

# Problems with Software?

by Axel Pfeiffer<sup>1</sup>

(A very similar – but not identical – text was published in the June 2004 edition of  
5 IFE, Springer)

## Contents:

**1. History, Status, Further Procedure**

**2. "Arguments", Paradigms, Views**

10 **3. Tactics**

**4. Interests**

**5. To do**

End of September 2003 the European Parliament commented on the proposal of the  
15 Commission regarding a directive for the patentability of computer-implemented  
inventions in a way which is - according to the Author's opinion - very question-  
able<sup>2</sup>. It does not seem as if the discussion on the question of the patentability of  
such inventions will calm down. Some observations as to the protagonists, interests,  
"arguments" and occurrences in the last few years are compiled below. Considera-  
20 tions relating to patent law can hardly be found in this text. In particular the Au-  
thor's opinion regarding legal questions is not described here, since it would be a  
repetition of what had already been discussed elsewhere<sup>3</sup>.

Brief annotation as to the usage: The term "software" in this text is used as a  
25 synonym for the term "program for data processing equipment ... as such" as it is

---

<sup>1</sup> Dipl.-Ing., Patent Attorney, Beetz & Partner, Munich

<sup>2</sup> <http://www3.europarl.eu.int/omk/omnsapir.so/pv2?LISTING=AfficheTout&PRG=CALDOC&FILE=20030924&LANGUE=DE&TPV=PROV#Title3>

<sup>3</sup> GRUR 2003, page 581, also available under [www.beetz.com](http://www.beetz.com)

used in the patent law. The Author understands this term as a "series of instructions for the understanding of a machine".

## 1. History, Status, Further Procedure

5

### 1.1 Prior to 1973

Until and through 1972 software in the field of industrial property has been a legal  
 nought in the sense that written law has not dealt with it. Neither in the copyright  
 10 law nor in the patent law software has expressly been mentioned. At this point of  
 time this wasn't surprising, since then the issue software was seen in considerably  
 different contexts than it is today. On the time line: end of 1972 the Author of this  
 text being a 10 years old schoolboy and freshman at secondary school possessed, as  
 the top novelty, a 10x5x2 cm pocket calculator with LED display, the most sophis-  
 15 ticated function of which was the percentage button. At that time, of course, hard-  
 ware was regarded as the everlasting bottleneck of digital technology, and software  
 was conceived as something that, being "soft" instead of "hard", could - if necessary  
 - be easily and quickly provided, if hardware existed on which it could run.

### 20 1.2 1973: European Patent Convention

At the beginning of the 70ies the European Patent Convention (EPC) was worked  
 out and adopted in 1973. Already then Art. 52 EPC had today's wording.<sup>4</sup>.

---

<sup>4</sup> **EPÜ Artikel 52:** Patentable Inventions

*(1) European patents shall be granted for inventions as far as they are new, involve an inventive step and are susceptible of industrial application.*

*(2) The following in particular shall not be regarded as inventions within the meaning of paragraph 1:  
 a) discoveries, scientific theories and mathematical methods; b) aesthetic creations; c) schemes, rules and  
 methods for performing mental acts, playing games or doing business, and programs for computers;  
 d) presentations of information.*

*(3) The provisions of paragraph 2 shall exclude patentability of the subject-matter or activities referred to  
 in that provision only to the extent to which a European patent application or European patent relates to  
 such subject-matter or activities as such.*

*(4) Methods for treatment ... by surgery or therapy ...*

From today's point of view (2003) this regulation is meanwhile 30 years old. It was the first one in which software has ever been mentioned in the industrial property law.

### 5 1.3 1978: Starting Business, Harmonization of Legislation

In 1978 the European Patent Office (EPO) factually started business and since then has been granting patents according to the criteria of the EPC. In the same year in Germany substantive rules of the Patent Act (PatG) were brought in line with the  
10 corresponding ones of the European Patent Convention, in particular to that effect that the articles in the Patent Act relating to patentability practically became corresponding articles of the EPC, having the same wording.

### 1.4 Since that time

15

In spite of a not at all simple nor direct development one can say that at the European Patent Office and the German Patent and Trademark Office the granting practice regarding software-implementable inventions can be described as decreasingly restrictive over the years. This course is flanked by various decisions of the responsible authorities (Boards of Appeal of the European Patent Office, Federal Patent  
20 Court, Federal Supreme Court) and is reflected therein. Today's handling of software exclusion of Art. 52 EPC corresponds to the view that it is directed against software as an invention, not, however, against software as a means of implementing an invention or against software as being comprised by the scope of protection  
25 of a patent.

### 1.5 1993: Copyright law

Only in 1993 the copyright law was clearly formulated regarding the protection of software in its present wording. In short, the copyright law prohibits the unauthorized reproduction of software, but not the accidental identical new programming or the different programming of a certain sequence of operations, which have already  
 5 been programmed in a certain way before. The copyright law does not provide protection of concepts. § 69g of the copyright law lays down that the provisions of the copyright law regarding software do not affect the application of the patent law regarding software..

## 10 1.6 1995: TRIPS

In 1995 the WTO concluded the Agreement TRIPS. Art. 27<sup>5</sup>, particularly clause 2, obliges the contractual states to take care that the protection provided by a patent extends to all inventions which are of technical nature and which are based on an  
 15 inventive activity. Clause 3 allows exceptions regarding patentability. Software, however, is not mentioned therein..

## 1.7 Open Source, Linux

---

<sup>5</sup> **TRIPS: Article 27** Patentable Subject Matter

*1. Subject to the provisions of paragraphs 2 and 3, 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.5 Subject to paragraph 4 of Article 65, paragraph 8 of Article 70 and paragraph 3 of this Article, patents shall be available and patent rights enjoyable without discrimination as to the place of invention, the field of technology and whether products are imported or locally produced.*

*2. Members may exclude from patentability inventions, the prevention within their territory of the commercial exploitation of which is necessary to protect ordre public or morality, including to protect human, animal or plant life or health or to avoid serious prejudice to the environment, provided that such exclusion is not made merely because the exploitation is prohibited by their law.*

*3. Members may also exclude from patentability:*

*(a) diagnostic, therapeutic and surgical methods for the treatment of humans or animals;*

*(b) plants and animals other than micro-organisms, and essentially biological processes for the production of plants or animals other than non-biological and microbiological processes. However, Members shall provide for the protection of plant varieties either by patents or by an effective sui generis system or by any combination thereof. The provisions of this subparagraph shall be reviewed four years after the date of entry into force of the WTO Agreement.*

Probably since the mid-90ies in the field of creating software a business model has established itself which is known under the name "Open Source". The most famous Open-Source product might be the operating system Linux which is highly welcome as a rival offer with regard to the ubiquitous operating system Windows and which is widely well-liked – also by the Author of this text.

Programming in the case of Open Source is based on the public availability of the source code for programs and program clusters, so that all users, who are prepared to and capable thereof, can supplement, update, change and amend the source code. In order to make possible - instead of an uncontrollable proliferation originating from a once created unit - a purposeful development, there are authorizing structures which check, consider, test and select the various proposals for supplementing or amending the source code and which, from time to time, declare and determine new or further developed versions of a program as the current version. Starting from this version distinguished by declaration, further operations can be performed.

The further developed source code is always available free of charge. Private users and companies can adopt it, use it and modify it for their own purposes. Business interests of companies are satisfied at least by customizing the source code and by offering services around the source code.

### 1.9 As of end of the 90ies: opposition

As of 1998 an organized opposition against the patenting of software-implementable inventions can be noticed. This opposition originates not exclusively, but at least partially - and according to the Author's opinion exclusively - from Open Source circles. Instead of making many words, by means of a footnote<sup>6</sup> reference is simply made to some internet addresses. At least parts of these struc-

tures and organizations might today no longer be regarded as mere interest groups, but also as highly efficient and experienced lobbyists, particularly in Europe and, most particularly, in Germany.

5 1.10 Approximately as of 2000: Efforts of the European Patent Organization (EPO)  
 In view of the time-and-again arising question of patentability of software-  
 implementable inventions, in the year 2000 the EPO being the administrator of the  
 EPC and of the European Patent Office started an attempt to modernize Art. 52 EPC  
 passed in 1973 in such a way that, for avoiding misunderstandings, the exclusion of  
 10 software, which misses the point from a factual point of view, should be cancelled.  
 However, at the formal meeting of the Governmental Conference forming the EPO  
 in Munich in fall 2000 this proposal was rejected. Thus, the revision failed and the  
 existing regulations were maintained.

15 1.11 Approximately as of 2000: Efforts of the EU  
 After the efforts of the EPO, being independent of the EU, to update the regulations  
 regarding software had failed, the European Commission took increasingly care of  
 this matter. They had, however, already dealt therewith before. In spring 2002 a  
 draft of an EU directive was tabled which got mixed reviews. Most of the people  
 20 professionally dealing with the patent law took a substantially favorable view of the  
 draft. The opponents of patents on software-implementable inventions, however,  
 were up in arms against it. One already tried to influence the drafting of the direc-  
 tive by the Commission, however with little success. After the draft had been pub-  
 lished at the beginning of 2002 and it became foreseeable that the European Parlia-  
 25 ment had to deal with it, one started to work on the respective national representa-  
 tives. There were e-mail campaigns, demonstrations, discussion groups, press inter-  
 views, petitions, web-strikes and the like, which helped this interest group to get an  
 astonishing echoe in the papers and to be heard by the members of the European

---

<sup>6</sup> [www.eurolinux.org](http://www.eurolinux.org), cf. there particularly the list of members, [www.ffii.org](http://www.ffii.org), [www.aful.org](http://www.aful.org),

Parliament. Statements of an imminent patenting of jazz rhythms even made it into the daily news.

The preliminary end point of this development was the treatment of the draft directive in the European Parliament end of September 2003. The result, however, might universally be regarded as poor from a material point of view as well as from the aspect of skills. The desired amendments, cancellations and supplementations formulated by the Parliament are numerous, a few of them being cited in the footnote<sup>7</sup>. The compilation which can be found there may look strangely incoherent. However, the impression will not get better, if the document is regarded as a whole. "Everybody can have a go" could have been the motto, and the result of the considerations looks like it. Nobody - irrespective of what ideology - can wish that something like that will ever enter into force.

## 1.12 Status today: legal principles

According to today's status the presently valid legal principles are illustrated:

The EPC and the German Patent Act define in the same way software as not being a patentable invention. This is an exclusion which substantially misses the point,

---

<http://lpf.ai.mit.edu/Patents/>, a little further off [www.fsf.org](http://www.fsf.org), [www.bustpatents.com](http://www.bustpatents.com) and others.

<sup>7</sup> **Amendment 8:** *Furthermore, an algorithm is inherently non-technical and therefore cannot constitute a technical invention.*

**Amendment 45:** *Member States shall ensure that data processing is not considered to be a field of technology within the meaning of patent law, and that innovations in the field of data processing are not considered to be inventions within the meaning of patent law.*

**Amendment 16 and others:** *Member States shall ensure that a computer-implemented invention making a technical contribution constitutes a necessary condition of involving an inventive step. The significant extent of the technical contribution shall be assessed by consideration ...*

**Amendment 103 and others:** *Member States shall ensure that the production, handling, processing, distribution and publication of information, in whatever form, can never constitute direct or indirect infringement of a patent, ...*

**Amendment 104 and others:** *Member States shall ensure that whenever a patent claim names features that imply the use of a computer program, a well-functioning and well documented reference implementation of such a program shall be published as a part of description without any restricting licensing terms.*

**Amendment 107:** *'Technical contribution', also called 'invention' means a contribution to the state of the art in a technical field. ...*

since, as a rule, in the case of the inventions in question software is just not the invention. Software is only used to "build", i.e., to program the invention.

5 The copyright protects written software from being reproduced. However, a protection of concepts is expressly not provided.

TRIPS obliges the contractual states to take care of the protection of all technical inventions by patents, an exception for software is not made.

### 10 1.13 Further procedure

Probably neither the EPO nor national governments will soon make attempts to make any amendments.

15 On EU level the directive will be further processed, what can also include considering the matter closed. A weakening of the lobbyist efforts of the opponents of the directive is not to be expected, rather on the contrary: the hitherto gained success, that on the EPO level the revision of the EPC had been prevented as well as that on the EU level an enormous influence on the course of things had been gained via the  
20 Parliament, lets one assume that this rather has an inspiring and encouraging effect.

## **2. "Arguments", Paradigms, Views**

25 The economic justification of patents as temporarily limited prohibitive rights for the protection of virtual/actual innovations resides in the fact that before the start of the development work patents provide the certainty that actual inventions will remain within the sphere of the developing company for a limited period of time, so that, beforehand, an incentive for the development is given, and that after the successful termination of the development work there exists the justified hope for am-

ortisation and also over-amortisation of the development costs, what can be regarded as the return for the development.

Subsequently those arguments, paradigms, views and insinuations are to be explained which are used by the opponents of patents on software-implementable inventions for supporting their requests and wishes. The fact that the "arguments" are mentioned here does not mean that the Author attributes soundness or legitimacy to them. They are mentioned here to show how it is argued, so that one can judge in which way and on which level influence is exerted and what must be expected in the future.

2.1 "Patents do not fit for software, since software is a special case, because ..."

well because ... in Europe 400,000 people are concerned/deal therewith, ... software is the reflection of human minds, ... software and language and thinking and creativity are the same, ... software had become a cultural asset, ... software has many interfaces, ... software has many standards which must not be patented, ... software-producing companies is grass root economy of small companies, ... otherwise programmers would have to make patent searches, ... "softwarepatents" can lead to over-amortisation, ... patents on software would constantly block the only possible solution, ... software in the internet can very easily be distributed and has a completely different structure regarding sale and costs, ... software is very exceptionally innovative, ... software allows the Open-Source business model, etc..

In many of these statements the word "software" can easily be replaced by the term "hardware" or "iron", and these statements will remain just as correct or half-correct as they have been for the term "software". This is a sign for the fact that the statement per se may be (half)correct, but that no software peculiarity is concerned. And further, the paradigm assumed as a matter of course, namely "different things must

be treated differently" is denied: A bicycle is certainly different from a truck. For good reasons, however, a traffic light is the law for both in the same way.

If possible over-amortisation is addressed (i.e. based on the protection provided by a patent the patentee receives more than he had to invest in his invention) and this is characterized as reprehensible, one only has to briefly say that it is an aim of any investment and generally of any economic activity to gain more than was put into it. Not a zero-sum game or losing out are stimulating, but the expectation of a gain. If this aim is defamed the generation of added value altogether is questioned.

10

As far as the easy distributability in the internet is concerned, then, from a legal point of view, the actual simplicity of infringing a law is used as a justification for abolishing this law. Imagine what would analogously happen if the association of surgeons would request to indemnify surgeons from the liability for medical malpractice, because incidental malpractice is likely to occur.

15

The unmatched innovative power of software is asserted as a matter of course, but this sweeping statement is nevertheless questioned here. Hardware, too, is extremely innovative, what can be told by everybody from the fact that he/she has to buy not only new software at least every four or five years, but also new hardware. Apart therefrom, software is not necessarily always fast and innovative. The classic example for non-innovative software was the Y2K problem at the end of 1999 caused by still running outdated software which could not handle four-digit year specifications. Another example: For more than one decade a known operating system had at last been using at least internally constant formats, e.g. the 8.3 file name format, although it had been obsolete for a long time.

20

25

And apart from the above statements the references to alleged software peculiarities neglect the fact that software regularly is not the invention. In none of the so called

“software patents” is software given. Software is only needed for "building up" the invention. Many of the software-implementable inventions could have been made by inventors having no knowledge of software. Hence, references to software characteristics of inventions miss the point since never ever software is the invention.

5

## 2.2 "Copyright is sufficient"

... and therefore the protection of software products by a patent would not be needed. If software alone is regarded, one can certainly agree to the above statement. However, as it is the case, the patent law is to protect inventions and the considered inventions are - as already said above - regularly not software. For avoiding repetitions reference is made to the end of the previous item.

10

## 2.3 "Software is non-technical"

15

Right from the start one can assume that this statement is just as correct as the statement "all Irishmen have red hair". May be that software-implementable inventions more strongly correlate with non-technical subject-matter than do metal-implementable inventions. This, however, does not alter the fact that the question of technical nature on the one hand, and the question of software on the other hand are independent of each other, and are to be answered separately, if one concrete invention is regarded. And again the same holds true as at the end of items 2.1 and 2.2: The question of technical nature cannot be judged when a piece of software is regarded, but when regarding an invention which regularly lies beyond the software to be implemented.

20

25

## 2.4 "30.000 illegal swpat" (swpat = "software patent")

Somehow the figure of 30.000 patents on software-implementable inventions hitherto granted by the EPO turned up. From the fact that software can infringe a patent one deduces that, with regard to the software exclusion in Art. 52 EPC, these patents are illegal. Of course, this arguments is based on the assumption that Art. 52 refers to the scope of protection of a patent, although the scope is expressly not defined in Art. 52 EPC, but in Art. 64 EPC<sup>8</sup> (reference) or. §§ 9 and following of the Patent Act. Nevertheless a prima facie contradiction is used for protraying the official practice of EPO and GPTO as contra legem.

10 According to the Author's opinion this "argument" is to a high degree impressive and substantially contributes to the scepticism and loss of confidence vis à vis the way of handling and the protagonists.

## 2.5 Stirring fears

15

This sounds like that: "Mozart would have had to become a café violinist, if Haydn had got a patent on his music/compositions". "Enterprises dealing with software are forced into an arms race/hard competition, soon we will have a situation as in the USA". "Cf. patenting-art.com". The injunction procedure of amazon.com against Barnes & Noble during Christmas time of the year 2000 (?) based on the one-click patent was just what one was waiting for in this connection. Full of relish one keeps account how in the USA enterprises sue each other. These aspects are put forward with routine and reliability, stated and highlighted again and again, and despite the obvious absurdity this will work to a certain degree according to the strategy

25 "aliquid simpliciter haeret".

---

<sup>8</sup> **EPÜ Art. 64:** Rights conferred by a European patent

(1) *A European patent shall confer on its proprietor from the date of publication of the mention of its grant, in each Contracting State in respect of which it is granted, the same rights as would be conferred by a national patent granted in that State.*

(2) *If the subject-matter of the European patent is a process, the protection conferred by the patent shall extend to the products directly obtained by such process.*

(3) *Any infringement of a European patent shall be dealt with by national law.*

## 2.6 Mixing it all up

In the argumentation against the patenting of software-implementable inventions  
 5 regularly the most various aspects are lumped together and are considered in combination. At the same time, invention, software, implementation, clarity, scope of protection, technical nature and triviality are considered, and if in this mixture/concoction something seems to be awkward, this is used for substantiating the fact that patents on software-implementable inventions are inappropriate. In this  
 10 way different permutations of actually always the same "arguments" come into existence. Different legal questions are represented mixed up with others which, in a skilled way, would have to be regarded separately and would then become clear.

## 2.7 Passing over other criteria of exclusion

15

One pretends that the software exclusion of Art. 52 EPC would be "the last barrier before the unlimited patentability" and if it falls there would be no longer limits for the patentability of human thinking. What is overlooked or intentionally ignored are two points, namely firstly that the existing software exclusion actually bypasses the  
 20 issue and, thus, has no "filtering effect", and, secondly, that there are substantially stronger barriers than the software exclusion, namely in particular the restriction of what is patentable to subject-matter which is technical and inventive. These aspects regularly go under, and instead the stale impression is conveyed that there are certain circles who aim for patenting all and everything.

25

## 2.8 Conspiracy theories

One alleges that those who favour patents on software-implementable inventions either have a lack of understanding or pursue dark intentions of their own. One as-

sumes that patent attorneys intend to open up a new field of business on the basis of the revision of the EPC or of the European Guideline, the USA want to usurp Europe with regard to the patent law, as it had already done in the case of Japan, etc.

5

Properly speaking, these ways of illustrating the situation are of secondary importance with regard to the alleged unlawfulness of patents on software-implementable inventions as discussed under item 2.4, for the shown scenarios only "work" if it were actually correct that patents on software-implementable inventions have previously been admissible, what is, however, not the case. For discrediting the standpoint in favour of a reasonable protection by a patent, these scenarios are readily appropriate.

10

2.9 "270,000 votes, 2,000 companies against patents on logic systems"

15

The opponents of the directive refer to a high number of supporters of their own position. This is regularly put forward and used for justifying the own requests. These figures cannot be verified. Nevertheless, they are used and it may be that the high figures quite impress the one or other member of the Parliament.

20

If one regards the allegedly 30,000 granted patents on software-implementable inventions as "vote by feet", the cited high number can be contrasted by an impressive figure, for these 30,000 "expressed opinions" have not been obtained by sending cheap e-mails or, in an efficient way, by fishing signatures on exhibition stands (Cebit, Systems) in the Open-Source hall, but by a behavior costing many thousands of Euros. And what must also be remarked here is that large IT associations, e.g. in Germany the "Gesellschaft für Informatik (GI)" (Association for Informatics), speak in favor of a reasonable protection by patents and for fitting the software industry into conventional views held by engineers.

25

## 2.10 Myth SME

"SME" stands for "small and medium-sized enterprises". The opponents of patents  
 5 on software-implementable inventions state that patents would especially affect  
 small and medium-sized enterprises. Here one flirts with the part of David in the  
 David-versus-Goliath scenario and tries to gain sympathizers. However, according  
 to the Author's opinion, seen in a sober light, it is just the other way round: If one  
 takes the David-versus-Goliath scenario, an innovative David having one or more  
 10 patents in hands would have better chances against a more portly Goliath, this being  
 in contrast to the situation where patents are denied for David as well as for Goliath  
 in the same way. In this case one would remain on the level of a trial of strength on  
 the market, and there Goliath might, as a rule, have the better chances.

15 By the way, the opponents of the directive absolutely make a difference between  
 "good" SMEs and "bad" SMEs, what shows that the interests of SMEs as such are  
 actually not concerned. The "goodies" are those who, themselves, write programs  
 and software, who are perhaps also innovative, but do not have their results pat-  
 ented. The "baddies" are those who are small enterprises, too, make software-  
 20 implementable inventions, but have the same patented, and then – with or without  
 own software – request royalties. If such an SME competes as David against  
 Goliath and perhaps even succeeds<sup>9</sup>, this is not regarded as a success of an SME  
 owing to the patent system, but the SME is a bad SME and although the sympathies  
 are not with Goliath, one can empathize with Goliath's getting annoyed with such  
 25 SMEs being a pain in the neck. In order to round off the matter this annoyance is  
 not put down to Goliath's infringement of the law, but to "software patents", con-  
 nected with the reference that soon SMEs, too, will have to worry about "software  
 patents" in the same way.

---

<sup>9</sup> e.g. Eolas, cf. eolas.com

## 2.11 "I emigrate"

A situation is evoked in which one can allegedly no longer generate programs for  
 5 there are nothing than patents. Dramatically emigration and loss of job are evoked  
 as only possible alternatives. Irrespective of the fact that in this way one cannot  
 evade the territorially extending protection of patents, and further irrespective of the  
 fact that the protection conferred by a patent just not extends to research and devel-  
 10 opment, it is further puzzling that if an audience of many hundred people are asked  
 who of them had to refrain from anything because of a patent, or who, as a precau-  
 tion, had made a patent search at least one single time, nobody puts his hand up.  
 One only gets answers that one is worried or that one should be allowed to act for  
 reasons of prevention (in particular to prevent a situation as it exists in the USA).

## 15 2.12 Consequences in Europe because of circumstances in the USA

Like a duck takes to water the opponents of the directive use real or alleged condi-  
 tions in the USA for substantiating their concerns in Europe. This is done directly,  
 or a in a more sophisticated way, with regard to preventing conditions like those in  
 20 the USA. This approach, however, is - according to the Author's opinion -  
 inappropriate since "US conditions" as far as they are actually alarming do certainly  
 not reside in the fact that the US law does not have a software exclusion in its pro-  
 visions regarding the filing of patents. If one is skeptical about the US legal system  
 this can often be put down to the noticeable frequency of law suits and to the - real?  
 25 - absurdities which are reported with relish, but it has nothing to do with provisions  
 regarding software. Therefore, conclusions as to software discussions in Europe are  
 out of place.

2.13 "One would not have been able to create Linux, if patents on software had been possible"

This "argument" is utterly wrong, for its premise is wrong: There are patents on software-implementable inventions, allegedly even 30,000 of them in Europe (cf. item 2.4), and, nevertheless, Linux and other Open-Source products live without being troubled in the least. They do not only exist in Europe, but also in the USA where they have already been through the alleged "Fall". For Europe the contradiction is explained - quasi-judicially, but completely wrongly with regard to the substance - by the fact that the legal situation, in particular Art. 52 EPC, makes the enforcement of the patents in question impossible, so that the patentees presently (still) collectively remain silent. An answer to the question how the contradiction is solved for the USA where there is no software exclusion corresponding to Art. 52 is not given any more.

15

2.14 Reducing patents to prohibitive rights

On the part of the opponents of the directive the design of interests by the patent law is reduced to the exclusive representation of the prohibiting aspect. Surely this aspect exists and it will certainly sometimes constrain enterprises in their commercial activities, but patents also have stimulating, remunerating and securing aspects and, thus a significant effect regarding the promotion of development, as it was illustrated above, before item 2.1. These aspects are regularly not mentioned, they are simply ignored.

25

2.15 "Patents on software are trivial"

The mention of this argument is intentionally made at the end of the previous enumeration, for the Author regards this argument as the likeliest one to be attributed a

core of possible relevance. In each of the above-mentioned contexts it becomes expressly or implicitly obvious that one regards the subject-matter of patents on software-implementable inventions - at least in the very most cases - as trivial and, therefore, the patent as wrongly granted. This view stirs up an essential part of the outrage directed against patents on software-implementable inventions.

Partly this opinion can certainly be attributed to ex-post considerations. Considering a particular patent the content thereof - starting from today's knowledge - is regarded as trivial. Sometimes the considered patents, however, are US patents of early filing dates, which - because of the then different regulations regarding lifetime - are still in force in the USA today, but not in Europe. However, not all cases of perceived triviality can be explained therewith. A classical case was again the one-click patent of Amazon, which - in brief - protects the idea that for purchasing from the internet one can - starting from the representation of the product on the screen - effect a purchase by only one single mouse click instead of several ones as it was common practice before (US 5 960 411). The cited one-click patent, however, is far from being the only one that causes discomfort. Almost all patents are regarded as trivial and one can even notice efforts to the effect to justify an inherent, system-required triviality for software-implementable inventions.

20

In its most proliferative excesses this line of argumentation must surely not be taken seriously, especially since it is obviously opposite to the very experiencedly put forward statement - as touched above under item 2.1 - according to which the software industry is allegedly extremely innovative, what is not in agreement with the notorious triviality of the inventions which come into existence in this connection.

25

Anyhow, the question should be raised here whether the reproach of triviality does not touch a sore point. If, confronted with the reproach, one refers to the provisions in the patent law that inventive step is one of the requirements of patentability, one

raises scornful laughter connected with the remark that this criterion is no longer applied and that it has finally not been or incorrectly used.

According to the Author's opinion the latter cannot be denied for every case. If the  
 5 opponents of the directive see themselves as "standing in a jungle of trivial patents" this is certainly presentation techniques and exaggerated. However, if one states that, in fact, patents are also granted on banalities and obscurities, this cannot be denied for every case - by the way also in technical fields which do not relate to software-implementable inventions. At this place - according to the Author's opin-  
 10 ion - attention should be paid to the following: It might be common sense that patenting trivialities is not desirable, for a lot of patents on simply modified articles would not be qualified to come under the justification of the protection conferred by a patent in general, it would even have the opposite effect, namely economically not desirable and unjustified obstructions and an encroachment of the law into the field  
 15 of economy.

At the annual meeting of the American Intellectual Property Law Association (AIPLA) at the end of October 2003 the Chairman of the Federal Trade Commission (FTC), Mr. T. J. Muris, held a speech which directly touched the problem  
 20 „questionable patents“<sup>10</sup>. On the part of the FTC relief is regarded as necessary and some proposals were made in this respect<sup>11</sup>, which, however, rather tend to be characterized as procedural approaches. Anyhow, one can notice that in the USA government agencies call for action and make attempts.

### 25 **3. Tactics**

It can be noticed that, at a progressive rate, tactical tricks suppress a matter-of-fact argumentation. This holds true for both sides in the discussion.

---

<sup>10</sup> cf. <http://www.ftc.gov/opa/2003/10/murisaippla.htm>

On the side of the opponents of the directive one notices that with staying power and intensity effective lobbying is staged by the right party at the right time. Already what becomes publicly noticeable thereof requires time, know-how and  
5 money. According to the non-provable assumption of the Author one or several persons are either paid for it or released from work by interested companies in order to do this job. It is hardly imaginable that this job is done exclusively "after closing time".

10 Also on the side of the supporters of a reasonable protection by a patent tactics can be recognized. The European Commission keeps the subject under the heading "computer-implementable inventions" just as if the computer had raised this problem in connection with the patent law or had been customized according to the invention. Clearly this is not the case: Neither are computers in the case of software-  
15 implementable inventions customized according to the invention nor do they raise problems regarding the patent law. It is rather the software which as implementation of the invention is accordingly customized, and it is the software exclusion in the patent law which raises the legal problems. However, in the hope to be able to let the directive pass unnoticed one chose an uncontroversial wording. Some of the  
20 material questions, however, were therefore not dealt with. What was striking, e.g., was that, strangely enough, the original draft of the directive of the Commission hardly dealt with the problems arising from the software exclusion of Art. 52 EPC. Rather the technical nature of software-implementable inventions was commented upon, what absolutely is an interesting problem, too, but - seen from a material  
25 point of view - is off target regarding the software exclusion of Art. 52 EPC.

---

<sup>11</sup> cf. <http://www.ftc.gov/os/2003/10/innovationrpt.pdf>

The result of all this is the attempt of the EPC to bring the respective regulations in line, which failed already years ago, and a, for the time being, unsuccessful project of a directive of the EU with open outcome.

#### 5 4. Interests

A look on the various interests may be helpful:

The opponents of patents on software-implementable inventions finally wish that their "matter", namely software, is free from interferences by the patent law. The wish is also based on the Open-Source idea and can be formulated as follows: "Open Source is a comparatively unique development model for software. It does not only enjoy life, but also lives from the idealism of the software engineers. It would be a pity if this was stopped by patents". Apart therefrom there certainly also exists the non idealistic wish regarding own interests, namely not to be molested by the rights of third parties.

Admittedly the idea originating from the Open-Source model is not unappealing and just in its most visible result, namely Linux as the alternative to Windows, one wishes that this track is successful. However, it is just not justified to state the irreconcilableness of Open Source and patents on software-implementable inventions. The best proof for the contrary, i.e. for the reconcilableness of protection by a patent and Open Source, is reality itself: Linux and other Open-Source products exist, although there are (allegedly) 30.000 "software patents". It has hitherto not been heard that Linux would have had any difficulties. And surprisingly this is not only the case in Europe, but also in the alleged boiling pot USA where everything is said to be much worse than in Europe. Also there, particularly Linux and generally Open Source, enjoy a happy life undisturbed by patents, cf. in this respect item 2.13.

Linux has long become socially acceptable in Europe as well as in the USA. It is so unrestricted by the law that even large companies as, e.g., IBM decided to support this track and to take on this image. To cut a long story short: According to the Author's opinion the insinuated irreconcilableness of Open Source, on the one hand, and patents on software-implementable inventions, on the other hand, does not exist. Reality in Europe as well as in the USA shows that the systems agree together. The - from a consumer's point of view - absolutely desirable creation and business model Open Source is not a reason to request that software-implementable inventions are excepted from being patented, for presently Open Source and the patent system in Europe as well as in the USA largely peacefully exist side by side.

A reasonable protection by a patent under the well understood criteria for patentability - here especially technical nature and inventive step - might be approved of by the majority of those active in the economic field. In Germany the large umbrella organizations, software lying in their range of competences, -first and foremost the "Gesellschaft für Informatik (GI)" - were in favor thereof. This corresponds to the view that the "matter" software is increasingly regarded in normal industrial-engineering contexts and, therefore, there ensue the same interests in particular with regard to the protection of real inventions.

20

Nobody or only cynics will wish a patentability of all and everything. This would reduce the system to an absurdity. If the actual handling of the system ranges beyond the economic justification of the system, this will lead to a loss of acceptance and reason and will thus become a disadvantage of location for the included territory.

25

## **5. To do**

Is the decision of the European Parliament a catastrophe? Or will it eventually be regarded as an incident? The Author cannot answer this. Since, however, from today's point of view, it is at least not certain that everything will come right by itself, in the following a to-do list is boldly presented:

5

### 5.1 The EU directive

As regards the EU directive, what has come from the European Parliament is very problematic. If one day there is to exist a reasonable EU directive, a lot of work on  
10 the substance as well as on convictions will be necessary to bring the same about. In view of the grinding mills one could absolutely become sceptical. Obviously the supporters of a reasonable protection by a patent are far worse organized than are the opponents of the directive, what the latter would certainly formulate in a modified way, namely that the supporters of a patent protection had exercised their influence rather not publicly, in brain trusts and on the level of the executive, whereas  
15 the opponents were active more at grass-roots level by addressing the public and the representatives of the legislative organ, who had "fixed it up".

Where this arm wrestling will lead to is - according to the Author's knowledge and  
20 view - not foreseeable. The opponents of the directive, however, are optimistic. Upon the question posed by an insecure supporter: "A complete rejection of the whole [i.e. the directive - Author's annotation] is what suits us, isn't it?", the master mind of the community, an educated linguist, mailed on 24 September 2003, directly after the decision of the European Parliament had been taken: „No. The directive is now our project. The roles have been exchanged”<sup>12</sup>.

25

### 5.2 Technical nature

---

<sup>12</sup> cf. <http://lists.ffii.org/archive/emails/swpat/2003/Sep/0723.html>

It is the Author's opinion that questions regarding the technical nature of an invention may absolutely be more critical in the case of software-implementable inventions than in the case of inventions which need, e.g., metal for their construction. Problems regarding technical nature correlate with software-implementability of an invention possibly in the same way as red hair with persons of Irish origin. Although it would de jure be justified - in view of the imminent software discourse - to answer "Other Issue" as to the question of the technical nature of inventions, it would nevertheless be helpful, if one also made a fair bit of headway as far as technical nature is concerned.

10

As regards the question of technical nature, in the last few years a systematic approach - similar to the Federal Supreme Court ruling "Rote Taube" - cannot be recognized, and such an approach, even beyond the (quasi-)judiciary bodies, is hard to make, for at last just the judiciary bodies must work out what is to be regarded as technical - irrespective of the fact that also from a material point of view it might be very difficult to develop a convincing definition or even only a catalogue of good criteria.

15

Nevertheless, if anything further is determined with respect to the question of technical nature of software-implementable inventions, as at least one indication what is inferable from the linguistic usage of the persons concerned could be used: In the jargon of the software-creating circles one can find:

20

term	hits in google.de (status : October 2003)
Data mining	2.560.000
Virtual reality	2.160.000
Data warehous(ing)	956.000
Software tool	280.000

Software interface	88.600
"Softwareschmiede" (software forge)	28.000

The usage as to software-related subjects, e.g. terms like "tool", "mining" and "Schmiede" (forge) strongly reminds us of primary and secondary sectors of adding value by industrial production: in this jargon, data are regarded as resources and the tools for unearthing, refining and processing them are built/implemented by software. Anyway, this is an indication that nowadays software as a means of implementing inventions is regarded in a largely normal engineering context, in which conventional technical considerations, e.g., optimizing the utilization of resources, price-reduction, improving reliability, creation of new offers by combining individual technical structures, and the like, are made and then lead to certain solutions which are (also) implemented by software. The resulting economic and legal (patent law) interests are then largely "normal", so that - from the Author's point of view - it cannot be recognized that in view of software as a means of implementing inventions an a-priori scepticism regarding the technical nature of such inventions would be expedient.

A modern definition of technical nature - according to the Author's opinion - would have to include the fact that software has become a "material" which is to be regarded in contexts as they were seen by the digital technology perhaps 25 years ago, by electron tube technology 45 years ago and before by relay technology. This is not to speak in favor of the technical nature of software itself, for technical nature does not matter, since software never is the invention; it shall rather be said that

- software shall not obstruct one's view on the technical nature of the inventions implemented by it,

- handling of, working with as well as talking about software, however, admits or excludes to draw conclusions regarding the technical nature of the implemented inventions,
- 5
- in which connection it can absolutely be a non-trivial object to concretely develop helpful criteria.

### 5.3 Definition of "software"

- 10 It may be surprising that the question of the definition of the term "software" is mentioned here as a separate item, and it may be that, from a practical point of view, too much importance is attributed to this question. Theoretically, however, this question is important, for just on the side of the opponents of the directive the opinion is more or less strongly upheld that normal sentences like "print the figures
- 15 from 1 to 10" and "correlate country of birth with color of hair for 10.000 persons" are not only software-implementable, but are themselves directly software. Various lines of argumentation are used therefor, which are only sweepingly characterized here by respectively one sentence.
- 20
- To put it very generally: software, as a work of speech and as sequence of instructions, is Janus-faced, and therefore it is normal speech, too; there is no difference between software and normal speech.
- 25
- To put it more concretely: A parser can be programmed for every sequence of instructions in normal speech, which parser makes the normal speech readable by a machine and supplements the lacking indications which are needed for being executed by the machine, so that the normal speech becomes comprehensible for the machine and, therefore, is software.

- In theory, independently of time, and, in practice, in the near or remote future, machines would understand speech and would contain so much background knowledge that they can use normal speech as well as software directly as instructions. The difference between normal speech and software that is made today is only based on the (still) present technical inadequacies, but is theoretically not present and, in practice, will disappear some time.

What is important is the question what software is if one again looks at Art. 52 EPC: If, in fact, normal speech were regarded as software, possibly many claim formulations in normal speech would directly come under the today valid software exclusion and would not be patentable. Certainly, as regards the result this is an absurdity, it is however endeavored - as illustrated above - to propagate just this, in order to except in this way software-implementable inventions directly from patentability. Therefore, if there remains any kind of software particularity in the regulations or is created, then one would have to consider the question what is actually intended by software, so that there won't be a bad end.

#### 5.4 Inventive step

Beyond the EU directive it would absolutely be advisable to examine whether and in how far the "fear" of trivial patents is justified. From a point of view of the patent law this would be an examination to the effect whether search results during official search are sufficiently complete and whether the criterion of inventive step is applied in a reasonable way when a patent is granted. If inadequacies show up, they would have to be overcome. However, the question of software is hardly concerned here any more. This finally justifies the question mark in the headline of this paper.