Hartmut Pilch's Speech at the Europarl Hearing of 2002-11-07

http://swpat.ffii.org/events/2002/europarl11/cusku/index.en.html

Workgroup
swpatag@ffii.org

2002-12-10

Hartmut was invited speak at a hearing of the European Parliament's Committee for Legal Affairs and the Internal Market (JURI) about the European Commission's software patentability directive proposal. He delivered the speech in an improvised manner based on presentation slides. The following written version was prepared later for publication on the Parliament's website.

Contents

1 EuroLinux Alliance – Largest Software Innovation Network in Europe

Dear Ms McCarthy, Dear Members of the European Parliament, Dear Ladies and Gentlemen, Thank you for inviting me to this hearing.

1

I have come here as a representative of the Eurolinux Alliance, which is supported by 300 Software Publishers, who develop mostly commercial software, half of it free and half proprietary, half of it running on Linux and half on other operating systems such as MacOS and MS Windows. Among our constituents are 500 paid software researchers from the private sector, 1,000 corporate executives, 7,000 Software Consultants, 128,000 individual supporters and 12 Associations in 8 countries.

We have also been the leaders in an informed public discussion on software patents. Try to search for keywords in Google, and our sites will usually be near the top. The Google ratings reflect the degree to which our sites are accepted as a reference in public discussions, including academic research in software economics. You will find that the public debate has been completely won by opponents of software patents. This is also reflected in the long list of prominent signatories under the Call for Action, which we have handed out to you as a printed documentation.

2 EuroLinux Companies – A Wider Product Range than Microsoft

MS WinNT	Linux (FI), Trolltech (NO), KDE (DE), SuSE (DE), Mandrake (FR) etc
MS IE	Opera (NO), iCab (DE), Konqueror (DE) etc
MS Office	KOffice (FR), StarOffice (DE) etc
MS SQL Server	MySQL (SE), Frontbase (DK) etc
MS Navision	ERP5 (FR), Enterprise Java Beans (FR) etc
Softimage	Blender (NL) etc

Many of these are SMEs who live on copyright licenses and are innovation leaders in their fields. Perhaps the best known is Opera Software, authors of a small and fast web browser and of much of the software which european telecommunication companies use in their mobile phones. We have brought a letter from Opera Software[1], in which they explain to you why patents are bad for their business and why they support our Counter-Proposal.

The debate is raging on in the USA as well. At the FTC hearings, many representatives of small and large companies complained that patents in the software field stifle innovation, and others stated that there is a "general animosity against software patents". The list of companies who have written long and clear statements against software patents comprises such names as Adobe, Oracle, Autodesk, CISCO, ... You can find some of their statements in Annex F (Quotations[2]) of the documentation which we handed out. It may be interesting to note that among the big american software

---

[1]opera-letter.pdf
[2]http://swpat.ffii.org/archive/quotes/index.en.html

companies, Microsoft was the only one to support software patentability at the hearings in 1994.

Surveys which have been conducted in the 1960s, 1990s and more recently all show the same picture: 80% of software salesmen are against software patents, 80% of patent salesmen are pro software patents.

Eicta and UNICE represent the patent salesemen, not the software salesmen.

To take the previous speaker, Mr. Hagedorn, as an example: Mr. Hagedorn is a patent salesman, not a software salesman. The top software salesman of his company, SAP's CEO Professor Hasso Plattner, has said publicly that SAP's business model does not depend on patents and that SAP is merely collecting them for defending its position in the USA, where patents have become important bargaining chips. SAP grew big without applying for a single patent. Only in 1997 they began worrying about other companies' patents and reacted by hiring Mr. Hagedorn. If we look at EICTA, we find a similar pattern. EICTA's german member organisation, Bitkom, did not have a position on software patents for a long time. Finally, when asked by governments to react, the patent lawyers of their large corporate members formulated their position from opinions which came entirely from the patent community in a language which is understood only by the patent community. Nobody from the software community participated in the decisions. Bitkom's recent decision to support the Directive Proposal was taken by a closed meeting of an IP working group which was attended by 7 patent lawyers of large companies, among them Mr. Hagedorn, and one representative of an SME, Phaidros AG from Germany, a writer of proprietary business modelling software which is an active supporter of the EuroLinux Alliance. The resulting Bitkom position paper sounded just like Mr. Hagedorn's speech you just heard, and the opinion of Phaidros AG was completely disregarded, although various studies, including the governmental study whose results Mr. Hagedorn just misrepresented here[3], clearly demonstrate that most member companies share the interests of Phaidros in being able to go on with their conventional business model which relies on copyright and not on patents.

Many economic studies have been conducted. All of them, even those conducted by the patent lobby, including the so-called economic impact study which the European Commission ordered from Mr. Hart and two colleagues in 2000, end up showing that patents are not a stimulus but an impediment to innovation in the field of software.

Appendix C (Software Patents in Action[4]) contains a series of examples which show that the 30000 patents on rules of organisation and calculation, which the European Patent Office has granted against the letter and spirit of the written law[5], are not an empty threat. Patents are already removing software from the net, traumatising software developers and taking steam out of software innovation.

Dear Ladies and Gentlemen, Please read our response to your rapporteur's working papers[6] from June in Appendix E. When looking at the statements which have emanated from the European Parliament so far, we feel that you are downplaying the problem and

---

[3] http://swpat.ffii.org/papers/bmwi-fhgmpi01/index.en.html
[4] http://swpat.ffii.org/patents/effects/index.en.html
[5] http://swpat.ffii.org/patents/index.en.html
[6] http://swpat.ffii.org/papers/eubsa-swpat0202/amccarthy0206/index.en.html

dismissing our concerns before even seriously examining them. Why are you doing this to us?

Please look again at the table of Microsoft software vs EuroLinux software. We are the European software creators. We are the vibrant, innovative European IT innovation landscape which you politicians have always been talking about. We are the European alternative to large US-dominated structures which you politicians have, often in vain, tried to foster. So why are you now planning to kill us?

3 The Directive Proposal – No Limits, No Clarification, Less Intellectual Property

The directive proposal of the European Commission seems to pursue two conflicting goals: (1) to make patents available to software innovators and (2) to ensure that patentability is limited to "technical contributions".

These goals conflict due to what some have called the Paradox of Abstraction[7]. The art of software innovation lies in abstraction. About 95% of the software patents granted by the EPO are relatively concrete and consequently trivial ideas of putting a computer to work for a certain purpose. The other 5% pertain to relatively sophisticated ideas in the field of higher mathematics. Some easy-to understand examples of typical EPO software patents[8] can be found in Appendix B of the documentation. By applying a test of "technical contribution", the European Commission is apparently proposing that in particular the more trivial software ideas should be patentable. It is proposing to legalise the granting of trivial patents on the use of the Universal Computer for certain concrete purposes, which are described in patent claims.

But if you look closer at the proposal, you will find that the talk about "technical contribution" may in fact have no meaning at all. In the end, a patent lawyer will be able to claim both the most trivial and the most abstract software ideas. Once the European Parliament has given its blessing to this directive proposal, there will be no longer a legal basis for denying the patentability of business methods, mathematical methods and all other rules of organisation and calculation, as long they are claimed as "computer-implemented inventions" along the lines of this directive.

There is one point where we agree with the European Commission: the exclusion of text claims, i.e. claims that read something like

> computer program on a data carrier, characterised by that ... [upon reading it into memory the process according to claim 1 is executed]

. It is important for systematic reasons to disallow text claims. Program texts are like patent descriptions: they describe a problem and a solution in a formal language. You wouldn't want to prohibit the publication of patent descriptions, would you? The freedom of publication is guaranteed by Art 10 of the European Convention for the Protection of Human Rights, and it has become clear that the freedom to publish source

---

[7] http://swpat.ffii.org/papers/ist-tamai98/index.en.html
[8] http://swpat.ffii.org/patents/samples/index.en.html

code is more important than ever today. Patent claims cannot constitute a basis on which the freedom of publication may be denied. Such an idea is simply logical nonsense and an outrageous assault on human rights. Independently of this directive we feel there is a need to state this clearly and to reinforce the freedom of publication especially with regard to computer programs in the upcoming European Constitution.

Even though the European Commission's proposal disallows text claims, it does not really protect the freedom of publication. The Explanatory Memorandum even fails to mention this freedom. It seems more concerned about about the freedom of the European Patent Office to set the rules of patentability. It attaches a paramount value to what it calls the "status quo of law", by which it means EPO caselaw. It is also very concerned about the rights of patent owners, and in particular the rights of the owners of 30000 illegal patents on computer-implemented rules of organisation and calculation. It pretends to want some kind of limit, but leaves this limit deliberately unclear and ineffective.

The result of passing this directive will be that programmers are less free to innovate and that their copyright property will lose much of its value. When you write a program, you will constantly have to worry that you may be infringing on thousands of patents. When you are infringing, you will have to withdraw your program from the Net. This is a traumatic experience — a deprivation of freedom and intellectual property. The proposed aim of the directive is legal certainty for owners of software patents, and there is no reason to doubt that this aim will achieved. This however comes at the cost of systematic legal uncertainty for software developpers and a decrease in the production of software, and a strengthening of monopolistic structures. Only people who do not write software, such as patent lawyers, patent officials, and corporate patent departments, will benefit and prosper. By putting its stamp under this directive, the European Parliament would burden itself with a special political responsibility. In other places in the world, where the patent system has also gone mad, it has done this on its own, without shifting the burden of responsibility to the parliament.

4 Counter-Proposal: Clear and Adequate Software Property Rules

In principle, it is a good idea for the European Parliament to set the rules of patentability. The European Parliament, not the European Patent Office, should be entitled to discuss and develop such rules, and it should do it with the necessary self-esteem and seriosity. Merely putting a Parliament's rubberstamp on the work of a patent office is worse than not legislating at all.

The objective of the legislation should be a clear and reliable distinction of what is patentable and what not. Roughly speaking, the dividing line should be something like the following:

> *A new machining tool is patentable, but the software which may be used to control it is not. Thus production with the tool will infringe, while the*

distribution of software or its use for simulation will not.

Or, to say it in a somewhat more obscure language, that of the TRIPs treaty:

Electronic Data Processing is not a field of technology in the sense of patent law.

Or, in the words of the German Federal Court of Justice (BGH) in its famous Dispositionsprogramm decision of 1976:

Stets ist aber die planmäßige Benutzung beherrschbarer Naturkräfte als unabdingbare Voraussetzung für die Bejahung des technischen Charakters einer Erfindung bezeichnet worden. Wie dargelegt, würde die Einbeziehung menschlicher Verstandeskräfte als solcher in den Kreis der Naturkräfte, deren Benutzung zur Schaffung einer Neuerung den technischen Charakter derselben begründen, zur Folge haben, dass schlechthin allen Ergebnissen menschlicher Gedankentätigkeit, sofern sie nur eine Anweisung zum planmäßigen Handeln darstellen und kausal übersehbar sind, technische Bedeutung zugesprochen werden müsste. Damit würde aber der Begriff des Technischen praktisch aufgegeben, würde Leistungen der menschlichen Verstandestätigkeit der Schutz des Patentrechts eröffnet, deren Wesen und Begrenzung nicht zu erkennen und übersehen ist.

However in all cases the plan-conformant utilisation of controllable forces of nature has been named as an essential precondition for asserting the technical character of an invention. As shown above, the inclusion of human mental forces as such into the realm of the forces of nature, on whose utilisation in creating an innovation the technical character of that innovation is founded, would lead to the consequence that virtually all results of human mental activity, as far as they constitute an instruction for plan-conformant action and are causally overseeable, would have to be attributed a technical meaning. In doing so, we would however de facto give up the concept of the technical invention and extend the patent system to a vast field of achievements of the human mind whose essence and limits can neither be recognized nor overseen.

The key point here is that the use of forces of nature must be necessary for creating an innovation. 10 years later, in 1986, the Federal Court of Justice pointed out that the use of innovative calculation rules for saving energy in air-plane motors is not a technical invention, because the calculation rules are solutions to mathematical problems, and once these rules are known, there is nothing particularly inventive in applying them to the managment of fuel-use in air-plane motors. That may have been a difficult borderline case. In ordinary electronic data processing, the matter is even clearer: anybody can create a solution on his desktop computer by pure calculation with well-known abstract entities, and no experimentation with forces of nature is necessary.

see BGH 1986-03-11: Flight Cost Minimisation (Flugkostenminimierung) Decision[9]

---

[9]http://swpat.ffii.org/papers/bgh-flug86/index.de.html

and BGH 1980-09-16: Rolling Rod Splitting (Walzstabteilung) Decision[10]

In Appendix D you can find our Counter-Proposal[11], which explains paragraph by paragraph what is wrong with the European Commission's proposal and how it could be rewritten in the spirit of the EPC and of the traditional doctrine of "technical invention". Numerous opinion leaders of the software world have endorsed this counter-proposal by signing our Call for Action[12], which you find at the beginning of the documentation in your hands, together with the list of signatories.

Another approach for patentability legislation is to specify what we want and what we don't want in terms of example patents.

In Appendix A you find a framework for assessing the results of patentability legislation and also for specifying what the laws should achieve in terms of results and benchmarks. We call it "Patentability Legislation Benchmarking Testsuite". If the European Union really wants to respect the subsidiarity principle, as stated in the directive's preamble, one way to do that could be to specifying only results and leave it to member states to decide on the means to achieve those results. Such a result-oriented approach can help us understand what we are talking about. It provides an unideological basis for dialogue. If the pro-patent people can point to any examples of what they might want to call "good software patents", we can try to work out adequate software property rules in such a way that the result which the the pro-patent people want can be achieved. Unfortunately, so far nobody in the patent lobby has named any examples of what could be "good software patents". Their argumentation has so far been a purely ideological one, based on belief in certain legal dogmas which have yet to be put to the tests of clarity and adequacy. Those who want software patents have yet to come up with a statement of what precisely their interest is, what kind of idea exactly they want to be appropriatable in what manner. Only then do we have something to negotiate about. Simply saying "Patents are the way to go, so let's leave matters to the EPO", as the European Commission and some of the previous speakers have been saying, is not a negotiable position. It is arbitrariness, and in view of the overwhelming popular support behind our position I would say that such an arbitrary approach in this discussion is not only unreasonable but tyrannical. The least that must be demanded from the European Commission and from EICTA should be that they specify in clear terms – understandable and reworkable by the man skilled in the art of software – what should be ownable by software creators and what not.

Dear Ladies and Gentlemen, you are sitting here as judges. I am sorry to say it: we can come here only as accusers. We are accusing some of those, whom you have invited as experts to this hearing. They have been infringing on our basic rights without a legal basis. They have piled up a horror gallery of 30000 software patents, owned mostly by American and Japanese companies, with obscenely broad claims, covering calculation rules which any programmer would think of in a matter of minutes or hours when faced with a customer's need. We are also accusing the patent establishment of

---

[10]http://swpat.ffii.org/papers/bgh-walzst80/index.en.html
[11]http://swpat.ffii.org/papers/eubsa-swpat0202/prop/index.en.html
[12]http://swpat.ffii.org/papers/eubsa-swpat0202/demands/index.en.html

making false representation claims. They have been speaking in the name of industry associations and governments while systematically disregarding the opinions of their constituents. We hoped that the European Commission would examine our accusations. Unfortunately they have become part of the problem. Their software property policy has been framed exclusively by patent law experts who all belong to the same small community. The only place where our voice has had a chance of being heard have been the parliaments. Parliaments represent a broad range of interests and are difficult for small expert communities to control. Dear Ladies and Gentlemen, you are now the judges, please conduct fair trial! One hearing is not enough, especially when it is based on a partial opinion paper from your rapporteur[13]. Please make sure you really find out what the legitimate interests, including property interests and freedom interests, of all sides are. If some so-called software industry representatives keep telling you that the "software industry" needs patents, please insist that they show you some of examples of the type of patents which they need. Please weigh the real interests and only then look for a compromise between them. The outcome of our trial should be the establishment of a software property system that can accomodate at least the most important legitimate needs of software creators and users. Software Property is the question, and 20-year monopolies on basic calculation rules can hardly be the answer. At least no software professional in his right mind would give you such an answer. If you thouroughly investigate the issue down to the level of case-examples, you will find that there is a wide consensus in the software field on what needs to be public knowledge and what can become a kind of private property. Why not just ignore all the confusing patent newspeak for a while and simply specify in terms of examples which software property claims are acceptable to software creators and which are not? You would then perhaps end up with a very straightforward "Software Property Directive", and the problem of bringing the patent system in line with this directive would be left to the national governments and to the EPO. They would then, I guess, find their way back to the correct interpretation of Art 52 EPC[14], as it has been laid down in the EPO Examination Guidelines of 1978[15] as well as in our "Counter-Proposal". If they wouldn't find the way back to the path of virtue, the European Court of Justice would by then be well equipped to give them a helping hand.

Thank you for your attention.

5 Annotated Links

. . .

---

[13]http://swpat.ffii.org/papers/eubsa-swpat0202/amccarthy0206/index.en.html
[14]http://swpat.ffii.org/analysis/epc52/index.en.html
[15]http://swpat.ffii.org/papers/epo-gl78/index.en.html