The Community Patent: Objectives of the Commission and present situation,* speech by Commissioner Frits Bolkestein, member of the European Commission in charge of the Internal Market and Taxation, Liège, 29th November 2001, <http://europa.eu.int/comm/internal_market/en/speeches/01-597.htm>
The Community Patent - Frequently Asked Questions <http://europa.eu.int/comm/internal_market/en/indprop/2k-41.htm>
[2] Brinkhof 2000, p. 600-604.

Developers of Open Source software seem to be particularly vulnerable to a proliferation of software patents.

Given these uncertainties, it would be more appropriate to concentrate on legal security and harmonisation of law than to contemplate major substantive changes in patent law. This is indeed the route chosen by the European Commission in the proposed Software patent directive, even though the explanatory memorandum accompanying the proposal suggests that software patents are good for the European economy.

The solution proposed by the Commission is to re-emphasise the requirement of *technical* contribution, which is considered fundamental to existing European patent law. In our view, however, it is questionable whether codifying this requirement will indeed improve legal security. Experience has shown that the notion of "technical" character (or contribution or effect) can have several meanings, and that patents in essentially non-technical subject matter can be drafted in such a way as to circumvent this prerequisite.

More fundamentally, it is questionable whether a requirement of technical contribution actually constitutes a relevant criterion. As the Commission admits in the explanatory memorandum, this test does not prevent (all) business methods from being patented. Indeed, the proposed directive does not categorically exclude business method patents. In view of the widespread objections against such patents, it might be advisable to move business methods from the list of subject matter excluded "as such" to the category of subject matter that is excluded "per se", similar to e.g. medical treatments.

The proposed directive fails to address other urgent issues, in particular the widely perceived problem of "trivial" patents. Many of the problems the patent system is currently experiencing can be attributed to dubious patent quality. Patents are routinely and all too easily granted with regard to inventions that display little or no inventive character. Arguably, many of the fears expressed by those opposing software patents, particularly by proponents of Open Source software, might be accommodated by raising or reviving the standard of inventive step. It would, however, be false to conclude that patent quality is simply an operational problem. Patent inflation is a fundamental problem that should be addressed by legislative action.

The proposed directive envisages an evaluation exercise some time after the directive has come into force. But ideally, the patent system should be made transparent, not just by means of occasional consultations and evaluations, but by integrating the gathering of "management" type data into the routine of the patent process. Statistics collected as a by-product of the mainstream administrative process apparently cannot satisfy the needs of decision makers. Rather, data should be collected specifically for that purpose, as an independent activity in its own right. Patents have a powerful potential to interfere with free competition in the market place, and in our view, such interference would justify an approach where the functioning of the patent system and its effects on the economy are monitored on a routine basis.

Therefore, we recommend the establishment of a "European Patent Observatory", an agency to be entrusted with the task of monitoring the way the patent system functions in Europe and of collecting all the data necessary to decide on future reforms of the European patent system.

- Central to the proposed directive is its emphasis on a requirement of technical contribution. However, the proposal fails to elaborate on the interpretation of the word "technical", particularly in the light of existing case law.
- Patent inflation (i.e. the routine granting of patents in "trivial" inventions) is the source of many problems in the patent system, particularly with regard to software-related patents. It

is not simply a quality problem, but a fundamental problem that urgently requires a legislative solution. A first step towards such a solution might be to add a recital to the proposed directive requiring a proper application of the inventive step test in respect of software-related inventions.

- The proposed directive does not categorically exclude business method patents. In view of the widespread objections against such patents, it might be advisable to move business methods from the list of subject matter excluded "as such" to the category of subject matter that is excluded "per se".

- Conclusive evidence demonstrating the positive or negative effects of software patenting on the economy does not exist. Future patent law reform in Europe should rely on facts rather than opinions. Therefore, we recommend the establishment of a European Patent Observatory, to be entrusted with the task of monitoring the way the patent system functions in Europe and of collecting all the data necessary to decide on future reforms of the European patent system.

8. References

8.1. Previous studies

The principal previous studies are listed below: first the EU-wide studies, next the other studies in alphabetic sequence of the member state names.

Hart, Holmes & Reid 2000

Robert Hart, Peter Holmes and John Reid, *The Economic Impact of Patentability of Computer Programs*, October 2000, <http://europa.eu.int/comm/internal_market/en/indprop/study.pdf>.

PbT Consultants 2001

The Results Of The European Commission Consultation Exercise On The Patentability Of Computer Implemented Inventions, July 2001, <http://europa.eu.int/comm/internal_market/en/indprop/comp/softanalyse.pdf>.

French interdepartmental working group 2001

Rapport du groupe de travail interministériel "Quelles protections pour les logiciels?" (Report of the interdepartmental working group "Which protections for computer programs") , July 2001, <http://www.industrie.gouv.fr/observat/innovat/carrefour/rapgrtravail.pdf>. There is no English translation to our knowledge.

Blind and others 2001

Knut Blind, Jakob Edler, Ralph Nack & Joseph Straus, *Mikro- und macroökonomische Implikationen der Patentierbarkeit von Softwareinnovationen: Geistige Eigentumsrechte in der Informationstechnologie im Spannungsfeld von Wettbewerb und Innovation*, November 2001, <http://www.bmwi.de/Homepage/download/technologie/Softwarepatentstudie.pdf>. This document includes a summary in English (which is also separately available from the Internet): *Micro- and Macroeconomic Implications of the Patentability of Software Innovations. Intellectual Property Rights in Information Technologies between Competition and Innovation*, November 2001. <http://www.bmwi.de/HomePage/download/technologie/Softwarepatentstudie_E.pdf>

Lutterbeck, Gehring & Horns 2000

Bernd Lutterbeck, Robert Gehring, Axel H. Horns, *Sicherheit in der Informationstechnologie und Patentschutz für Software-Produkte - Ein Widerspruch?*, report commisioned by the German federal Ministry of Economics and Technology, December 2000, <www.sicherheit-im-internet.de/download/BMWi_Gutachten_englisch.pdf>.
Security in Information Technology and Patent Protection for Software Products, a Contradiction? Unofficial transation by the German federal Ministry of Economics and Technology, <http://www.sicherheit-im-internet.de/download/BMWi_Gutachten_englisch.pdf>.

Dutch Ministry of Economic Affairs 2001
De juridische en economische aspecten van het software-octrooi (Legal and economic aspects of the software patent), Ministry of Economic Affairs, December 2001, <http://www.minez.nl/publicaties/pdfs/01I21.pdf>. There is no English translation.

Verkade, Visser & Bruining 2000
D.W.F. Verkade, D.J.G. Visser & L.D. Bruining, *Ruimere octrooiëring van computerprogramma's: technicality of revolutie?* (Broadened patenting of computer programs: technicality or revolution?), Ite*R* series nr. 37, Den Haag: Sdu Uitgevers 2000. [ISBN 90-540-9267-X]. An early version is available on <http://www.nwo.nl/iter/ovc.pdf>.

UK Government 2001
Should Patents be granted for Computer Software or Ways of Doing Business? March 2001, <http://www.patent.gov.uk/about/consultations/conclusions.htm>.

8.2. Other references

Basinski 2002
Erwin J. Basinski, 'European Union's proposed Directive on computer-implemented inventions', *World E-commerce & IP Report* - Vol. 2 (2002) 3 (Mar.), p. 3-5.

Beresford 2000
Keith Beresford, *Patenting Software under the European Patent Convention*, London: Sweet & Maxwell 2000.

Beresford 2001
Keith Beresford, 'European patents for software, E-commerce and business model inventions', *World Patent Information* - Vol. 23 (2001) 3 (Sep.), p. 253-263.

Bessen & Maskin 2000
James Bessen & Eric Maskin, *Sequential Innovation, Patents and Imitation*, No. 00-01, Massachusetts Institute of Technology, January 2000, <www.researchoninnovation.org/patent.pdf>.

Brinkhof 2000
Jan J. Brinkhof, 'Die Schlichtung Patentstreitigkeiten in Europa. Über Traum und Wirklichkeit', *GRUR* 2001, p. 600-604.

Fellas 1999
John Fellas, 'The Patentability of Software-related Inventions in the United States', *EIPR* 1999, p 330-333.

Fujimura 2001
Motohiko Fujimura, 'Recent revision of JPO's examination guideline for computer software-related inventions', AIPPI: *Journal of the Japanese Group of AIPPI* - Vol. 26 (2001) 4 (July), p. 210-212.

Gervais 1998
Daniel Gervais, *The TRIPS agreement: drafting history and analysis*. London: Sweet & Maxwell, 1998.

Granstrand 1999

Ove Granstrand, *The Economics and Management of Intellectual Property. Towards Intellectual Capitalism*, Cheltenham, UK/Northampton MA, USA: Edward Elgar 1999.

Jaffe 1999

Adam B. Jaffe, *The U.S. Patent system in transition: policy innovation and the innovation process*, Working Paper 7820, National Bureau of Economic Research, Cambridge MA.

Kingston 2000

William Kingston, 'The Case for Compulsory Arbitration: Empirical Evidence', *E.I.P.R.* 2000, p. 154-158.

Lang 2000

Johannes Lang, 'Europe grants e-commerce patents too', *Managing Intellectual Property* - (2000) 97 (March), p. 13-15.

Machlup 1958

F. Machlup, An Economic Review of the Patent System, Study no. 15 of the Subcommittee on Patents, Trademarks and Copyrights of the Committee on the judiciary, United States Senate, 85th congress, second session, 1958, p. 80.

Maier & Mattson 2001

Gregory J. Maier & Robert C. Mattson, '*State Street Bank* ist kein Ausreißer: Die Geschichte der Softwarepatentierung im US-amerikanischen Recht', *GRUR Int.* 2001, p. 677-690.

Merges 1999

Robert Merges, 'As Many As Six Impossible Patents Before Breakfast: Property Rights For Business Concepts And Patent System Reform', *Berkeley Technology Law Journal*, Vol. 14, 1999, p. 577-615. Also available on <http://www.law.berkeley.edu/institutes/bclt/pubs/merges/siximp.pdf>.

Parker 2001

Tabitha Parker, 'Internet patents are here to stay', *Managing Intellectual Property* - (2001) 107 (March) p. 12-16.

Peterman 1999

Chad J. Peterman, 'Reforming the system: the American Inventors Protection Act of 1999', *Patent World* - (1999/2000) 118 (Dec./Jan.), p. 14-15.

Rivette & Kline 2000

Kevin G. Rivette & David Kline, *Rembrandts in the Attic: Unlocking the Hidden Value of Patents*, Boston(MA): Harvard Business School Press 2000.

Schiuma 2000

Daniele Schiuma, 'TRIPS and Exclusion of Software "as Such" from Patentability', *IIC* - Vol. 31 (2000) 1, p. 36-51.

Shapiro & Varian 1999

Carl Shapiro & Hal R. Varian, Information Rules, *A Strategic Guide to the Network Economy*, Boston Massachusetts: Harvard Business School Press 1999.

Sterne & Lee 2002

Robert Greene Sterne & Michael Q. Lee, How to Take Advantage of Business Method Patents, World e-Commerce & IP Report March 2002, p. 7-11.

Straus 1996

Joseph Straus, 'Bedeutung des TRIPS für das Patentrecht', *GRUR Int.* 1996, p. 179-205.

Taffet & Hanish 2001

Richard S. Taffet and Marc S. Hanish 1, *The Business Method Patent: The Uproar Rages - Should It?* <http://www.thelenreid.com/articles/article/art91.htm>. This article

originally appeared in Vol. X-2, 2001 of International Legal Strategy: Legal Risk Management for Japan Business Executives.

Tang, Adams & Paré 2001

Puay Tang, John Adams & Daniel Paré, *Patent protection of computer programmes*, <http://europa.eu.int/comm/internal_market/en/indprop/sofstudy.pdf>.

Tauchert 1998

Wolfgang Tauchert, 'Examination of software-related applications at the German patent office', *Patent World* - (1998) 103 (July), p. 33-37.

Tauchert 1999

Wolfgang Tauchert, 'Zur Patentierbarkeit von Programmen für Datenverarbeitungsanlagen', *Mitteilungen der deutschen Patentanwälte* - Vol. 90 (1999) 7 (July), p. 248-252.

Wolfgang Tauchert, 'Patentschutz für Computerprogramme - Sachstand und neue Entwicklungen', *GRUR* - Vol. 101 (1999) 10 (Oct.), p. 829-833.

The contents of both publications are nearly identical.

An English translation is available. See below.

Tauchert 2000

Wolfgang Tauchert, 'Patent protection for computer programs - current status and new developments', *IIC* - Vol. 31 (2000) 7/8 (Nov.), p. 812-824.

English translation of Tauchert 1999.

Thomas 1999

John R. Thomas, The Post-Industrial Patent System, *Fordham Intellectual Property, Media & Entertainment Law Journal*, 1999, p. 3-59.

Toren 2000

Peter Toren, 'Software and business methods are patentable in the US (get over it)', *Patent World* - (2000) 125 (Sep.), p. 7-9.

Ward 2002

Peter Ward, 'The European Commission's proposal for a Directive on the patentability of Computer-implemented inventions: clarity or confusion?', *World E-commerce & IP Report* - Vol. 2 (2002) 3 (Mar.), p. 5-7.

Zoltick 2000

Martin M. Zoltick, 'Like it or not - E-patents are for real!', *World Licensing Law Report* - (2000) 7 (July), p. 20-22.

New .eu Domain

Changed Web and E-Mail Addresses

The introduction of the .eu domain also required the web and e-mail addresses of the European institutions to be adapted. Below please find a list of addresses found in the document at hand which have been changed after the document was created. The list shows the old and new address, a reference to the page where the address was found and the type of address: http: and https: for web addresses, mailto: for e-mail addresses etc.

Page: 2 **Old:** mailto:koffermann@europarl.eu.int
Type: *mailto* **New:** mailto:koffermann@europarl.europa.eu

Page: 9 **Old:** http://europa.eu.int/eur-lex/en/lif/dat/1991/en_391L0250.html
Type: *http:* **New:** http://europa.eu/eur-lex/en/lif/dat/1991/en_391L0250.html

---

Page: 30 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/com02-92en.pdf

---

Page: 31 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/02-32.htm

---

Page: 31 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/com02-92en.pdf

---

Page: 32 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/com02-92en.pdf

---

Page: 32 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/02-32.htm

---

Page: 33 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/com02-92en.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/com02-92en.pdf

---

Page: 34 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/02-32.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/02-32.htm

---

Page: 37 **Old:** http://europa.eu.int/comm/enterprise/enterprise_policy/analysis/observatory.htm
Type: *http:* **New:** http://europa.eu/comm/enterprise/enterprise_policy/analysis/observatory.htm

---

Page: 37 **Old:** http://europa.eu.int/comm/employment_social/family/observatory/who.html
Type: *http:* **New:** http://europa.eu/comm/employment_social/family/observatory/who.html

---

Page: 38 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/1127.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/1127.htm

---

Page: 39 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/litigation.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/litigation.htm

---

Page: 40 **Old:** http://europa.eu.int/comm/internal_market/en/speeches/01-597.htm
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/speeches/01-597.htm

---

Page: 40 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/2k-
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/2k-

---

Page: 42 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/study.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/study.pdf

---

Page: 42 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/comp/softanalyse.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/comp/softanalyse.pdf

---

Page: 45 **Old:** http://europa.eu.int/comm/internal_market/en/indprop/sofstudy.pdf
Type: *http:* **New:** http://europa.eu/comm/internal_market/en/indprop/sofstudy.pdf

---