

# Answers to Frequently Asked Questions

## Protection of investment

*Isn't protection by patents necessary in order to protect one's investment against copying? There is no progress without such a protection!*

Copyright is the perfect protection of investment for software.

Copyright has served well during the last twenty years as the motor of the software industry. With software it works even better than with books. One of the reasons is that in software a strict separation between editable source code and executable binary code is possible.

Software patents, on the other hand, are only used strategically in the current practice – i.e. in the USA. As investment protection they are much too inflexible, since they require waiting times ranging from at minimum six months to some years and cost several ten thousand Euros. Copyright becomes effective automatically and immediately.

Maybe there are a few companies which could profit from software patents, but nobody can seriously argue that software patents are useful for the whole industry.

*If a developer has spent much time in an epoch-making algorithm, wouldn't it be adequate to reward him with a patent?*

The developer can – with the copyright as investment protection – transform his discovery into software. This has been working very well in the past.

A potential imitator, who only knows the original as executable binary code, but not the source code, must spend the same amount of work as the original developer. On which economical or moral reason should he be prohibited from doing so?

The small inventor who obtains wealth from a patent and hard work is, by the way, nothing more than a nice fairy tale. In hard reality the biggest companies of the world are using patents by the hundreds as weapons against each other or against smaller and more flexible competitors.

## Current situation

*All the time I read "European software patents". I thought they do not exist currently, but should only be allowed in future?*

That's not true. About 30 000 European software patents have been granted against the law. The current jurisprudence ("status quo") differs very much from the current state of law (art. 52 EPC).

*The European patent offices and the courts are not stupid and will be able to prevent trivial patents and defend the respectable entrepreneur against absurd claims.*

The European patent offices **already have granted** thousands and thousands of trivial patents, e.g. patent no. EP 394160 on the progress bar or patent no. DE10108564 on reception of e-mail – and this against the current law (art. 52 EPC), which clearly prohibits software patents.

Even assuming the European courts would have enough expert knowledge to see through a patent lawyer operating with expert vocabulary, in most cases it is sufficient to threaten legal action with a case value of millions of Euros in order to force a small or medium company or even a single developer to give up.

In USA current legal cases impressively show where the development leads. Our only chance is to give no ground to those cases, this means retaining the current legal situation.

*Is there any hope for stopping software patents in Europe? I've heard they've already been decided upon?*

Some proponents of software patents like to spread these rumors, but actually it is **not** true. During the public session of the European Parliament on November 7, 2002 both parties, proponents and opponents of software patents, had time to speak.

The commissions on culture and industry have clearly voted against any extension of patentability of software as such and strongly criticised the direction draft. **Nothing has been decided yet!**

But it is true that not much time is left for voicing your own interests in the decision process. Act **now!**

*Isn't the battle already won? The German Government is voting for retaining the status quo and not allowing an extension of patentability.*

"Status quo" here means the current practice of jurisprudence and includes legalizing the over 30 000 European software patents, the vast majority of which are trivial and broad (see below). This "status quo" does not need to be extended: it is sufficient to legalize it to get the same situation here which exists in the USA today.

It would be completely different, if "status quo" would mean the current law – art. 52 EPC – which strictly forbids software patents. But this is not meant.

## Trivial patents

*Wouldn't it be better to solve the problem of trivial patents by demanding a minimum level of invention instead of not accepting software patents in general?*

All recent experiences clearly show that this will not work.

The software patents already granted in Europe strongly show that the patent offices are not able to prevent trivial patents. So the often stated better quality of European patent examinations is pure fiction.

*Aren't trivial patents an exception?*

No, they're the rule. If you have experience in software programming, it should be easy for you to convince yourself:

The FFII has collected and documented about 10 000 of about 30 000 European software patents, available via the URL <http://swpat.ffii.org/patente/txt/>. Randomly pick any of these patents, read and understand the claims and judge yourself:

- How large do you think is the effort for getting from the problem to the patented solution idea, in comparison with the effort for reading the patent application document?
- How high do you think is the probability that a programmer could accidentally violate a patent not known to him?
- If a customer would ask you to solve exactly this problem, how probable would you think it was, that your independently developed solution would violate this patent?

For a small number of those patents we have an easy understandable short explanation.

Notice: The original aim of the patent system is the documentation of knowledge in patent applications. At this point it should be said that this kind of "documentation of knowledge" in patents is completely useless for the programmer.

## Software patents and Free Software

*Did not even representatives of the Free Software movement speak in favor of software patents in a letter dated 2003/4/22?*

Among the signatories of this letter there is an organization which claims to represent 500 companies from the Free Software area. This organization is **not** known to the **real** representatives of the Free Software in Europe – FSF Europe, Linux-Verband, etc. – and did not receive a mandate from them.

*Wouldn't it be better to vote for an exception for free software instead of trying to prevent software patents altogether?*

Such an exception would be the same as prohibition of software patents in general, since free software may be commercial software as well. The proponents of software patents clearly know this and try to exploit this misunderstanding such that there will be at most an exception

for non-commercial software. With this, nothing would be won, since a patent only claims commercial use of an idea. The possibility of commercial use is an important point of free software.

*Some free-software-projects are developed non-commercially. Could they be attacked by software patents at all?*

Yes. The patent owner can claim that the existence of this free software hurts him commercially.

Especially in the case of a non-commercial development, the mere threat of a lawsuit is often sufficient to force the developers to abandon the project. This is because there are no monetary means to finance the lawsuit.

*Can software which is distributed as source code be attacked by patents at all? ("source code privilege")*

By the earlier directive draft of the EU commission of February 20, 2002, software could be attacked only from the moment when it is executed on a computer – thus, not the author, but the customer is vulnerable. This doesn't help me as the author, however, since my customer will hold me liable for patent claims by third parties.

The current directive draft of the European Council of November 8, 2002 contains a new article, by which publication of source code can already be a direct infringement.

*If software patents are so dangerous for free software, why does free software also exist and grow in countries which have software patents?*

The great success of Free Software easily makes one oversee the damage already done by software patents. Some projects which had to be given up due to software patents were Free Software.

As long as software patents in Europe officially do not exist, many patent holders abstain from charges, because a wave of legal cases would heat up the debate about European software patents.

*Isn't it possible to work around software patents and use alternative methods? For example Ogg / Vorbis instead of MP3?*

In some cases it is really possible. The Ogg / Vorbis developers have done patent research and hope their format won't vulnerate patents in the USA. On the other side, there are many areas where patents are so central and broad that working around them is impossible (e. g. panorama images).

But you can never be sure: patent research is not reliable. Even JPEG was believed for many years not to be covered by any patents. Now courts have to decide whether this is indeed true.

At least it is always a competitive drawback if you have to work around a file format which has been established as a de facto standard. Especially in the software sector, interoperability is very important.

## Proposals for a solution

*Wouldn't it be a useful compromise to grant software patents for five years only?*

A shorter patent duration would of course shorten the duration of damage.

But this is not allowed by international law: the TRIPS agreement demands a minimum duration of patents of at least 20 years.

*What should happen instead?*

Since software patents have been proven to have a negative impact on the economy they should not be granted at all.

A revision of the patent laws should make this clear. In practice a more narrow definition of the word "technical" is necessary.

In all recent drafts on directives such a clear definition of "technical" is missing – and thus a clear separation of what is patentable and what is not.

The FFII have worked out their own proposal for a directive draft, which corrects this flaw.

For the word "technical", the "Rote Taube" judgement of the German federal high court of justice has been taken as a base.

This proposal could be used as a starting point for a new directive draft.

However, a new directive proposal should look like this: its effects should be measured taking into account software-economical points of view, based on **example patents**.

*How can we reach this goal?*

The decision of the EU parliament is scheduled to be taken on **September 24, 2003**.

Ask your representative to listen to the arguments of the FFII and to consider them in reaching a decision! The EU parliament publishes contact information. As a supporter of the FFII you also have access to the contact information gathered by the FFII.

Ask your representative to vote against the current draft of the directive which would legalise software patents.

As the parties concerned we must stand up for our interests **now**, instead of leaving the debate to the patent departments of a few big companies, who ask the EU parliament "in the name of the software industry" to introduce software patents.

## Further Information available on the Internet:

- <http://swpat.ffii.org>