Written Statement to the Extended Board of Appeal

for G03/08 ("limits of patentability in the field of computing", Referral under Article 112(1)(b) EPC)

Submitter: **ASOLIF**

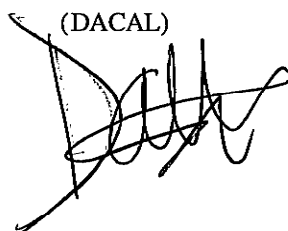**FEDERACION DE ASOCIACIONES DE SOFTWARE LIBRE DEL ESTADO ESPAÑOL**

G-06556955

C/VISTAHERMOSA 1 06200

ALMENDRALEJO (BADAJOZ)

Telf: 924 677 316 Fax: 945 069 750

President: Daniel Armendáriz Crespo Aranda Landa

(DACAL)

(This statement has being provided by FFII e.V., Mr. Rebentisch, for the ASOLIF submission)

Overview

A The quest for legal protection

In October 2008 the President of the European Patent Office issued a Referral G03/08 to its Enlarged Board of Appeal with questions on how to examine and grant software patents in Europe. Accordingly, we are grateful for the invitation to contribute a written statement to this process. The exclusion of computer programs from the scope of patent law is a highly controversial issue which, at the same time, is of great political and economic significance for the competitive rules governing software development in Europe.

A.1 Patent failure in a nutshell

The patenting of programs for computers was debated as a possibility so that computer programs might be awarded protection, however, it has never been endorsed by a legislator in Europe. Moreover, the European Patent Convention lists under Article 52(2) a statutory exclusion of software-related methods which are not considered as inventions. Its 2000 revision added further limitations regarding inventions in fields of technology. The current political controversy stems from the patent office's permissive practice of granting rights related to software, as well as affirmative opposition decisions which govern the granting process, and the unidirectional influence this administrative practice exerts on corresponding national post-grant court decisions in the contracting states.

A major concern of software authors is that overlapping exclusive rights without merit might compromise the normal commercial exploitation of their works. The prevalent opinion states that patents should not become hunting licenses. Patenting of software methods rewards authors of a highly formal documentation on derivative development aspects with a patent. That exclusive right is granted at the expense of the freedom of profession and action, which (potentially) excluded software developers want to enjoy without unreasonable hidden risks. The core of the patent paradox manifests itself in a novice question: "How can someone be excluded by a patent for something new?" When a state imposes a patent system, priority becomes a key rule of the game. Patenting requires a social contract concerning the relevance of such a priority. For most commercial activities, novelty is deemed unimportant and we would be unwilling to accept such a rule of the game, for instance if a business privilege system was based on application priority.

How to strike a balance between meritless and priceless solutions - and on which level? A precedence of ideas and principles underlying computer program works that corresponds to the novelty of software-related methods in

patent terminology could have limited value in a dynamic market driven by evolutionary imitation competition. In particular, this would be the case if patent applicants contributed less to the arts than developers who are restricted by institutional, financial or capacity-related barriers to obtain patents. As regards patenting, the two choices small players face are as following: To become either a market player with "defensive patents" or a patent "troll" without a business, except that of licensing. For larger players we usually find strategic patent portfolios with the option to enter perfectly legal cross-licensing cartels.

Ideas are said to be cheap.[1] What if not research & development investment and creativity, but the mere capacity of patent agents was the bottleneck for obtaining patents in the software field? The patent system will malfunction when applied to a field of potentially ubiquitous matter. The very purpose of an incentive instrument is to generate more output of the scarce matter than would be the case under laissez-faire conditions. Therefore scarcity under laissez-faire is a fundamental precondition which has to be met for a rational application of the system. A scarcity of ideas and principles underlying software works does not exist in the software field. Rather, this field is characterised by an opulent abundance of creative solutions. Actual professionals skilled in the art deplore the inherent triviality of most software patents and show little professional respect for the substance of these patents. Raising the inventive step requirement seems to be a suboptimal solution to the triviality problem. How much would a software developer pay for the disclosure of knowledge that is documented in a patent filing? Which patents do professionals want to read to gain relevant knowledge? In the answers to these questions you may find the solution to the triviality problem.

A.2 Hammer and nail

One of the earliest proposals for an international program protection through patent and trademark offices was made by Helmut Oehlschlegel[2] who suggested that the patent office create a program licensing library to promote the re-use of programs and allocate developer capacity with greater efficiency. Today, the proposal of a central software repository conjures up a distinct notion of a planning economy on developers, but that notion is carried by the hubris of software patent proponents spinning the tale of the corpus of patents reflecting the state of the art of data processing. Computer algorithm progenitor Donald E. Knuth, who describes his work as art, speculated in the early 1970th about the prospect of future patenting in his field but later

---

1 Spivack, Peter G., Comment, Does Form Follow Function? The Idea/Expression Dichotomy In Copyright Protection of Computer Software, 35 U.C.L.A. L. REV 723, 729-31 (1988), 744: "At the lowest level of abstraction, a computer program may be thought of in its entirety as a set of individual instructions organized into a hierarchy of modules. At a higher level of abstraction, the instructions in the lowest- level modules may be replaced conceptually by the functions of those modules. At progressively higher levels of abstraction, the functions of higher-level modules conceptually replace the implementations of those modules in terms of lower-level modules and instructions, until finally, one is left with nothing but the ultimate function of the program.... A program has structure at every level of abstraction at which it is viewed. At low levels of abstraction, a program's structure may be quite complex; at the highest level it is trivial".

2 Oehlschlegel, Helmut: Sollen und können Rechenprogramme geschützt werden?, GRUR 1965, 465-468

dismissed it as an appropriate instrument for software-related protection.[3]

What the legislature around the world did put into place were sui generis protection regimes as the applicable instrument of law for the works of software developers which enjoy authors' rights. An existing regime - copyright - is applied with certain special limitations and modifications to computer programs. As a means of software protection, copyright has generally been met with wide acceptance among software developers and users. It gave legal certainty to the commercialisation of software services. The actual works apparently incorporate the essential elements that deserve legal protection.

From this perspective, imitation competition may be perceived as desirable, while patent law underlies exactly the opposite assumption. Concerning the protection of software, the literature also discusses unfair competition law which can be used as an instrument to achieve the necessary politico-economic outcomes, namely to protect a market from too much imitation competition. A competition law based protection instrument for software emerges from actual works; without a need for formal "disclosure", without slow and bureaucratic examination, and without the friction costs of patenting which may translate into prohibitive market entrance barriers. Today, the debate about software protection by unfair competition law looks academic but provides a more flexible alternative.[4]

It is remarkable that both legislation and jurisprudence found little merit in protecting underlying ideas and principles under copyright or competition law. Granting authorities, on the other hand, felt invited without changes of statutory laws to intervene in the software sphere and to treat the alleged protection gap with their own medicine: patent protection.

A.3 Invited by Procrustes

As a response to small and medium software entrepreneurs' reluctance to obtain patents for themselves, software patenting proponents and institutions intend to raise awareness about the merits of patenting. That is just one example for institutional preconceptions about the universal desirability of patents for any subject matter. Political agitation and intervention of the EPO into public and parliament deliberations are symptoms of a widespread patent evangelism and institutional inbreeding. It comes as no surprise that the referral is communicated together with a reissued propaganda leaflet - its insightful title being "Patents for software?" - that outlines a misleading description of the current practice. From an orthodox patent perspective, market conduct ought to be streamlined with the institution's perception:

> "A computer-implemented <u>invention</u> is an <u>invention</u> whose implementation involves the use of a computer, computer network or other programmable apparatus, the <u>invention</u> having one or more features which are <u>realised wholly</u> or partly <u>by</u> means of <u>a computer</u>

---

3 http://www.heise.de/ct/Donald-E-Knuth-ueber-MMIX-und-die-Kunst-des-Programmierens--/artikel/124680
4 Sidler, Peter: Der Schutz von Computerprogrammen im Urheber- und Wettbewerbsrecht, 1967

program. The term software is <u>ambiguous</u>. It is generally understood as the implementation of an algorithm in source or object code, but <u>without distinguishing between technical and non-technical processes</u>."[5]

It is surprising that it still seems opportune to contribute to cognitive dissonance with washing machine legends and general claims of benefits to society. Apparently irrelevant for the normative question about a specific scope extension is advocacy of the following kind:

"It is not only the innovators that benefit from patents. As consumers we all benefit in incalculable ways from the development of technology facilitated by the patent system. As employees, our jobs may depend on a particular technology and the patents protecting it. Finally, as citizens, we all benefit from the technological progress supported by the patent system and the contribution it makes to the European economy."[6]

Many elementary normative shortcomings of patent reasoning were debunked by Fritz Machlup in 1958[7]. Those generic revenants, classical fallacies of patent reasoning, refuse to fade despite their falsification in literature. Economists are quoted but seldom understood by lawyers. Even for the domain of classical engineering, empirical results put the shiny statement above into a tenuous position. For software as a subject matter we refer to the studies of Bessen/Hunt[8] and others[9].

Though patent granting in the software sphere is a public service without ordering by a democratic legislature, software authors cannot simply opt out from the exposure to the legal risks when they disagree with patenting. Practitioners are challenged to spent their financial resources to oppose grants, to obtain patents for "defensive purposes", to advocate why they do not find the system appropriate, why their field "should be excluded" and any arguments why software patenting was "broken" are dismissed with ease: patenting is said to be equally "broken" in classical engineering and software was by no means special.

All patent reasoning tends to lead to the finding that the software sphere can under no circumstances withstand the eligibility of a patent. A non-application of patent law for software is said to be impossible:

•on grounds of non-executive trade agreements with the underlying objective that patent systems are not abused as a barrier to international trade, and their creative wide interpretation.[10]

5 European Patent Office: Patents for software? http://www.epo.org/topics/issues/computer-implemented-inventions/software.html
6 European Patent Office: Patents for software? http://www.epo.org/topics/issues/computer-implemented-inventions/software.html
7 Machlup, Fritz: An Economic Review of the Patent System, Washington: Congress, 1958, p. 40f
8 Bessen, James; Hunt, Robert M.(2004): An empirical look at software patents, in: Journal of Economics and Management Strategy 16, no. 1, pp. 157-189, 2007
9 SSP.EU Kampagne, wiki collection of relevant economical studies: http://stopsoftwarepatents.eu/wiki/Studies
10 For the research debate around the "policy laundry" phenomenon, cf. Hosein, Ian: International Relations Theories and the Regulation of International Dataflows: Policy Laundering and other International Policy Dynamics, Paper

•on grounds that software was an ambiguous term and not well-defined with a grey layer to hardware, as if the physical character was an essential condition for being patentable.[11]

•because of patent reasoning and terminology. You may find there cannot be such thing as a software invention under the EPC. But how should a patent application be addressed that describes something that is no invention? A "software-related non-invention"? Loaded terminology such as "computer-implemented inventions" implies an invention, thus eligibility to a patent in principle.

•on grounds that the legislator made a non-exhaustive exclusion list when what was really meant was the question as to whether the matters were technical or not; all software is described as technical when run by a computer powered by electricity, when data represents physical objects, when the algorithm calculates "faster", i.e. takes less time etc.

A.4 Bitter sweet porridge

The diagnosis concerning the political economics of software patenting is threefold:

1. The patent reasoning lacks clear negative teachings for key terms such as "invention", "technical", "industrial"/"useful", "field of technology".[12] We do not find stable conceptions about what does not constitute an invention, what is not patentable, what is not technical and so forth.

2. The patent system is unable to describe the practical scope of an exclusion and so the reasoning mechanism leads to an exclusion of the exclusion. Patenting is conservative in what it excludes from patenting and liberal in moving the boundaries set by the legislator by analogue reasoning on grounds of "fairness".[13]

3. Patent reasoning often carries the cynical qualities of an inquisitory examination because of sublimated preconceptions about economic desirability. It is ignorant towards the statutory mandate and it is driven by direct or indirect conflicts of professional or institutional interest.[14]

---

presented at the annual meeting of the International Studies Association, Le Centre Sheraton Hotel, Montreal, Quebec, Canada, Mar 17, 2004; in this referral we see amici argue on grounds of TRIPs 27. Cf. Lenz (2005):TRIPS and European software patent legislation for the FFII, http://eupat.ffii.org/papers/europarl0309/amends05/lenz05/ and Madras High Court August 6, 2007 suggestion to invoke WTO dispute settlement in lack of its competence.

11 UK case law polemics against an "any hardware approach" of the EPO is an example for such sock puppetry.

12 Cf. Machlup, Fritz: An Economic Review of the Patent System, Washington: Congress, 1958, p. 40f

13 [2008] EWCA Civ 1066, [2009] RPC 1, "It can also be said in favour of Symbian's case that it would be somewhat arbitrary and unfair to discriminate against people who invent[?] programs which improve the performance of computers[?] against those who invent[?] programs which improve the performance of other machines."

14 As an example cf. Pilch-Tauchert: "Is it really appropriate to make software patentable, without conducting a systematic study of the economical effects of such a change first. Tauchert, DPMA: Absolutely. We don't need economic studies. The reality speaks for itself. The market already made its judgement. We receive every year thousands of applications

Without stable boundaries that limit the scope we operate in a sweet porridge environment which threatens the governance of the system: In 2000, for instance, the EPO President suggested to the Trilateral group that similar to IBM/Computer program product[15], creative interpretation could set the course for patenting business methods, too:

> "It would be possible to argue by analogy with the discussion of "programs for computers" in T1173/97 (OJ 1999, 609) that a claim directed to an abstract business method itself is not necessarily for a business method "as such"."[16]

The constellation is always the same: No prior democratic authorisation process based on a "rationale of a market system" (Hayek) but a case-driven legal escapism that is in principle immune to argument. Even if a legislator had democratically endorsed patenting of software and all relevant stakeholder had expressed their broad consent, the political economics of patent reasoning would reveal its unbalances when applied to another fresh field.[17]
Patenting is merely limited by an informal patent community consensus to stay back and to refrain from seeding patent eligibility to yet another commercial activity for which the patent system was not designed. King Midas only learnt his lesson when faced with personal consequences:

MISCUERAT PURIS AUCTOREM MUNERIS UNDIS
FUSILE PER RICTUS AURUM FLUITARE VIDERES.
ATTONITUS NOVITATE MALI DIVESQUE MISERQUE
EFFUGERE OPTAT OPES ET QUAE MODO VOVERAT, ODIT.
COPIA NULLA FAMEM RELEVAT; SITIS ARIDA GUTTUR
URIT, ET INVISO MERITUS TORQUETUR AB AURO
AD CAELUMQUE MANUS ET SPLENDIDA BRACCHIA TOLLENS
DA VENIAM, LENAEE PATER! PECCAVIMUS INQUIT
SED MISERERE, PRECOR, SPECIOSOQUE ERIPE DAMNO[18]

The Referral gives you the opportunity to take responsibility for developers and reconsider the ongoing escapism and unintelligible legal fabrications of the Boards of Appeal such as T1173/97 and to terminate a trickle-down persuasion

---

for software patents and our patent system is profitable. It feeds without state subsidies 20000 patent specialists.", http://eupat.ffii.org/events/2000/linuxtag/index.de.html

15 T1173/97

16 EPO-President note to Trilateral Offices, Examination of "business method" applications, May 19, 2000, http://www.european-patent-office.org/tws/appendix6.pdf

17 In principle many other copyrighted works, for instance movies could be next. Of course story plots can be outlined in the formal fashion of an invention and claimed. Is the terminology stable to discard such applications? How would you argue in legal terms that a story inventor in the movie industry should not be able to get a patent for his motion picture implemented invention; notably the plot is not patentable "as such": Only when a movie-implemented invention is novel, involves a genuine inventive step and has technical effects which can be found in recording and editing. Presumably the problem of the movie sector would not be patenting but the quality of the granted patents on plots... and patent offices should better raise the bar. Consider that these inventions can be implemented in movies, animations and theatre play performances, therefore it would be unfair to exclude the [currently non-patentable field] from patentability. TRIPS 27 mandates that patents have to be granted in all fields of technology. Acting has strong technical aspects insists the Method Actor's guild.

18 OVID, Metamophosis, XI

of professional national judges by continuous granting under disruptive interpretations.

B The decision on the scope

B.1 Great responsibility without a rationale of a market system

> "In the absence of guidance from the law and its preparatory documents, and in view of the existence of divergences of opinion regarding how the computer program exclusion should be applied, it is considered appropriate at this stage to refer the questions ... to the Enlarged Board of Appeal for its opinion" [19]

In the absence of legislative initiatives an Opinion of the Enlarged Board of Appeal could well cause almost identical effects as the proposed EU software patent directive 2002/0047/COD without a broad political and economic debate concerning the merits of software patentability. Online pundit PA Alex H. Horns expressed general discomfort with a political Referral:

> "In fact, this undertaking looks as if Ms Brimelow wishes to dump the by and large still unresolved problem of the limitations of patentability ... to the Enlarged Board of Appeal as a political body after the truly political bodies competent... have failed so far to find a solution that is self-contained, non-recursive and/or non-circular with regard to the concept of technicality and which has potential to last for many decades."[20]

We agree that Referral G3/08 forces a predominant political decision on the Enlarged Board of Appeal. It is doubtful that an Opinion would then lead to such a politico-economic debate and decision process, because the Opinion is authoritative:

> "How and where can the outcome of this referral be challenged?
>
> There is no mechanism to challenge an opinion of the Enlarged Board of Appeal as it constitutes the highest instance of the EPO. "[21]

In G02/06 it was clarified that Boards of Appeal of the EPO are not courts or tribunals of a member state of the EU, and there is no power under the EPC for a Board of Appeal to refer questions to the ECJ.

With that power comes great responsibility.

A fundamental constraint of the Opinion is that an economic rationale may not even be considered by the members of the Enlarged Boards of Appeal because these grounds are beyond institutional competence as well as available expertise. To demonstrate the limitations let us outline in generic terms a normative blank slate pattern for any matter that is not patentable yet:

---

19 Referral G03/08, p. 18
20 Horns, Axel H., Brimelow did it!, Ipjur Blog, Oct 25, 2008
http://www.ipjur.com/blog2/index.php?/archives/6-Ms-Brimelow-did-it!.html
21 http://www.epo.org/topics/issues/computer-implemented-inventions/referral/cii.html

> Making patents eligible for a entire new subject matter is primarily decided on the base of economic considerations. In terms of a simplified economic calculus it needs to be demonstrated that a market scenario in which patents can be obtained for the subject matter performs better than a market scenario without intervention by the government. Such reasons could, for instance, lie in an imperfection under dynamic competition which would justify limiting imitation competition. For an economist only a state intervention in the market with the patent instrument needs to be justified but not an absence of such an incentive system.

Examination procedures, Boards of Appeal, the Enlarged Boards of Appeal and Patent Courts are as a matter of fact unable to take decisions on grounds of an underlying rationale of a market system which would justify the application of patent law to a new activity or subject matter. Although their decisions often affect the scope of patent law, they are unqualified to take into account the rationale of a market system.

A decision of the Enlarged Board of Appeal[22] mentions the political detachment and autonomous character of the Munich patent system:

> "The European Patent Organisation as a public international organisation has an internal legal system of its own ... The EPC provides an autonomous legal system for the granting of European patents. In legal terms, neither the legislation of the contracting states nor the international conventions signed by them are part of this autonomous legal system. Within the framework of the system established by the EPC, legislative power rests with the contracting states alone and is exercised by either an intergovernmental conference (Article 172 EPC) or the Administrative Council (Article 33 EPC). The EPO is not itself party to the WTO and the TRIPS Agreement. Thus, the obligations deriving from the TRIPS Agreement do not bind the EPO directly but only such contracting states of the EPC as are members of the WTO and the TRIPS Agreement. In this connection, it is irrelevant whether some or all of the contracting states of the EPO are party to the TRIPS Agreement."

Founding father Otto Bossung propounded a sceptical view concerning a strong governance role for the European Patent Organisation at large.[23] The European

---

22 G 2/02 and G 3/02, (OJ 10/2004), dated, 26 April 2004
23 "Bedenklich ist dabei, daß die Exekutive der Organisation - das EPA - in einem bei internationalen Organisationen bisher unbekannten Ausmaß - an allen staatlichen Organen vorbei - über Individualrechte entscheidet und die Organisation gleichzeitig eine eigene Legislative umfangreicher Kompetenz - losgelöst vom parlamentarischen Raum besitzt. Unvermeidbar war dies, weil ein europäisches Patentsystem zwar leicht auf geduldigem Papier in Vertragtexten niedergelegt werden kann, aber ohne größere Möglichkeiten zu einer autonomen Anpassung an die unvorhersehbaren Erfordernisse der Praxis nicht lebensfähig ist.", Bossung, Otto: Die Münchner Diplomatische Konferenz über die Einführung eines europäischen Patenterteilungsverfahrens. in: Mitteilungen der deutschen Patentanwälte; May 1973

Patent Convention and national level equivalents as an applicable legal base leave the mission and objectives of patenting open. It remains the domain of a legislator, ideally pluralistic representatives from a parliament which are rooted in different legal traditions, political ideologies, ethical and religious beliefs to express their objectives in statutory provisions, and to balance the patent system with other fields as competition enforcement, standard setting and industrial policy. These representatives would conduct a pluralistic discourse with their electorate, consult and consider civil society, industry and social partners and request the provision of an economical impact assessment. Normative decisions remain beyond the scope of patent practice.

B.2 Flexible competence

In the Referral G3/08 the President endorsed a dynamic conception of the substantive provisions which is derived from a de minimis principle and draws conclusions for the assumed mandate of her institution:

> "As the EPC was drafted, the feeling was that it was better not to define the exclusion precisely in law, but rather that the matter should be left in the hands of the EPO and the national courts. This flexibility is important as technology develops and new technologies emerge. Nevertheless, to quote a working group in 1972: "it was stressed that a matter as important as computer programmes should not be left in a state of prolonged uncertainty pending legal developments". Diverging decisions of the boards of appeal have indeed created uncertainty, and answers to the questions arising from these decisions are necessary to enable the further, harmonious development of case law in this field."[24]

What we ought to keep in mind is as follows:

1. The patent practice needs to adapt to new technologies as they emerge. This does not imply a flexibility of statutory rules and interpretation.
2. Software has little in common with technology within the meaning of patent law, in particular the processing of data does not constitute a field of technology and lacks technical character. Permissive adaptation would weaken established legal concepts.
3. EPO activities deserve a specific contractual legal base under the EPC instead of open competence and usurpation, in particular because the institution is entrusted to grant rights for the contracting states which are effective on national markets.
4. Alleged "feelings" of the EPC founding fathers are a rather insufficient legal base and run into verification problems. It is common for a diplomatic conference to consider different options. Plenipotentiaries had no rational means to award the EPO with a dynamic permission to grant.
5. Consequently, related by the decisions does not justify expanding the statutory mandate, in particular as neither the BoA or the EBoA are patent courts but simply administrative institutions.[25] Decisions as T-1173/97 demonstrate a

---

24 Referral G03/08, p. 2
25 Decision "House of Lords Oct 26 1995, RPC 1996, p. 76 BMLR 1997, p. 201, IIC 1998, p. 325-331, GRUR Int 1996, p.825-831 (Terfenadin) (Merrell Dow v. Norton)." explains that EPO practice is not binding on national courts but had "great persuasive value" because they

disruption of the exclusionary interpretation concerning software which anticipated a change of law that did not take place.[26]

B.3 Risk burden-sharing

A flexible interpretation of statutory laws invokes the analogy by which it would be left to a prosecutor to make up a criminal offence and prove the guilt. In fact, in light of the exclusionary character for commercial activities, highest standards ought to be applied by the granting authority.

Once granted by the EPO, a European patent operates as a series of national patents for those contracting states in which protection is sought by the applicant. National customs enforcement authorities have to assume the validity of the granted right. In the contracting states both civil and criminal enforcement measures are available. Stronger civil enforcement measures for patents are harmonised under Directive 2004/48/EC, for instance, Mareva injunctions and Anton Piller orders. When we talk about enforcement we have to consider the role of "patent management companies", for instance Sisvel and its German trade fair raids concerning mp3 technology and the rush to judgement you find in the media coverage.

Since a patent granted by the EPO can be enforced against competitors to exclude or even criminalize certain commercial activities, the legal principles of nulla poena sine lege scripta et certa have be considered (cf. Bestimmtheitsgrundsatz Art. 103 Abs. 2 GG). A conservative scope for patent eligibility leads to no potential regulatory distortion, while market intervention would. Indeed, an infringement of a granted patent may even constitute a criminal offence for market participants in some jurisdictions, notwithstanding financial and commercial consequences. Thus in dubio mitius a smaller scope of patent protection applies. More conservative standards would lead to more legal certainty for businesses because patent ambushes deter the exploration of new commercial fields.

Article 138 EPC provides that all granted European patents may be revoked by national courts if the patent does not meet essential patentability criteria. It is apparent that any invalidation of a patent by a national court reveals an overly permissive examination practice which cannot be blamed on careless examiners but on examination guidelines. A supranational granting administration has - in absence of statutory clarifications or a European patent judiciary - to follow the most restrictive national post-grant interpretations[27] based on what it was authorised to grant. The examination and opposition practice for European patents of the EPO has to be aligned with the least permissive interpretations from all the contracting states unless there is permission to go beyond. In case of doubt, it would have to reject applications than to put an unjustified burden of risk on the market. As noted above, the

were "decisions of expert courts (the Boards of Appeal and Enlarged Board of Appeal of the EPO) involved daily in the administration of the EPC". We dispute the legal grounds for the assert of a court status.

26 For the mandate governing the BoA competences cf. EPC 21.

27 For EU member states Article 22(4) of the Brussels Regulation on jurisdiction and the recognition and enforcement of judgments in civil and commercial matters (EC/44/2001) confers exclusive jurisdiction on national courts for the validity of European patents.

current patent practice follows the exact opposite scheme and does not include the social costs of unauthorised granting. A consideration of the underlying social contract as a legal fiction would help internalising the social costs of unauthorised granting out on third parties.

B.4 Legislative role

As argued above, normative decisions concerning substantive patent law belong to the exclusive domain of the legislator. When patenting practice and administrative decision-making of the EPO Boards of Appeal (including national court decisions) diverge, legislative clarifications which are beyond the scope of administrative decision making bodies or courts come into play. Bipartisan proceedings before a court or answers to a structured interview posed by a President to members of an institutional body in a granting authority are an inadequate substitution for a broad democratic debate and legislative decision. The President, however, considers the harmonisation of national patent office practice to be an administrative task:

> "Currently there are concerns, also expressed by national courts and the public, that some decisions of the boards of appeal have given too restrictive an interpretation of the breadth of the exclusion. It is clear that the European Patent Office should have the leading role in harmonising the practice of patent offices within Europe. "[28]

On the substance, she articulates her awareness that the European Parliament, for instance, has acknowledged "growing concerns about undesirable patents in various fields and about a lack of democratic control over the processes by which such patents are granted, validated and enforced"[29]

For a role in national office harmonisation the European Patent Office would require a mandate. The European Patent Convention has to provide for these harmonisation objectives. Article 4(3) EPC provides "the task of the Organisation shall be to grant European patents. This shall be carried out by the European Patent Office supervised by the Administrative Council." Preliminary findings show the contracting states did not authorise the EPO to formally harmonise their national patent office practice.[30] Instead, this role was left, for example, to the Strasbourg convention and the EC legislator, as was also the case with biological matters (98/44/EC).

For those contracting states which are member states of the European Union further consideration needs to be attributed to the inclusion of patent law in the acquis communautaire by the adoption of the Directive 2004/48/EC and Directive 98/44/EC. Thus member states which avoid acting in breach of Article 292 TEC have an obligation to seek further harmonisation of their patent practice within the community legislature.

Uniform rules for European patents can admittedly be reformed by a Diplomatic

---

28 Referral G03/08, p. 2
29 2006/2557(RSP) European Parliament resolution on future patent policy in Europe
30 It is questionable if Article 112 EPC in connection with Article 21 EPC provides an applicable legal base.

Conference. For instance it was proposed for the EPC revision in 2000 that the phrase "programs for computers" be deleted from Article 52(2)(c) EPC
•see preparatory documents CA/PL 6/99; CA/PL PV 9, points 24-27; CA/PL PV 14, points 143-156; CA/100/00, pages 37-40; CA/124/00, points 12-16; CA/125/00, points 45-73 and MR/2/00, pages 43-44;

•for the discussions during the EPC 2000 revision conference, see MR/8/00; MR/15/00; MR/16/00 and MR/24/00, pages 69-71

In a Conference Resolution adopted on 29 November 2000, the Diplomatic Conference as regards computer programs states that, "recognising the importance of the issue", the Conference has agreed to "maintain for the present the European Patent Convention's current provisions on software". However, it affirmed an exclusion of data processing and business methods by the inclusion of the phrase "in all fields of technology" borrowed from TRIPS.

Furthermore, the European Parliament rejected an attempt to make the EPO software patent practice legally enforceable throughout Europe following an epic lobbying struggle:

> "The rapporteur underlined that the Parliament wanted to avoid a situation in which all softwares would be patentable. Referring to the current practice in several Member States, the USA and the European Patent office, Mr ROCARD deplored the recent increase in the number of patents granted for softwares and simple processes, which should be covered only by copyright rules. He stressed that, in the long run, the interests of European industry would be better protected under a free-for-all regime than under a maze of patents covering even the most trivial processes and softwares. The plenary adopted the three identical amendments calling for the rejection of the common position by 648 votes in favour, 14 votes against and 18 abstentions."[31]

Irrespective of our dissent with the line of the Council in the directive debate, we endorse that legislative harmonisation within the EU is the applicable harmonisation approach. The unwillingness of the outgoing EU-Commission to propose a new harmonisation proposal which differs from the escapist EPO granting practice is a symptom of the EU democratic deficit and the governance problems under the current European patent regime. The key governance questions on how to defend political primacy, how to keep the EPO in democratic checks and balances and how to avoid administrative capture remain unresolved.

We would like the institutions to stay away from partisan intervention in political debates. However, there are constructive examples of how patent institutions themselves can help to find a proper solution to rule out software patenting. Positive examples include the test suite workshops of the British patent office concerning the effects of definitions of "technical" in software examination or the insightful WIPO Open Forum on Substantive Patent Law. We fail to believe that patent offices are incapable of providing technical

---

31 Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions - Outcome of the European Parliament's second reading (Strasbourg, 4 to 7 July 2005) - 10804/05

assistance for a legal exclusion of software from patentable subject matter that meets market requirements and the spirit of the European Patent Convention.

C. The nature of software

C.1 Patenting the mind?

While software professionals find it somewhat patronising to see patent reasoning applied to their arts, patent experts find it less promising to enter into discussions about the nature of software and algorithms themselves. This division ought to be bridged.

During the Diplomatic Conference concerning the EPC, there appeared to be some ontological confusion, manifesting itself in the problem of translating the term "program for computers"; namely the question as to whether this term had the same scope as "Programme für Datenverarbeitungsanlagen".[32] Although the Diplomatic Conference intended to contain the effects of this exclusion on established patent practice in engineering, it could not be overruled by subsequent expansive institutional interpretations of EPC Article 52(2), so that an exclusion was indeed laid down.[33] For the sake of clarity, the Diplomatic Conference inserted the infamous Article 52(3) "as such"-clause which later allegedly caused confusion. It ought to be closely examined in the context of the whole non-exhaustive list of "mental" exclusions in Article 52(2).

Quite like in reflections on the human mind, you find numerous approaches to thinking about the complex nature of software. We often observe advocates who approach patentability from the software/algorithm/mathematics perspective. Advocates from this "user perspective" easily get trapped within software philosophy and ontological confusion. One of the more recent published publications from this viewpoint was presented by Ben Klemens.[34] Significant for these discussions are proofs as to why software inherently cannot be patented: Because that would equal patenting thought. Software is, indeed, abstract and systematically, software is listed in Article 52(2c) together with other methods which are equivalent to "mental steps". As computer scientists know, Alan Turing started his classic paper.[35] with a human doing calculations (a computer) and outlined how to build a machine that would do the same.

Patent reasoning is quite selective in the application of the Vienna Convention principles of interpretation. It seems fair to doubt that "ordinary meaning" is applicable for the control of a technocratic machinery of examination. Often patent institutions fail to communicate the specific instrumental role of tests in the examination process. Just because a test is named "inventive step" (or "obviousness") it seems inappropriate to address the triviality concerns of practitioners with a general clause filter. And when that is proposed, it serves a political agenda to prevent reforms concerning scope. In the same way, the term "technical character" has to be considered in the view of its precise examination objectives. In that respect dogmatic tools as the German teaching concerning "controllable forces of nature..." involve no philosophic

---

32 BlfPMZ 1976, p. 264 ff.

33 Lenz, Karl F. (2001): Grenzen des Patentwesens, Tokio, Norderstedt, 2002, p. 227ff.

34 Klemens, Ben: Math You Can't Use: patents, copyright, and software, Brookings Institution Press, 2005

35 Turing, Alan (1936): On computable numbers, with an application to the Entscheidungsproblem, Proceedings of the London Mathematical Society, Ser. 2 Vol. 42, 1937

considerations about the essence of technology but serve the objective of drawing a stable line that can be applied by patent practitioners in a predictable way. Alternatives, if there are any, push the frontier indefinitely.

It is helpful to ensure that the engineering problem of patent examination follows an instrumental rationale instead of fairness or justice based on ontological considerations. Software professionals and legal scholars fall into similar traps here. Patent opposition solves the metaphysical problem by relying on easier tests such as novelty, while other tests are "made fit" to achieve the desired results for a decision on the actual case. Arbitrary interpretation paves the way for studious rule-bending in "case law". The premise of uniform "case law" in patent decisions, which also underlies the Referral, needs to be carefully checked. The current corpus of "term jurisprudence" of the Boards of Appeal demonstrates practical difficulties. We doubt that the concept of having institutional "case law" is better than the accountable engineering of examination rules and political governance.

C.2 Defining software

C.2.1 Environment and conventional effects

During the software patent directive debate, it was proposed the following narrow set of definitions with the main objective of preventing misconceptions of legal practitioners:

> A "computer" is a realisation of an abstract machine consisting of entities such as input/output, processor, memory, storage space and interfaces for information exchange with external systems and human users.
>
> "Data processing" is calculation with abstract component entities of computers.
>
> A "computer program" is a data processing solution which can, when described in a predefined language, be executed by computers.

For the purpose of this Referral another set of definitions[36] is put forward with the following special characteristics: a) An execution aspect, b) software (program(s)) narrowly defined as a set of instructions and c) a very broad definition of computers as any programmed apparatus. The definition of a computer program shows a certain similarity to the one found in the USA Copyright Act 17 USC §101 (1988):

> "A "computer program" is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result."

and according to NASA Parametric Cost Estimating Handbook:
> "Software, in general, is a set of programs and accompanying documentation that direct computers to perform desired functions. In

---

36 Referral G03/08, p. 3

simple terms, a software program is a set of instructions for a computer."[37]

In the same spirit American Heritage Dictionary carries an active conception of software:

> "software - The programs, routines, and symbolic languages that <u>control</u> the functioning of the hardware and <u>direct</u> its operation."[38]

All these American definitions highlight that a conventional programs pursues a purpose/result/effect which is missing from the Referral definition. It was expressed by the term "solution" in our definition that is largely rooted in a European perception concerning the processing of data.

C.2.2 Solving a data processing problem

The German standard definition defines a program as a complete solution to a data processing problem:

> "Eine zur Lösung einer Aufgabe vollständige Anweisung an eine Datenverarbeitungsanlage zusammen mit allen erforderlichen Vereinbarungen"[39]

DIN 44300 was rendered obsolete by international standard provisions in ISO/IEC 2382-1:

> "Software: All or part of programs, procedures, rules, and associated documentation of information processing systems.
>
> Data processing: The systematic performance of operations upon data
>
> Data processing system: One or more computers, peripheral equipment, and software that perform data processing."[40]

Note that ISO 2382 distinguishes between data processing and information processing. Information processing includes data processing but also includes data communications and office automation. For us, the standard definitions reflect a professional view which supplements the legal perspective.

C.2.3 Immaterial sphere

The Referral definition describes "software" and "programs for computers" as synonyms. This opinion can be generally supported. From a linguistic perspective, the term "software" is slightly different from "program(s)".[41] The German translation "Programmausstattung" demonstrates that software may also be a set of programs.[42] As a word which lacks a plural, software cannot be

---

37 http://cost.jsc.nasa.gov/pcehhtml/pceh.htm
38 The American Heritage Dictionary of the English Language, Fourth Edition, Houghton Mifflin Company, 2000
39 DIN 44300, 40
40 ISO/IEC 2382-1
41 Mind the idiosyncratic Council quote above which speaks of "softwares"
42 Meritless to distinguish between the program as such and the program as software

decomposed. That makes a term "software" even more precise and comprehensive than the term "program".
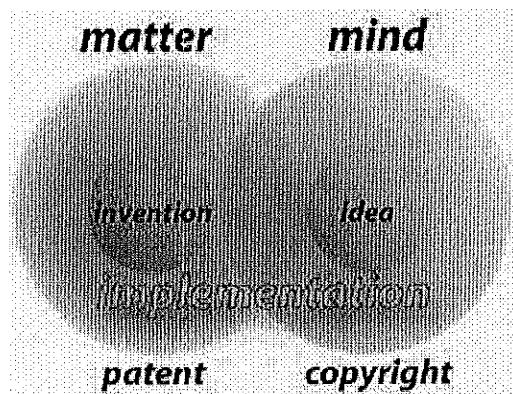
Software is the immaterial sphere and it has its physical counterpart, the hardware. Accordingly the Encyclopaedia of Computer Science proposes a simple negative definition of software:

> "Software is the non-hardware part, including associated documentation, of a system being implemented or implemented in part with a computer or an embedded processor."[43]

The soft-hard dichotomy references to the old separation between mind and matter, information and medium/carrier, or in Descartes' dualism terminology res cogitans and res extensa.

D The relation to software

D.1 Software "as such"

Software is not only excluded from patentability, but it is considered a non-invention. The legal term "software as such" abbreviates the original wording "relates to such (non-inventions) as such". We advise that one is cautious when using abbreviations such as "software as such". The applicable legal wording in EPC Article 52(3) is:

> The provisions of paragraph 2 shall exclude patentability of the [programs for computers, aesthetic creations, algorithms, ..] referred to in that provision only to the extent to which a European patent application or European patent relates to such [programs for computers, aesthetic creations, algorithms or other non-inventions] as such.[44]

Instead of "software as such", it may be more appropriate to speak about software-related and software-unrelated methods while software-unrelated inventions may of course involve the use of software.

The EPC term "program for computers" (German: "Programme für Datenverarbeitungsanlagen") demonstrates that we have to consider the program on a carrier in conjunction with the runtime device for the processing of data as a unit. The non-invention is supposed to run on conventional computers. In the Referral we find "methods" mentioned which can be "implemented" in a computer:

> "For the purposes of this referral, the methods referred to in hypothetical examples are intended to be methods which can be implemented wholly by computer. "[45]

We have demonstrated above that software covers the whole immaterial

---

43 Rosen, Saul: Encyclopedia of Computer Science, Van Nostrand Reinhold, 1992.
44 EPC Article 52(3)
45 Referral G03/08

aspect sphere which runs in a computer. In our case, it seems safe to assume that methods which can be "implemented wholly by a computer" relate to software as such and are prima facie covered by the non-exhaustive list of the European Patent Convention Article 52(2). As we argue above, the legal question to determine if they are patentable is whether these methods relate to software. The patent business's creative reasoning marks the attempt to overcome an exclusion of software-related methods by renaming them computer-implemented inventions, "an invention whose implementation involves the use of a computer, computer network or other programmable apparatus, the invention having one or more features which are realised wholly or partly by means of a computer program."[46] The EPO is fully aware that it grants software patents which cover usual data processing. Computer programs (as such) are explicitly mentioned in the examination guidelines:

> "4.15 Computer programs
>
> In the particular case of inventions in the computer field, program listings in programming languages cannot be relied on as the sole disclosure of the invention. The description,..., should be written substantially in normal language, possibly accompanied by flow diagrams or other aids to understanding, so that the invention may be understood by a person skilled in the art who is deemed not to be a specialist in any specific programming language, but does have general programming skills. Short excerpts from programs written in commonly used programming languages can be accepted if they serve to illustrate <u>an embodiment of the invention</u>."[47]

For an interpretation of Article 52 EPC, according to the applicable legal standard method of interpretation, we refer you to the analysis of Lenz (2001)[48]. As of T1173/97 Lenz argues:

> [T1173/97 finds] "that computer programs are to be divided in the two subsets "software as such" and "other software". This opinion can not be reconciled with the results derived above by a systematic interpretation. The Board does not explain its reasoning but merely derives it from a superficial overview of paragraphs 2 and 3. A use of the common methods of legal interpretation as outlined above is not found in this decision. On the next page the Board asserts that the limitation "as such" has to be understood in the sense that computer programs as such are only computer programs without a technical character. This is as far removed from the wording, that a litigation based on a patent granted under this unlawful interpretation stands opposed to the principle of lawfulness (Art. 104.2 GG). It is a total redefinition if the limitation in paragraph 3 that had nothing in common with the meaning of the law. The Technical Board of Appeal clearly transgresses the boundaries of judicial competence. Whoever wants to replace the wording "as such" by "without a technical character" has to do it by a change of the

---

46 EPO, Patents for Software?, p.3, http://www.epo.org/topics/issues/computer-implemented-inventions/software.html
47 EPO Examination guidelines, 4.14a
48 Lenz, K. F.(2001), Interpretation of Article 52 of the European Patent Convention in view of the question, to what extent software is patentable, translated to English, 2005. Available online by the FFII: http://eupat.ffii.org/analysis/epc52/exeg/epue52exeg.en.pdf

Convention according to the required legislative procedures. Jurisdiction cannot do this."

The phrase "as such" does not have any special meaning in law. It is an intonatory particle, also called appositive. Linguist Hartmut Pilch once explained why: "This is a bit like taking the it in it is raining out of context and asking what mysterious entity this it is that is raining there."[49] König 2001 argues

> "In der Nominalgruppe "Datenverarbeitungsprogramm als solches" bewirkt "als" in Verbindung mit "solche" eine Gleichstellung der Basis (Datenverarbeitungsprogramm), und zwar mit all ihren begriffsbestimmenden Merkmalen, mit sich selbst; diese Gleichstellung ist kontextunabhängig und verweist ohne Wenn und Aber auf die fachlexikalische Bedeutung der Basis. Insoweit kommt dem Appositiv "als solche" hier gleichsam die Qualität einer Verweisung zu. Kontextunabhängig bedeutet im vorliegenden Zusammenhang: jenseits der Kategorien und Kriterien des Patentrechts. Demgemäß meint "als solche" (nur) soviel wie "jeweils für sich" im Sinne einer "Alleinstellung"."[50]

In everyday language the meaning is "as the word is usually understood; in the strict sense of the word."[51] It appears, therefore, that the phrase "as such" is superfluous, and, indeed, it does not appear in the Greek, Polish, Swedish and Swiss patent laws, despite that the subject-matter-related provisions of these laws are essentially local elaborations of the EPC.

D.2 Software methods are programs for computers

Software methods constitute preparatory design works leading to the development of a computer program which are, according to the EU Software Directive[52], in the legal scope of the term "computer program":

> "the term 'computer program` shall include programs in any form, including those which are incorporated into hardware; whereas this term also includes preparatory design work leading to the development of a computer program provided that the nature of the preparatory work is such that a computer program can result from it at a later stage"[53]

An article written by Schölch frames the whole software patenting problem in the context of inappropriate patent grants for software "design methods".[54] On an international level, the WIPO Copyright treaty specifies:

---

49 Pilch, Hartmut: eMail to Jeroen Dekkers and bxl@ffii.org, 24 Oct 2004
50 König, Reimar: Patentfähige Datenverarbeitungsprogramme - ein Widerspruch in sich, in: GRUR, ed. 7, 2001, 577ff.
51 Oxford Dictionary, Oxford, 1989
52 Council Directive of 14 May 1991 on the legal protection of computer programs (91/250/EEC), (OJ L 122, 17.5.1991, p. 42–46)
53 Council Directive of 14 May 1991 on the legal protection of computer programs (91/250/EEC), (OJ L 122, 17.5.1991, p. 42–46)
54 Schölch, Günther: Patentschutz für computergestützte Entwurfsmethoden - ein Kulturbruch? in: GRUR 2006, 969ff.

"Computer programs are protected as literary works within the meaning of Article 2 of the Berne Convention. Such protection applies to computer programs, whatever may be the mode or form of their expression."[55]

The European Software Directive implements the aforementioned WIPO Copyright Treaty where "the term 'computer program` shall include programs in any form, including those which are incorporated into hardware; whereas this term also includes preparatory design work leading to the development of a computer program provided that the nature of the preparatory work is such that a computer program can result from it at a later stage" and is is clarified that the "function of a computer program is to communicate and work together with other components of a computer system and with users and, for this purpose, a logical and, where appropriate, physical interconnection and interaction is required to permit all elements of software and hardware to work with other software and hardware and with users in all the ways in which they are intended to function". The remarkable aspect for our analysis is not the actual scope of copyright protection under the software directive, but the described interactive nature of software that cannot be separated from its runtime-environment and medium.

Hypothetical examples of "inventions" that can be "implemented" in "software and hardware" demonstrate how they might relate to software. Software can also be embodied in hardware. Software, under the software directive and other copyright laws, has all the essential properties the software patent reasoning of the institutions denies for their emasculated fiction of a "software as such".

D.3 Access right to software methods

The European Software Directive Article 5(3) specifies what can explicitly be carried out legally by a software user, namely the right to examine the software methods (here: "ideas and principle") from the study of the running software:
> "The person having a right to use a copy of a computer program shall be entitled, without the authorization of the rightholder, to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program if he does so while performing any of the acts of loading, displaying, running, transmitting or storing the program which he is entitled to do. "[56]

It is implied by the European Software Directive that an inspection of software methods is desirable and does not even require permission of right holders. Also the European Patent Office is entitled to study available software works to determine and document the state of the art concerning software-related methods. Concerning source code, which, for software professionals, disclose the most, the EPO admits:

> "It is not the policy of the EPO to require or examine source codes [...]. Moreover, given the length and complexity of source code listings, which

55 WIPO WCT, Article 4
56 Council Directive of 14 May 1991 on the legal protection of computer programs
    (91/2 /EEC) (OJ L 122, p. 42–46)

can often stretch to hundreds of pages, it would be quite impossible to examine them."[57]

Consequently, software developers are forced to search all available patent applications to avoid a potential independent "reinvention" of patented software-related methods by their works. Indeed, independent "reinvention" is a typical standard infringement scenario. Developers cannot obtain a certificate of non-infringement for their works that would give them full legal certainty that their works do not infringe a patent. Their patent agent is not liable for a unreliable search because patents are not well-defined rights. Additional uncertainties arise from equivalence.

D.4 Services and sectoral considerations

As an outspoken proponent of patent reforms in Europe, we suggest that a diffusion of patent law application from the second to the third sector be monitored closely. Typical service patents relate to software and business methods. There we find a collusion of the patent system with general commercial and intellectual activities for which freedom of action is considered as a fundamental right.

On the second page, the Referral document claims software was a product. But software is not considered a 'product' under private law but a service, related to the tertiary sector of the economy. The box packaging of software evoking the impression of a product is losing relevance and has never been more than a marketing concept. Naturally a "tertiarisation" of the economy as a whole at the expense of the industrial sector also involves a trend in which the classical realm of patent law would lose relevance or at least need to be reconsidered. An example where software substitutes classical hardware is software-defined radio. Yet, a market shift towards pure software solutions is no invitation to expand the inherited patent regimes of classic engineering to data processing. Rather the expectation is either to do as the Romans do or to create a new social contract for software patenting.

Even if you agree with service patenting, the established patent system design fails to suit the specific needs of service markets for historical reasons. It was set up for classical engineering. The rough criteria of "industrial applicability" expressed an underlying intention to keep the service sector out of the realm of patent law. Since industrial application alone fails to draw a line to business method patenting, the term "invention" was included under the heading of "field of technology"[58] and the term "technical" became a feature of the term invention (or reinstatement as critics argue).

The EPC 2000 reform added a limitation to inventions in "fields of technology". The phrase "in all fields of technology" was borrowed by the Diplomatic Conference from the TRIPS Article 27 non-discrimination clause.[59] TRIPS is an

57 EPO: Patents for Software?, http://www.epo.org/topics/issues/computer-implemented-inventions/software.html
58 Schrader, Tobias: Technizität im Patentrecht, Berlin: Heymanns, 2007, p. 53f.
59 „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. Subject to paragraph 4 of Article 65, paragraph 8 of Article 70 and

annex to the GATT agreement from the Uruguay round and enforcement subject to WTO trade dispute settlement mechanism. The WTO Doha Declaration clarified that "each provision of the TRIPS Agreement shall be read in the light of the object and purpose of the Agreement as expressed, in particular, in its objectives and principles."[60] Nevertheless, TRIPS treaty Article 27 has been used by lobbyists to persuade legislators to expand the scope of patentable subject matter.

TRIPS Article 27 is a regulatory non-discrimination trade clause while the same provision in the EPC 2000 serves as a limitation of patentable inventions and affirmation of the status quo. In the EPC 2000 systematic context, it reflects the intention of returning to a rough general services exclusion. The provision clarifies that inventions[61] which are not in fields of technology are generally not eligible to a patent under Article 52 EPC, but Article 53 makes other subject matter patentable. Inclusion in a field of technology does not suffice in defining the term invention.

What is a "field of technology"? According to Duden the term "Technik" is defined as "alle Maßnahmen, Einrichtungen u. Verfahren, die dazu dienen, die Erkenntnisse der Naturwissenschaften für den Menschen praktisch nutzbar zu machen: die moderne T.; ein Wunder der T.; auf dem neuesten Stand der T. "with "Erkenntnisse" being "durch geistige Verarbeitung von Eindrücken u. Erfahrungen gewonnene Einsicht ".

Data processing does not reside in a "field of technology". It is affirmed by the non-invention concept of EPC Article 52(2). The European Parliament insisted data processing was no field of technology. And it is perceived as a flexibility among members of the German Parliament:

> "Das TRIPS gibt zwar den Rahmen vor, lässt aber noch erheblichen Auslegungsspielraum. So ergibt sich insbesondere aus dem TRIPS keine Definition zu der wichtigen Voraussetzung, was unter dem „Gebiet der Technik" zu verstehen ist. Auch das „Übereinkommen über die Erteilung Europäischer Patente" (EPÜ) kennt keine genaue Definition des Technikbegriffs, legt aber in Artikel 52 EPÜ fest, dass keine Patente für Erfindungen erteilt werden dürfen, die lediglich „mathematische Methoden" und „Programme für Datenverarbeitungsanlagen" zum Inhalt haben."[62]

In Germany the Bundespatentgericht viewed TRIPS as an affirmation of the German practice to consider technicity as a central aspect of the inventions:

> „[TRIPS] führt zu keiner anderen Beurteilung der Patentfähigkeit. Abgesehen von der Frage, in welcher Form das TRIPS-Abkommen - unmittelbar oder mittelbar - anwendbar ist ..., würde nämlich auch die

---

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.

60 WTO Doha Declaration

61 We stressed that software-related methods are not inventions under the EPC 52. For this reason a limitation to inventions in a fields of technology would have no effect.

62 Bundestag Drucksache 15/3941

Heranziehung von Art 27 Abs 1 TRIPS-Abkommen hier nicht zu einem weitergehenden Schutz führen. Mit der dortigen Formulierung, wonach Patente für Erfindungen auf allen Gebieten der Technik erhältlich sein sollen, wird nämlich im Grunde nur die bisher schon im deutschen Patentrecht vorherrschende Auffassung bestätigt, wonach der Begriff der Technik das einzig brauchbare Kriterium für die Abgrenzung von Erfindungen gegenüber andersartigen geistigen Leistungen, mithin die Technizität Voraussetzung für die Patentfähigkeit ist (in der Entscheidung des BGH ‚Logikverifikation' ist insoweit die Rede von ‚nachträglicher Bestätigung' der Rechtsprechung durch die Regelung in Art 27 Abs 1 TRIPS-Abkommen). Auch der Ausschlusstatbestand des § 1 Abs. 2 Nr. 3 und Abs 3 PatG kann vor dem Hintergrund, dass er auf dem Gedanken des fehlenden technischen Charakters dieser Gegenstände beruht, nicht im Widerspruch zu Art 27 Abs 1 TRIPS-Abkommen gesehen werden."[63]

For the comprehension of the non-inventions under Article 52(2), the criteria "field of technology" is not relevant apart from patentable inventions under the Article 52(3) exemption. The debate on how patent law can adapt to the market conditions of service markets remains unresolved. Rule 27(1) EPC implementation requires the patentee to "specify the technical field to which the invention relates". That could lead to a simple solution to specify fields for examiners which are prima facie not technical, for example, the field of "data processing". It needs to be assessed whether a general field of technology test could simplify examination and draw the line in a comprehensible and non-arbitrary way, so that examination capacity is better allocated.

---

[63] Bundespatentgericht 17 W (pat) 69/98 "Suche fehlerhafter Zeichenketten" , 1.4.6

E Referred Questions

E.1 Question 1 - Polyphem blinded by "no man"

CAN A COMPUTER PROGRAM ONLY BE EXCLUDED AS A COMPUTER PROGRAM AS SUCH IF IT IS EXPLICITLY CLAIMED AS A COMPUTER PROGRAM?
The Referral mentions why that cannot be the case:

"...the substance of the claims... is often identical".[64]

In its disposition program decision, the BGH[65] stressed the principle that not a verbal dressing but the substance matters. Common sense indeed, popular lore has it: "Calling a chair a table does not make it any less a chair."

While it remains to be debated whether an equivalence interpretation is to be favoured over a strict literal interpretation of patent claims in an enforcement context, "substance over form" is indispensable for the application of the law which sets conditions and exclusions for the grant of rights.

It was highlighted above that the term "computer program as such" lacks legal precision concerning the legal exclusion of software-related methods under the Article EPC 52. As suggested above, it would be more advisable to generally frame the problem in terms of relation to software contrary to the T1173/97 interpretation.

We can put any interpretation to a simple test if software works when run on a general purpose computer or electronic commerce applications, as a web shop can infringe a patent. Indeed, such claims are being granted irrespective of the exclusion. It seems hard to argue that you can claim a computer program but that the patent does not relate to such matter. It seems even harder to defend the possibility of infringing a patent solely by conventional use of a computer program, but the patent would not relate to a computer program.

According to the bizarre decision T424/03 a "computer-implemented method" can never be "identical" with a computer program as such, the claim category of a computer-implemented method is distinguished from that of a computer program. However, the patent would still relate to software as such.

It is undisputed that a program work sent to the patent office would have to be refused on formal grounds as would a mouse trap or a machine when submitted in a post packet to the patent office. Because Article 52(3) speaks of a "patent" related to the software as such, we have to assume that these formal requirements are beyond the scope of the provision. As a result, the statutory exclusion is not to be understood to apply to "computer programs as works" but "methods related to software (as such)".

Common sense advises against the sophistry that was introduced by T1173/97 concerning a "software as such" subset.[66] T424/03 interprets that term in a narrow way that demonstrates why T1173/97 erred with its interpretation.

---

64 Referral G03/08, p. 4
65 BGH 28 X ZB 23/74, 1976 - Dispositionsprogramm
66 EPC Article 52(3)

E.2 Question 2 - Escape underneath the sheep

(A) CAN A CLAIM IN THE AREA OF COMPUTER PROGRAMS AVOID EXCLUSION UNDER ART. 52(2)(c) AND (3) MERELY BY EXPLICITLY MENTIONING THE USE OF A COMPUTER OR A COMPUTER-READABLE DATA STORAGE MEDIUM?

Of course an exclusion cannot be avoided merely by mentioning the use of a computer or a computer-readable storage medium, otherwise we would find patent examination here put in the position of the Cyclops Polyphem who, blinded by "no one", lets his perpetrators escape under the skin of sheeps. Lord Justice Dillon did chose his words with care:

> "It would be nonsense for the Act to forbid the patenting of a computer program, and yet permit the patenting of a floppy disc containing a computer program, or an ordinary computer when programmed with the program; it can well be said, as it seems to me, that a patent for a computer when programmed or for the disc containing the program is no more than a patent for the program as such."[67]

All programs for data processing machines are not conceivable without their medium and a runtime environment, a general purpose computer. Indeed, we have both algorithms and programs for computers covered by the EPC Article 52(2) list of non-inventions. Both the medium and the runtime environment serve general purpose functions for the conventional performance of the software works. This applies in particular when we consider the exclusion of software-related methods. Software, like all data, is indifferent from its medium type and the medium is neutral towards the stored software content. As for internet communication networks, we refer you to the mere conduit principle[68] found in the E-Commerce Directive, the neutrality of the communication service provider concerning the transmitted data. In a similar fashion, the computer is neutral towards the performed software.

If the patent was unrelated to software how, then, can a patent be infringed by software (where the actual relevant solution is found) ordinarily performed?

In this regard we recall the preliminary findings in the Amazon One-Click patent opposition case:
> "5) Patentable invention, Article 52(2) and (3) EPC.
>
> The actual practice of the EPO concerning computer implemented invention leads to consider subject-matter as non patentable only if it does not include a single technical feature. In the present case, <u>at least a computer system is included in the granted subject matter.</u> Therefore, the Opposition Division is of the opinion that the granted claims fulfill the requirements of Article 52(2) and (3)."[69]

---

67 Genentech Inc's Patent (UK) [1989] RPC 147

68 Directive 2000/31/EC of the European Parliament and of the Council of 8 June 2000 on certain legal aspects of information society services, in particular electronic commerce, in the Internal Market ('Directive on electronic commerce') (OJ L 178 , P 1-16), Article 12

69 http://www.digitalmajority.org/forum/t-23571/ffii-vs-amazon:add-a-computer-and-the-epo-will-give-you-software-patents

.

Cases in administrative practice like these clearly demonstrate the need to return to an application of the "core theory". While we disagree with the substance, the quote demonstrates that a bulk inclusion of "software inventions" is applicable as a simple rule for patent practitioners: "At least a computer system is included". In the spirit of the EPC, where software-related methods are considered to be non-inventions, it is an applicable rule to assume applications that stay in the realm of data processing and are carried out in software are prima facie non-inventions. It would be still possible to show that an application is nevertheless an invention, although it involves the use of a non-invention subject matter.

(B) IF QUESTION 2 (A) IS ANSWERED IN THE NEGATIVE, IS A FURTHER TECHNICAL EFFECT NECESSARY TO AVOID EXCLUSION, SAID EFFECT GOING BEYOND THOSE EFFECTS INHERENT IN THE USE OF A COMPUTER OR DATA STORAGE MEDIUM TO RESPECTIVELY EXECUTE OR STORE A COMPUTER PROGRAM?

Lenz highlighted the notion of a technical character of an invention is customary but the "further technical effect" lacks a legal base.[70] Under the T1173/97 interpretation of "as such" it is made provision for the distinction of computer programs into those that have a "further technical effect beyond the normal physical interactions between the program and the computer", and into those that do not have such a further technical effect. The latter are considered to be "as such", whereas the former are considered to be patentable subject matter. In this way, virtually all software is patentable, as all that is necessary is that the applicant claims some kind of further technical effect, a task not particularly difficult, as the word "technical" remains undefined. The very essence of the "further technical effects" is that these may be not technical in the classic understanding.

The EU Software Directive explains the "function of a computer program is to communicate and work together with other components of a computer system and with users and, for this purpose, a logical and, where appropriate, physical interconnection and interaction is required to permit all elements of software and hardware to work with other software and hardware and with users in all the ways in which they are intended to function".[71]

It was acknowledged by the EPO that the notion of a "further technical effect" of T1173/97 was a preliminary solution for admitting software patenting, but in light of the expected deletion of the exclusion by the Diplomatic Conference, the EPO did not consider it necessary to apply it to the examination practice:

> "This scheme [for patenting software as "computer-implemented inventions"] makes no mention of the "further technical effect" discussed in T1173/97. There is no need to consider this concept in examination, and it is preferred not to do so for the following reasons: firstly, it is confusing to both examiners and applicants; secondly, the only apparent reason for distinguishing "technical effect" from "further technical effect" in the decision was because of the presence of "programs for computers"

70 Lenz, Karl F. (2001): Grenzen des Patentwesens, Tokio, Norderstedt, 2002, p. 227ff.
71 Council Directive of 14 May 1991 on the legal protection of computer programs (91/250/EEC), (OJ L 122, 17.5.1991, p. 42–46)

in the list of exclusions under Article 52(2) EPC. If, as is to be anticipated, this element is dropped from the list by the Diplomatic Conference, there will no longer be any basis for such a distinction. It is to be inferred that the BoA would have preferred to be able to say that no computer-implemented invention is excluded from patentability by the provisions of Articles 52(2) and (3) EPC. As for the lack of need to consider "further technical effect", this assertion is based on the proposition that, according to the scheme put forward, no patent would be granted which should have been refused for lack of further technical effect. This is because the existence of an objective technical problem overcome is itself sufficient proof of the requisite further technical effect. Further, it is to be remarked that this scheme of examination should not lead to refusals where previously a patent would have been granted, since the requirement for an objective technical problem is long-established. The only change is an explicit statement of the already implicit consequences of the lack of such a problem. Examiners should however be familiar with the concept of "further technical effect", since it may be employed by applicants or by parties in an opposition. "[72]

As the T1173/97 is not to be applied in examination for practical reasons and serves merely the political agenda of making software-related methods patentable, it would be inappropriate for a Opinion to endorse the decision.

E.3 Question 3 - Effects on the real word

(A) MUST A CLAIMED FEATURE CAUSE A TECHNICAL EFFECT ON A PHYSICAL ENTITY IN THE REAL WORLD IN ORDER TO CONTRIBUTE TO THE TECHNICAL CHARACTER OF THE CLAIM?

The question seems to be derived from the T1173/97 reasoning we criticised above, and the consideration to what extent terms as technical, a reinstatement of the invention problem according to some analysts, should be loaded to define the limitations of the patent domain in general, in particular when there is no uniform consent about the practical scope and definition.

For pragmatic purposes we endorse the German tradition of "technical" relating to "controllable forces of nature" (to be contrasted with uncontrollable fundamental forces) or framed in terms of applied natural science.

The core fallacy of T1173/97 is that just because the items listed in the non-exhaustive list of exclusions which concerns mental operations (algorithm, software, organisational rules, ...) lack a technical character they would be only to be excluded "as such" but patentable when they have technical features. Non sequitur. The list of exclusions that is non-technical could also be framed in terms of modal properties. Sometimes "non-technical" in the terminology of patent reasoning just means "not patentable" in respect of a rather generalised concept like technical programs should be patentable and non-technical programs not patentable. After all the exclusion in EPC Article 52(2)

---

72 Note of the EPO-President to the Trilateral conference, Appendix 6: Examination of "business method" applications
(EPO), 19.5.2000, retrieved from http://www.european-patent-office.org/tws/appendix6.pdf (not available anymore)

applies irrespective of technical effects, character or considerations to all software.

It is apparent that past reasoning from the EPO mainly redresses political objectives:

"There will undoubtedly continue to be debate as to what constitutes a technical problem and what does not. This is exactly the same debate as we had under the "technical contribution" scheme, we have merely transferred it to a different stage of the examination. The bonus is that we [at the EPO] can still use the decisions of the BoA, whether or not they used the contribution approach, as guidance as to what is considered to be technical. "[73]

(B) IF (A) IS ANSWERED IN THE POSITIVE, IS IT SUFFICIENT THAT THE PHYSICAL ENTITY BE AN UNSPECIFIED COMPUTER?

Insufficient. An unspecified computer would be general purpose equipment for the conventional performance of the data processing instructions. An instruction for the use of a universal purpose computer system should not be considered patentable irrespective of the physical or technical nature. The general purpose computer defines the semantic space for the performance of the software in the same way a piano defines a space for musical expression.

Under the current EPO practice it is common to construct a technical effect in the display of the computer or the cathode ray of a monitor, scattered precedence was glued into examination guidelines. Such a technical effect is a mere formality for a circumvention of the exclusion.

The use of the technical means needs to be an essential part of the solution. A claimed object that consists of steps for the use of generic universal computer and peripherals, also called "program for computers" or "computer-implemented solution" or software-related method, is not an invention in the sense of patent law, regardless of the form in which it is claimed. A claimed subject should be an invention in the sense of patent law only if it contributes knowledge to the state of the art in a field of applied natural science.

(C) IF (A) IS ANSWERED IN THE NEGATIVE, CAN FEATURES CONTRIBUTE TO THE TECHNICAL CHARACTER OF THE CLAIM IF THE ONLY EFFECTS TO WHICH THEY CONTRIBUTE ARE INDEPENDENT OF ANY PARTICULAR HARDWARE THAT MAY BE USED?

Irrelevant.

E.4 Question 4 - The activity of programming

(A) DOES THE ACTIVITY OF PROGRAMMING A COMPUTER NECESSARILY INVOLVE TECHNICAL CONSIDERATIONS?

No, at least not necessarily. The term programming has no special meaning or relevance in patent law. Technical considerations assume an applied use of

---

[73] Note of the EPO-President to the Trilateral conference, Appendix 6: Examination of "business method" applications

(EPO), 19.5.2000, retrieved from http://www.european-patent-office.org/tws/appendix6.pdf (not available anymore)

controllable forces of nature without mental intermediation. Thus the mental process of programming carried out as a human activity is beyond the definition.

Following T 769/92 (OJ 8/1995, 525), the requirement for technical character may be satisfied if technical considerations are required to carry out the invention. Such technical considerations must be reflected in the claimed subject-matter. However, programming cannot be considered as "carrying out" inventions or inventing.

With respect to
> "The effects caused by a computer program (which may or may not contribute to its technical character) may occur when the program is executed (for instance how much memory it occupies, how quickly it carries out the tasks for which it was programmed, ' etc.). On the other hand, there may be <u>effects relating to software development</u> which affect the programmer in his work (ease of maintenance of the program, flexibility, portability, reusability etc.). "[74]

we mention that neither of these problems are technical in the sense of patent law nor do they involve the use of controllable forces of nature. The grounds for comparison lies within the sphere of data processing:
•"less occupation of memory" relates to the occupation of existing memory by the program. The available memory of the conventional computer remains unaffected.
•"how quickly" relates to the performance of the data processing itself, the time to solve a data processing problem. The actual speed of the conventional computer remains unaffected.
These examples even relate to the level of an algorithmic optimizations and are applied in the software for a data processing problem. Other aspects mentioned above are software management and maintenance or convenience considerations.

With the term "effects relating to software development" we face the confusing notion of this question 4: the process of creation of software works and considerations by the author. Dancing for instance usually involves technical and physical considerations. What people consider when they dance or program is subject to mental inspection. For sure, patent institutions do not mind what programmers actually think and consider but would present their own legal fiction with the underlying political agenda to further expand patentable subject matter. A "considerations" approach warned Kraßer would lead to unlimited patentability.[75]

(B) IF (A) IS ANSWERED IN THE POSITIVE, DO ALL FEATURES RESULTING FROM PROGRAMMING THUS CONTRIBUTE TO THE TECHNICAL CHARACTER OF A CLAIM?

Irrelevant.
(C) IF (A) IS ANSWERED IN THE NEGATIVE, CAN FEATURES RESULTING FROM PROGRAMMING CONTRIBUTE TO THE TECHNICAL CHARACTER OF A CLAIM ONLY

---

74 Referral G03/08, p. 12
75 Kraßer, Rudolf: Erweiterung des patentrechtlichen Erfindungsbegriffes?, GRUR 2001, p. 959-965.

WHEN THEY CONTRIBUTE TO A FURTHER TECHNICAL EFFECT WHEN THE PROGRAM IS EXECUTED?

It is understood that question c) aims at the term "Erfinderische Tätigkeit" (inventive step) to add another interpretation path based on legal fiction. The hidden objective is to parallel inventing and programming with a view to the German phrasing of EPC Article 56.

Programming is the human process for the creation of a software work. Division of labour concerning the development process is common and also a matter of development methodology fashions and management techniques. The abstraction level of the data processing is irrelevant. Which development management methodologies were used, how many persons participated and what skills these individuals contributed is irrelevant for the resulting work. For software protection only aspects that are work-immanent are relevant. The same standards should apply to inventions.

We are concerned that any "further technical effect" test ultimately follows the political objective to undermine the substantive exclusion of software from patentable subject matter. We advise to watch the practical implications of the "further technical effect" teaching which we find absolutely inappropriate and legally unfounded:

> "A further technical effect which lends technical character to a computer program may be found e.g. in the control of an industrial process or <u>in processing data</u> which <u>represent physical entities</u> or in the <u>internal functioning of the computer</u> itself or its interfaces under the influence of the program and could, for example, affect the efficiency or security of a process, the management of computer resources required or the <u>rate of data transfer</u> in a communication link."[76]

There is no grounds for that in the EPC. A data rate e.g. means "faster" processing of data (data per time). If the effect is solely achieved by software there is no technical effect but an optimization of data processing.

As a final conclusion, it does not matter what despicable circumvention teachings the patent office practice and applicants come up with. The substance of the legal provisions matters. It would be better to have a political governance of examination than administrative "case law" which follows a political or institutional bias to overcome the legislative constraints for the EPO in order to patent software and business methods. Whether software should be patented or be shielded from patents remains first and foremost a political struggle. No matter what is "technical" in patent law, patenting of computer programs is no technicality that should be left to administrative decision making.

---

[76] EPO examination guidelines, 2.3.6 "Programs for computers",
    http://www.epo.org/patents/law/legal-texts/html/guiex/e/c_iv_2_3_6.htm