The European IPR Helpdesk

Bulletin

N°26, July - September 2017

SOFTWARE PROTECTION

CONTENTS

Software is nowadays essential for the creation of many new products, having become a key enabler for innovation, as well as a strategic driver for growth. Its intangible nature, diversity of uses and, in particular, the various ways to protect it make software a complex asset.

This new bulletin issue will assist you in better understanding the current state of play in software protection, revealing the importance of establishing the boundaries between the pure creations of the mind and technical inventions.

As an introduction, an article on software protection at European Union (EU) level by Teresa Gomez-Diaz, a Research Engineer at CNRS/LIGM (University of Paris-Est, Marne la Vallé), sets the basic concepts and legal rules in this field. It also introduces the concept of free and open source software (F/OSS) and how it appears in the open access policies of the European Commission (EC).

Then, Stefano Gentile, Legal Officer of the European Commission's Central IP Service, has contributed with an overview of the recently released 1.2 version of the European Union Public Licence (EUPL).

An article on the general regime of software patentability in Europe is provided by Sigmar Lampe, Counsel IP and Licensing at the Université du Luxembourg. He summarises the current situation of computer-implemented inventions at the European Patent Office (EPO).

In addition, two European Patent Attorneys at Barker Brettell answer our questions in an interview that aims at assisting you to further understand computer-implemented inventions in Europe as well as at providing some guidance on how software companies can best manage their software-related Intellectual Property Rights (IPRs). Furthermore, BMAT, a Spanish company operating in the music industry, shares its expertise with us, answering our questions regarding software protection from the perspective of a small to medium size enterprise (SME).

You will also find in this issue a short article regarding the 2017 Campaign of the EU Open4Business series of events, organised by the European Commission in cooperation with the local Enterprise Europe Network partners, where different services and projects of the European Commission are presented to local stakeholders to better support European SMEs.

As per usual, the Bulletin reports information about past IPR events together with the latest news from our Helpline service. In addition, we give you the chance to test the lessons learned with our brand new software crossword quiz as well as your knowledge on patent searching with the usual patent quiz.

Wishing you an inspiring read!
Your Editorial Team

Software Legal Protection, FOSS, and EC Open Access Policies

Teresa Gomez-Diaz
Research Engineer, CNRS/LIGM

1. Introduction

The legal protection of software can seem complicated, but this usually reflects some lack of knowledge about the concepts that play a role in this game. Our goal is then to introduce the basic concepts and to facilitate the understanding of the functioning of some essential legal aspects around software protection. We also present the free/open source software concept and show how it appears in the open access policies of the European Commission.

2. Legal framework

In the international landscape, software copyright is protected as any other copyright work by the Berne Convention. At European Union (EU) level, protection is secured by Directive 2009/24/EC (the "Directive"). It may be unbelievable to some but evident for others that poetry and computer programs are protected under the same law.

Along with this international legal context, each country has its own copyright law, and this whole system evolves through judicial interpretation or through legislative intervention at national and EU level, which usually seeks harmonisation[1].

To complete the legal framework and subject to legal compliance, parties can regulate the terms of software-related agreements, such as distribution licences, collaboration, employment, etc.

3. On the production side

What is understood by software or computer program is, in fact, a set of different objects that can include source code, compiled code or executables, interfaces, documentation and even preparatory design work. But protection can apply differently; for example underlying ideas and algorithms, as well as interfaces, are not protected.

Copyright protection applies automatically, once the software starts to be written and the rights are associated to the person who writes the software, who can claim authorship rights. If the software is developed in a professional context, the employer is entitled to exercise all the associated economic rights, unless it is otherwise specified in a contract. The employer is then identified as the rightholder or the owner of the program. If there is no employer, all the rights are associated with the author, except if there are agreements specifying other conditions.
Originality is an essential requirement of copyright law, this is why new functionalities need to be well dated and documented. In the Directive the originality concept is defined as the "author's own intellectual creation".

The notion of author in software can be simple in the case of only one writer. However, this notion can become less simple in contexts where there are different actors and different roles. For example there can be parts of the code that have been updated and rewritten by different persons, but the former authors are still part of the authors' list. In other situations, persons external to the project can propose code to correct a bug or to add new functionalities, and this code will be adapted by other developers to be integrated into the main code[2]. Note that when different authors employed by different entities participate in development of the software, the entity's percentage of ownership derives from the percentage of the corresponding authorship.

4. On the customer or user side: licences

It is clearly stated in the Directive: the running, loading, reproducing, translating or arranging of a computer program can only be done upon the corresponding (written) authorisation. This is why software is distributed under licences. Licences can only be granted by the software owner (i.e. the rightholder).

The usual business model consists of selling licences that give the right to use the software under certain conditions, and does not include the sale of the software itself, as the source code is rarely a part of the commercial exchange.

Licences, sometimes named end-user licence agreements or software licence agreements, are the contracts between the licensor and the licensee (the purchaser of the licence),

>

1 For further information on European Union Copyright law harmonisation see Margoni T. (2016) The Harmonisation of EU Copyright Law: The Originality Standard. In: Perry M. (eds) Global Governance of Intellectual Property in the 21st Century. Springer, Cham. You can also access the draft version at SSRN here.

2 You can find an analysis of the author's concept in Gomez-Diaz, T. (March 2015) Article vs. Logiciel: questions juridiques et de politique scientifique dans la production de logiciels. 1024 – Bulletin de la société informatique de France, numéro 5, mars 2015, pp. 119–140.

and establish the rights granted under the licence. In the case where you find a computer program which does not include any licence or without any other legal mention, the default legal context that applies is "All rights are protected", and means that no one can run, load, etc, the software (except the rightholders).

5. Free/Open Source Software (FOSS)

In 1985, Richard Stallman publishes the GNU manifesto and launches the Free Software Foundation (FSF), a movement that establishes other ways to deal with software, based on the user's liberty concept. This movement creates the free software definition that we reproduce here:

"A program is free software if the program's users have the four essential freedoms:

- *The freedom to run the program as you wish, for any purpose (freedom 0).*
- *The freedom to study how the program works, and change it so it does what you want it to do (freedom 1). Access to the source code is a precondition for this.*
- *The freedom to redistribute copies so you can help your neighbor (freedom 2).*
- *The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this."*

In 1998, the Open Source Initiative (OSI) was launched around the concept of open source software. Despite what some may think, the concept of open source software is not just about access to the source code. The distribution terms of open source software must comply with ten different criteria, where the first one is as follows:

"Free Redistribution: The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale."

As one can understand from the definitions, these two movements correspond to very different philosophies, but are sometimes confluent in common goals as, for example, software quality.

A computer program falls within the category of free and/or open source software if the licence complies with the conditions established in the definitions. Most of the licences used in this kind of software make it both "free" and "open source" so there is no real legal distinction between these two concepts, despite the philosophical aspects. However, many members of the software development community prefer not to adopt one or the other philosophy, this is the reason why the term Free/Open Source Software (or FOSS) is widely used.

It is common to find a "non warranty clause" in free/open source licences in order to indicate that the software is provided "as is", and that the developers have no liability with respect to the (bad) use of the software. Another kind of clause is the "reciprocity clause", where the licensor asks the licensee to respect some conditions, such as keeping initial copyright information in derivative work or to release derivative work under the initial work's licence. This last example corresponds to the strong Copyleft clause[3] that can be found in the General Public Licence (GPL licence) developed by the FSF.

There are usually other kinds of business models[4] associated to free/open source software, as it is possible to sell support, or to provide customisations or additional functionalities under different licences.

Companies can also invest in free/open source software following different collaboration models, for example by engaging its own staff in the collaborative development effort. By doing this, the company can make an external software product evolve to reach its own goals, while assisting the development community as a whole.

One of the most important reasons to invest in these developments is that the collaborative model usually produces more reliable and of higher quality software. Indeed, users can detect bugs and signal them, or even propose correction code. The developers can validate and integrate the proposed code in the main version of the software or find corrections otherwise. The production of new versions is thus faster and the dissemination mechanism is shorter than in software produced inside one company with more traditional development models.

6. European Commission free/open source software policies

In 2004, the European Commission decided to distribute its own produced software under a licence that grants all Free (or Open Source) freedoms and created the European Union Public Licence (EUPL)[5] to be adapted to European copyright law and terminology. The EUPL has been certified by the OSI since 2009 and has been recently updated to a new version 1.2[6] to extend coverage to data and documents (among other works) and to be compatible with a wider range of other open source licences.

Free/open source software has been present in the European Commission open access policies at least since 2012 and is integrated in its strategic vision: Open innovation, open science, open to the world. It appears in the funding of the H2020 Work program ICT 2016-17 as a way to increase use and effectiveness of EU-funded projects.

Free/open source software extends more and more its area of influence, as we can see for example in the European Commission's strategy for the internal use of Open Source Software (first adopted in 2000).

3 For a classification of free/open source licenses you can consult: Gomez-Diaz, T. (September 2014) Free software, Open source software, licenses. A short presentation including a procedure for research software and data dissemination.

4 For further information on the GPL license see Eben Moglen's plea for Free Software before the European Parliament, 2013-07-09

5 See Commission Implementing Decision (EU) 2017/863 of 18 May 2017 updating the open source software licence EUPL to further facilitate the sharing and reuse of software developed by public administrations.

6 For further information on the EUPL read Schmitz, Patrice-Emmanuel, (2013) The European Union Public Licence (EUPL), International Free and Open Source Software Law Review, 5(2), pp 121-136 DOI: 10.5033/ifosslr.v5i2.91, as well as the following article of Stefano Gentile, Open Source: An Overview of the New EUPL v1.2, that you can find next on this Bulletin issue.

Open Source: An Overview of the New EUPL v1.2

Stefano Gentile
European Commission's Central IP Service

The European Commission has recently released a new version of the European Union Public Licence (EUPL). Keeping to the tradition of copyleft open source software licensing (OSS), the EUPL version 1.2 introduced a number of updates while maintaining its most distinctive feature: the downstream cross-copyleft flexibility. Let's break down the technical jargon to understand what this flexibility is all about.

The so-called "copyleft" licences are a particular sub-set of open source licences designed to keep software (and any modification thereof) open source. They do this by including a copyleft clause, which is an obligation triggered when the software is distributed and that requires such distribution to be done under those very same copyleft terms. In simpler terms, the copyleft clause generates a "viral effect" that extends to any modified version of the original code or to any additional code that is incorporated with the original code.

While copyleft licensing is indeed a well-established means to distribute open software, the proliferation of copyleft licences has resulted in a growing number of cross-compatibility concerns. In fact, whenever software integrates components released under two (or more) copyleft OSS licences, each licence would require the compulsory application of its own terms to the downstream distribution of the software. However, where these licences happen to be incompatible, the distribution of the resulting software may be hindered.

Here is where the EUPL shows its best qualities: In the case of cross-copyleft problems, a unique "compatibility clause" introduced in the text of the EUPL enables users to opt for the application of the concurring copyleft licence, thereby removing potential licence conflicts. This flexibility is triggered whenever the integration of EUPL software with a component released under another copyleft licence (e.g. GPL) would otherwise result in a downstream distribution lock. Thus in the above example, software consisting of both EUPL and GPL-licensed code can be released under the terms of the GPL.

It is for this reason that the EUPL is sometimes labelled as "flexible" copyleft in an attempt to further dissect the copyleft licence family to a finer granularity and perhaps to distinguish it from the uncompromising "strong" copyleft licences, whose most prominent representative is the GNU GPL. Labelling aside, it should be noted that the described EUPL flexibility is confined within a pre-determined list of "compatible licences". This means that the flexibility will only occur in relation to licences that ensure comparable open source freedoms and safeguards. Outside of these specific circumstances, the EUPL remains by all means a strong copyleft licence just like the GPL.

The list of EUPL compatible licences has been updated with the release of version 1.2 and currently includes:
GPL v.2 and v.3, AGPL v.3, OSL v.2.1 and v.3, EPL v.1, CeCILL v.2 and v.2.1, MPL v.2, LGPL v.2.1 and v.3, LiLiQ-R and LiliQ-R+.

The list may now be updated to later versions of the same licences without the need to produce a new version of the EUPL, as long as these would guarantee the same level of freedoms and safeguards from exclusive appropriation.

Another feature introduced by the new EUPL release is the inclusion of ancillary works within the scope of the licence. While not being software per se, ancillary works serve the purpose of supplementing and/or

Stefano Gentile, IP Legal Officer, European Commission

supporting the software and are therefore often distributed together with it. This is typically the case of manuals and technical specifications. In order to avoid that these would circulate devoid of clear licensing terms, the text now extends its scope to them. To this end, the wording has been modified from its previous reference to "software" to the more general copyright term: "work". In spite of this, the licence is and will remain inherently a software licence. This means that while the new terminology brings legal certainty to the ancillary works accompanying a software distribution, it is clearly the latter (i.e. the software) that justifies the adoption of the EUPL as a licensing instrument. In this regard, given that the application of software licence terms may be unsuitable for other type of copyright works, the EUPL "compatible licence" list features a specific licence exclusively for these ancillary works - the Creative Commons Attribution Share-Alike (CC-BY-SA).

The concept of share-alike introduced by Creative Commons is akin to that of copyleft in that both require copies or adaptations of the work to be released under the same or similar licence as the original. Nevertheless, software licences such as the EUPL and non-software licences like the CC-BY-SA should never be used interchangeably. Their benefits (and usefulness) are only retained within the specific scope of application for which these licences were designed (respectively software, and non-software works).

With the new release, the EUPL aims at providing licensors with a reliable legal platform for the distribution of open source software, while supporting licensees in the adoption and development of EUPL-based software by reducing possible licensing frictions down the line.

Software Patents at the EPO – From Exclusion "as such" to "further technical effect"

What is the situation today?

Sigmar Lampe
European Patent Attorney, Counsel IP and Licensing, Technology Transfer Office, Université du Luxembourg, Luxembourg

Although the European Patent Convention (EPC) states in Article 52(1) that "European patents shall be granted for any inventions, in all fields of technology, provided that they are new, involve an inventive step and are susceptible of industrial application", it is important to note that there are further requirements which a presumed invention needs to fulfill. This becomes particularly relevant when considering computer programs, or software.

Criteria for patentability

Novelty Article (54 EPC)
The invention must be new to the world, i.e. any previous public disclosure of the invention before the date of filing of the patent application would destroy this novelty. Moreover, it is irrelevant by whom this disclosure was made, even by the inventors themselves.

Inventive Step Article (56 EPC)
The invention shall also comprise an inventive step, i.e. it must not be obvious to the skilled person in the light of the state of the art. The inventive step is evaluated by assessing whether the technical solution is obvious or not to a person skilled in the art when this person is faced with the problem described in the patent application.

Industrial applicability (Article 57 EPC)
This requirement is fulfilled if the invention can be made or used in some kind of industry, including agriculture.

An essential further requirement is that the subject matter of the patent application must qualify as an "invention". It may be surprising that there is no positive definition in the EPC of what is understood by an invention. There is, however, a non-exhaustive list of exclusions which are, by definition, not considered inventions and therefore cannot be patented.

Examples of such "non-inventions" are e.g. schemes, rules and methods for performing mental acts, playing games or doing business, and programs for computers (Article 52(2)(c) EPC). It is worth noting, however, that these exclusions apply only to the extent that an invention relates to such excluded subject matter "as such". In other words, anything that is not excluded subject matter "as such" would theoretically qualify as a potential invention.

This naturally leads to the follow-on question: what does "as such" mean?

From exclusion "as such"…

Although Article 52 of the European Patent Convention had been phrased essentially like this since signing the EPC in 1973, it was mostly due to the appearance of software-based solutions in all aspects of technology, and life in general, in the 1990s, that the question of what was meant by excluded subject matter "as such" became a real issue and more clarity was required. A longer debate followed in Europe about how the exclusion of computer programs "as such" should be interpreted. A full recount of this development would go far beyond the scope of this article, therefore suffice it to say that over the span of almost 20 years the EPO case law has slowly converged on an interpretation that is more or less stable, albeit not necessarily intuitive.

If the limitation to the exclusion from patentable inventions were interpreted literally, we would find that almost every computer program would cause at least a trivial technical effect when running it, such as electrons moving in transistors or positioning the read/write head of a disk drive. In the end, every computer program would easily constitute more than a computer program "as such" and therefore not be excluded from potentially patentable inventions.

Following that approach, however, inventions might qualify as potentially patentable merely due to trivial technical effects which may have been separate inventions at their times, like microprocessors or memory chips, but without contributing a novel and inventive technical solution to the knowledge in the field today. It may also be possible to simply include known technical elements into the claim, so that by clever wording alone a computer program might escape the legal exclusion from patent-eligible inventions. In this connection, it is useful to recall that an invention which escapes the exclusion from patentability is not thereby automatically patentable, but it merely has just overcome the first hurdle. By qualifying as a potentially patentable invention, it still needs to satisfy the substantial requirements of novelty and inventive step (see insert), just like any other invention.

…to "further technical effect"

Therefore, at least at the EPO, the analysis has largely shifted away from the exclusion-from-invention criterion to a more thorough technical analysis of the invention at hand when assessing inventiveness. It is the analysis of the presence of an inventive step during which each invention is carefully scrutinized for the technical solution which it contributes to the state of the art in a non-obvious manner. The inventive step can only be confirmed when the invention proposes solving a problem through technical means. The problem solved should advantageously be a technical one, but it is also accepted that it may be of non-technical nature (e.g. an accounting problem) as long as the solution provided is technical.

By applying this approach, the EPO ignores all non-technical features of the invention from

>

the assessment of the inventive step, and concentrates on the technical ones instead. It then follows in a rather straightforward manner that the already known trivial technical effects described above cannot form part of an inventive technical solution which would go beyond what was previously known. The analysis of the remaining technical features (if any) which positively contribute to a technical solution may eventually lead to the confirmation or denial of an inventive step. The reduction of the analysis of the inventive step to exclusively the solution provided through technical elements, while excluding all non-technical features, leads to a concept known as "further technical effect". This term describes the effect which the non-trivial technical features of the invention contribute to the solution of the underlying problem.

Examples of computer-implemented inventions:

- controlling machines or industrial processes;
- controlling the internal processes in a computer;
- helping to use resources in a computer more efficiently, or increases its performance;
- increasing the safety of the computer;
- making a computer system easier to use;
- improving the transmission or storage of data;
- providing a device with new functionality.

Summary

The discussion in Europe has moved on from the exclusion of computer programs "as such" and now centres around a thorough analysis of the inventive step. It is assessed whether technical features of the invention provide a non-trivial "further technical effect" in a non-obvious manner, whilst providing a solution to the (technical) problem.

How does the situation in the U.S. compare with Europe?

At the international level, we have recently seen an approximation of the strictness of the U.S. patent system to the EPO approach, if not in the reasoning, but more so in the stricter approach to excluding patentability of abstract ideas. This follows from several remarkable decisions by the U.S Supreme Court which mark a turning point from the previous relatively permissive practice in the U.S. to a much more restrictive approach when it comes to patent-eligible subject matter [Bilski v. Kappos, 561 U.S. 593 (2010); Mayo Collaborative Servs. v. Prometheus Labs., Inc., 132 S. Ct. 1289 (2012); Alice Corp. v. CLS Bank Int'l, 573 U.S., 134 S.Ct. 2347 (2014)].

Essentially the United States Patent and Trademark Office now applies a machine-or-transformation test and denies patentability for abstract ideas. Interestingly, although the legal provisions and their interpretations as manifested through case law are substantially different in Europe and the U.S., this sometimes leads to results which in the end may be surprisingly similar.

Conclusion

As long as software forms part of a computer-implemented invention, it should be possible for it to overcome the exclusion from potential inventions for computer programs as such under the EPC.

For evaluating the patentability of a computer-implemented invention in Europe, it is largely irrelevant if it is implemented in the form of software or hardware, as long as it represents a technical solution to a problem, thereby providing what is called a "further technical effect". Just like any other invention, a "computer-implemented invention" is subject to compliance with the standard patentability criteria of novelty, inventive step and industrial applicability.

It is typically the inventive step which presents the biggest hurdle at the EPO, not the implementation of the technical solution in the form of software.

For further reading:

- European Patent Office: "Patents for software? European law and practice", 2013
- World Intellectual Property Organisation: "Patenting Software"
- Dennis Crouch: "Patenting Software in the US as compared with Europe", September 29, 2014
- Rupert A. Knights and Craig A. Redinger: "Patent Eligibility of Software Patents in the U.S. and Europe: A Post-Alice Consideration", Landslide, Volume 8, Number 1, 2015

For more in-depth practical and legal information, the reader is recommended to seek professional advice.

INTERVIEW

"There are many areas where a computer-implemented invention that is enabled by software can be considered patentable, which can result in the software itself being the subject of a valid patent."

In this issue, we have interviewed two European Patent Attorneys at Barker Brettell. The interview will assist you in further understanding computer-implemented inventions in Europe as well as provide some guidance on how software companies can best manage their software-related Intellectual Property Rights (IPRs).

There is a common understanding that it is not at all possible to obtain a patent for a software program, is it true?

No, this is not true. In Europe it is possible to patent software, provided the software achieves what is known as a "technical effect".

The European Patent Convention excludes computer programs "as such" from being considered inventions, and therefore being patented. However, the European Patent Office (EPO) sets a low bar for avoiding this statutory exclusion. An invention claimed using any technical feature (which may be as basic as pen and paper) will overcome this bar.

Any invention that is defined as being computer implemented, even if this is only implicitly, will therefore pass this test. This is not, however, the end of the story, because all inventions also have to be both new and inventive to be patentable.

For assessing inventive step, the EPO requires that the claimed invention achieves a technical effect. Anything "non-technical" is excluded from contributing to the technical character of the invention, and cannot therefore be taken into account in assessing inventive step. Commercial/business-related information and outcomes are, for example, considered to be non-technical, as are ways of presenting information or playing games. A computer processing non-technical information will therefore not be considered inventive unless doing so achieves a technical effect. Positive examples of a technical effect include: the provision of data about a technical process (e.g. the state of a machine); provision of data that is applied directly to a technical process; controlling or interacting with a technical process outside of a computer (e.g. a process plant or measurement device); cryptography; and image, sound or video processing.

There are many areas where a computer-implemented invention that is enabled by software can be considered patentable, which can result in the software itself being the subject of a valid patent.

What does it mean in practice that the software implementation solves a technical problem in a non-obvious manner?

To be considered inventive, any claimed invention (not just software) needs to solve a technical problem in a way that would not be obvious to the skilled person. Examination of patent applications at the EPO always follows a structured test known as the "problem-solution" approach in deciding whether a claimed invention is actually inventive. The starting point is to consider what is the closest prior art to the claimed invention, and see how the claimed invention is different. The effect of this difference is then used to define an "objective technical problem", which is posed to the nominal skilled person. If the same problem has been solved elsewhere in the same way as in the claimed invention, for example if the solution is mentioned in another piece of prior art, it could be considered obvious to the skilled person how to solve it, and the claimed invention would then not be inventive.

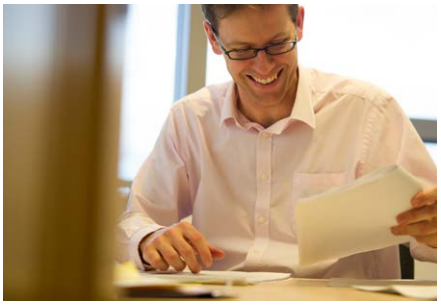
In the case of software inventions, the only difference is that any non-technical features (e.g. aspects of a business method) are not considered to be part of the problem to be solved. A software invention that merely implements a business method would therefore not be inventive, because the objective technical problem would be how to implement the business method on a computer. A skilled person with knowledge of computer programming would find this obvious. A software invention that implemented a new cryptographic method could, however, be inventive, even though the invention would be implemented purely in software.

Could you give us some practical examples of computer program patents? Which kind of software patents do applicants want to get protection for?

In general, applicants want to get patent protection for software when it is the software itself that enables the invention, and particularly when other components of the invention, for example the hardware the software runs on or interacts with, are already known. If it is not possible to get patent protection for the hardware alone, not being able to patent a software-enabled invention can restrict an applicant's ability to protect their intellectual property and can make getting investment more difficult.

Software is used to implement many kinds of new ways to acquire and process measurement data, for example to implement new types of algorithms that

David Pearce, Partner at Barker Brettell, EPA, CPA



David Combes, Associate at Barker Brettell, EPA, CPA, EUIPO

>

process data to obtain measurements that were not previously possible, or to make measurements faster or more reliable. Software is also used to monitor and control external hardware, one example being controlling fleets of automated robots within an industrial setting. Various rules relating to how such robots can interact with each other and operate in an efficient way, for example in planning routes around a factory or warehouse setting, can be implemented in software.

Computer modelling of real world systems can also be patentable in some cases, in particular when there is a link to the external environment from the modelled environment. One example is thermal modelling of air flow within a computer data centre, where cooling of computer equipment is of critical importance. Being able to model airflow to identify and resolve potential problems allows a software-based model to be used to suggest changes in the external environment that can result in substantial energy savings.

As a more general example, there is currently substantial interest in developing autonomous vehicles, being undertaken by many different companies. While the hardware to enable such vehicles to work is already in existence, software that would enable safe and efficient operation is still very much in the development stage. New software-based methods to allow a vehicle to operate more safely, more efficiently, and without human interaction would certainly be patentable, at least in principle.

What advice could be given to software companies regarding the management of their software-related IP rights?

While other rights such as copyright exist for protecting software, patents remain the best way of broadly protecting any technical invention, which can be key to obtaining a competitive advantage and for obtaining investment. It is therefore always worth at least considering whether a patent might be worthwhile when developing new software-related inventions. If there is any question about whether an invention might be patentable, our advice would be to always speak first to a qualified European patent attorney, who will usually be able to provide basic initial advice for free that will enable a decision to be made on whether to seek a patent.

The most important point of advice is to make sure that the invention is kept secret until that decision has been made, because a public disclosure of the invention can result in the chance of getting a patent being lost.

Contact
Barker Brettell LLP
www.barkerbrettell.co.uk

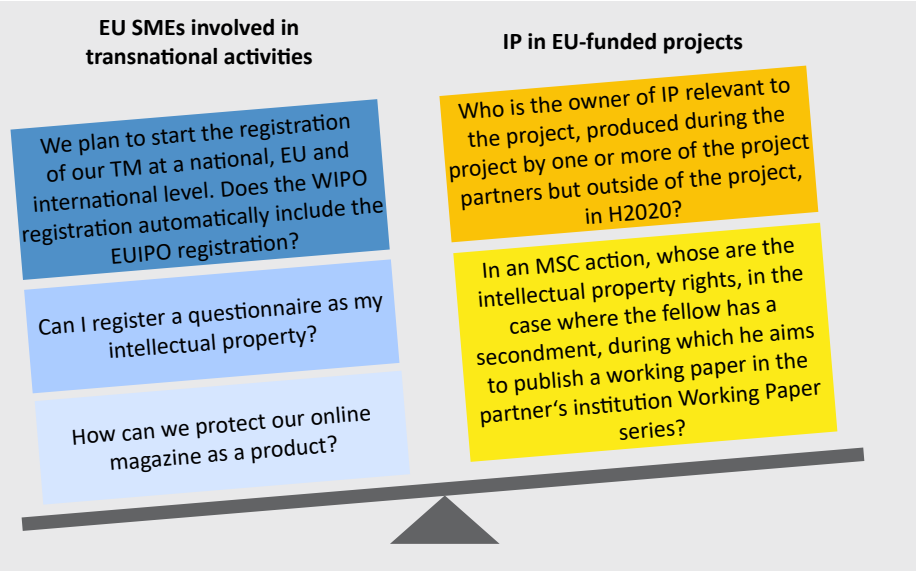
Your IPR Queries Matter to Us: Ask the Helpline

The European IPR Helpdesk Helpline answers your questions concerning intellectual property (IP) within three working days. You get practical, first-line support directly from our IP experts, and free of charge.

If you are curious about the type of IP queries that the Helpline has recently been dealing with, these are shown in this illustration.

If you would like to talk to one of the IP experts of our Helpline, please dial +352 - 25 22 33 – 333

www.iprhelpdesk.eu/helpline



FREQUENTLY ASKED QUESTIONS

IPR GENERAL :

In the case where we sell a product in several countries, and the family patents protecting this product have a different legal status in the patent prosecution process in each one of the countries - for example: "patent granted" in China, Taiwan and the United States, but "patent pending" at the European Patent Office and in Japan and India - how should we mark the product?

Is the state of the family to be marked on the product defined by the territorial regulations of the country where the headquarters of the applicant is settled? Or, in the case where the patent became granted firstly in any one of the application countries, this allows us to mark "patent granted" on our product instead of "patent pending", although the product is also going to be sold in other countries where the patent is still pending or even not applied for?

Patents grant rights to their right holders for a certain territory each time. Even if we literally talk about the same invention, patents in patent families are independent applications that can be rejected or granted accordingly in the relevant territories. Hence, if a patent is granted in China, Taiwan and the United States then the goods that will be commercialised in these countries should have the marking "patent granted" along with the number of the patent.

On the other hand, if the patent is pending in some territories (such as at the European Patent Office and in Japan and India) the marking should be "patent pending" along with the number of the application. Please note that the marking on the product should be changed from "patent pending" to "patent granted" once and if the patent is granted in some or all of these territories. As patent rights are territorial rights, the marking on the product should follow the rules of the country in which protection is given or is sought, that is to say the territory of commercialisation. Put simply, you cannot commercialise a product with a patent marking "patent granted" in the USA if this product is not protected by patent rights in this country. The place of the headquarters of the right holder of a patent is irrelevant. Hence, you should designate the patent marking according to the patent status in every country. For instance, if there is a patent granted in the US you should mark the product as patent granted. On the other hand, if in Japan the patent is still pending you should designate the product as "patent pending" and change the marking once the patent is granted in the future.

We would like to highlight that you are not allowed to put a patent marking on your product in a country where you have not applied for a patent, since in some countries making false claims is a serious offence under the respective laws and could be considered as an attempt to mislead the public.

The situation could be problematic where you offer a product online and it is possible to buy it in different countries with different statuses. Patent marketing for products that are sold online is a rather complex issue and for this reason we suggest you to resort to a patent professional with the relevant experience. To be on the safe side you can provide virtual marking (a link that leads to a web page where the user can find all the information about the patented product and in which territories it is protected). Alternatively, you could provide the information on the status of application/grant in all countries applied/granted. Lastly, you could ask the owner of the patent (licensor) on how to deal with this situation. There may be guidelines by the right holders for this occasion.

EU-FUNDED PROJECT:

I presently have a conflict on the negotiation of a consortium agreement of a RIA project, which started on 1 May 2017. The project is quite frankly orientated towards industrial developments and we have a divergence as regards intellectual property provisions and joint results in particular, art. 8.1 of the Consortium Agreement.

1. Joint ownership shall also be granted to the Party who provided access to the Background which was required for the development of the Result (we accepted it, even if not all parties have consented to identify their needed background).

2. Direct use defined by the coordinator as "direct utilisation of results means that the owner of the results uses them within other research projects as well as for further development, creation and marketing of a product or process" shall be "free of charge and without prior consent". It is specified however that this right expressly excludes the right to use the Background of a Party without the express written consent of such Party, under the terms and conditions to be negotiated and agreed between the Parties.

Regarding joint ownership, the clause you sent us is not incompatible with the provisions of the GA – partners are allowed to agree on alternative ownership arrangements which would be more beneficial to background owners.

On the other hand, the GA makes it compulsory for beneficiaries to identify their background in writing. This obligation is set up under article 24.1 GA; non-compliance may result in sanctions such as the reduction of the grant (see article 24.2). We understand from your email that this has not been done by all partners. We would therefore suggest making your acceptance of this clause conditional to the identification of background in writing by the defaulting parties, because it is one of their obligations, but also because agreeing to this clause without knowing which background is relevant or may be used will create a lot of legal uncertainty.

Regarding the direct use of results the definition made by your coordinator is quite standard. Remember however that the minimum access rights set forth in the GA are mandatory and cannot be set aside. Therefore if one of the co-owners needs the other co-owner's background in order to exploit the joint results, access to such background will have to be granted to the requesting party, to the extent necessary. The right to use the related background will not be automatic – it will have to be requested (cf. access rights provisions) – but cannot be annulled by a clause imposed by one party in the CA.

INTERVIEW

"First, we need software protection through patents in order to obtain a high level of protection for the results of our researches."

BMAT, a Spanish company operating in the music industry, shares its expertise regarding software protection from the perspective of a small to medium size enterprise (SME) with us.

Could you tell us briefly about BMAT's activity and what it means for the music industry?

We work with the aim of making music a better place. There are many ways to improve such a wide industry and our input is providing information that makes it more transparent and fair for both artists and users.

We develop software for audio recognition to track the music that is being played worldwide on TV and radio channels, clubs and digital service providers (such as Spotify, Deezer, Google Play, etc). Then, we report this data to 80,000 labels and 90 Collective Management Organisations, so that they can distribute the copyright royalties among their members more accurately.

BMAT was born 11 years ago as a spin-off of Pompeu Fabra University and we still have our initial passion for music and technology. That is also the reason why we are so much aware of innovating within audio technologies.

You define yourselves as music innovators, why is that?

We are music innovators because of our software solutions, but we would rather like to think about its final implication, which is the service it gives to the music business community. We strongly believe that developing a technology that improves distribution of royalties makes the music world a fairer place for the artists and we are glad to be helpful in this process. It increases the music long tail and it helps the most unknown artists.

Does BMAT invest many resources on innovation and software innovation?

Our R&D department works every day to improve the potential of our fingerprinting software. In the last two years, we have invested 24% of our income in R&D and we normally participate in several European projects mainly with the aim of developing our technology for audio recognition, but also in many other activities. For instance, even though we cannot see an immediate profit from it, we are starting to research the applications of blockchain technology within the music industry. We like it, we are curious about it and we feel we must act according to that "instinct".

What are those European projects you participate in and how do you feel supported by them as a European SME?

We have already participated in several European projects (such as H4H or MIReS) and now we are getting ready for two big projects which are to be announced. We will lead one of them and cooperate in the other one along with companies, startups and institutions from all over Europe. Both are focused on innovation and that is the reason why we regularly keep applying to participate in more projects; they let us come up with new approaches and acquire a very solid expertise from all the players involved and technologies we use. They are multidisciplinary, creative and very stimulating for such a dynamic and curious company as BMAT.

Why is software protection important for BMAT?

First, we need software protection through patents in order to obtain a high level of protection for the results of our researches. But, most importantly, we think of it as a fundamental asset, capable of adding value to the entire company. At the Legal Department, we think that intellectual property protection, including trademarks, secures our rights and their legal protection. Furthermore, protecting our software also has a positive business effect. In this regard, it is something that underlines the development efforts assumed by a company and shows its hunger for innovation.

How does BMAT protect its software?

Even though the concept of software patent does not exist in the EU, our software often involves hardware background and technical effects, so we normally manage to obtain patent protection for our achievements. We believe that, in certain cases, it might be a mistake to consider software only as a copyrightable product, since patent protection would surely fit better the features of this specific type of results. We also think that the boundaries between patentable and non-patentable software should be reconsidered in Europe: currently it is not possible to patent software as such. In order to be patentable, software must present a "technical effect". We consider this kind of prerequisite too restrictive and we believe that it might turn out to be discouraging for software innovation.

Besides, we are very careful when it comes to granting any kind of right over our software. That is why we pay a lot of attention to being clear and transparent in our agreements. We set out very clearly the temporal and territorial limits of the licences which we grant and the types of use authorised under the licence.

What advice could be given to software companies regarding the management of their software-related IP rights?

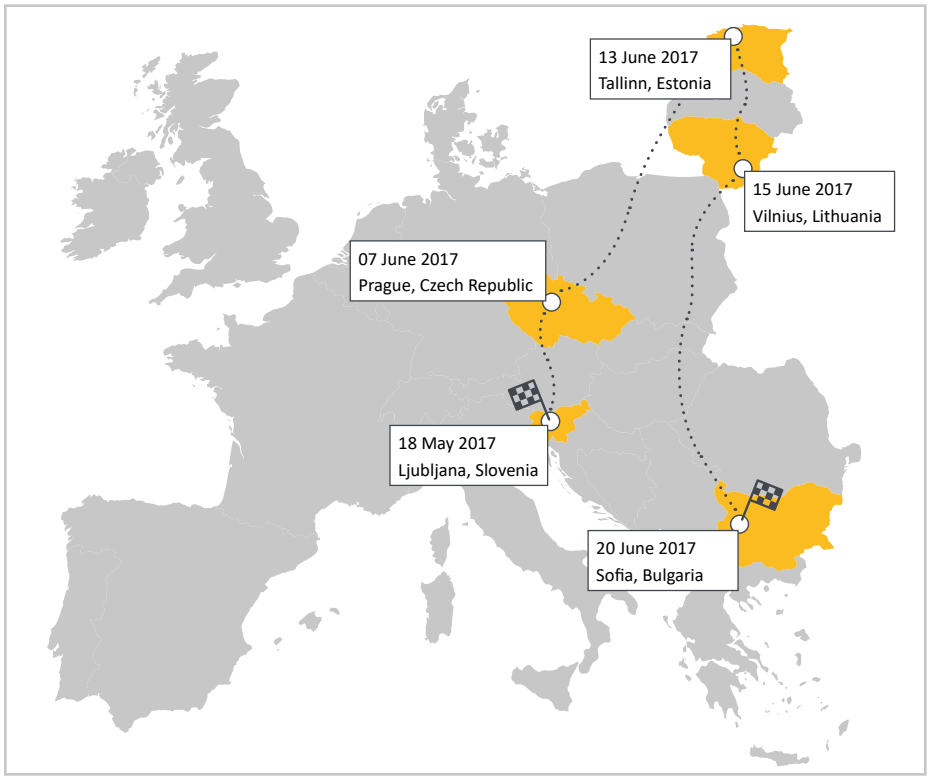
On the one hand, beware of the risks that might threaten your industrial property assets. Software protection can be expensive, especially for SMEs, and requires the help of experts and professionals, but we think that it is better to be on the safe side. On the other hand, do keep in mind the importance of a good patent portfolio for your business relationships. IP protection is too often seen as an unavoidable expense, and too seldom as a powerful investment which will end up helping your company shine and show the best of itself.

EU Services at Your Doorstep: Open4Business 2017 Campaign

The European IPR Helpdesk

The European Union provides a wide range of support services for small and medium-sized enterprises (SMEs), but many entrepreneurs are unaware of the help they could be receiving. This is why the European Commission launched its EU Open4Business campaign back in 2015. Each year since then, the campaign has targeted SMEs in five countries, telling them about services they can get from the Enterprise Europe Network and the financial support available under EU programmes.

In 2017 the campaign targets Bulgaria, the Czech Republic, Estonia, Lithuania and Slovenia. A new approach has been adopted this year: the campaign kicked off with a series of information days for organisations which advise small businesses. The aim was to inform business advisers about the EU services, covering a wide spectrum.



The information days attracted from twenty to sixty participants in each country. The events were organised by the European Commission in close cooperation with the local Enterprise Europe Network partners. The programme included access to finance and advice services offered at the EU and national level. COSME financial instruments, EU regional funds and the LIFE programme as well as national funding opportunities were all covered.

On the services side, the Network partner search and market access service, SOLVIT, the European IPR Helpdesk and the Erasmus for Young Entrepreneurs programme were explained. A typical day concluded with a case study where entrepreneurs explained how they benefitted from EU services.

During this one-month long roadshow, the European IPR Helpdesk promoted its services with the support of its Ambassadors and invited external speakers, and held meetings to extend its stakeholder network in these countries.

The second phase of the campaign will begin in September with a mix of radio spots, social media and print advertising calling on entrepreneurs to contact the local Enterprise Europe Network office or consult the Your Europe Business portal to obtain a broad range of information about their rights in the EU single market.

The European IPR Helpdesk on Tour: Take a Look at a Selection of our Recent Events

In the last three months the European IPR Helpdesk Team participated in a number of IP events all over Europe, and provided several IP workshops building capacities in IP management among SMEs and researchers.

Meet us at these upcoming conferences

- *15 October 2017: Bucharest, Romania*
  Forum for Innovation
- *24-25 October 2017: Tallinn, Estonia*
  Manufuture 2017

Upcoming IP training events

- *14 September 2017: Brussels, Belgium*
  IP and Coffee: Joint webinar session
- *15 September 2017: Cappadocia, Turkey*
  IP Conference for TTOs
- *19 September 2017: Oslo, Norway*
  IP Management in H2020, at the Annual Event of Innovation Norway
- *27 September 2017: Brussels, Belgium*
  IP Management & IP Exploitation in H2020 for the Common Support Service
- *28 September 2017: Malaga, Spain*
  IP Commercialisation
- *04 October 2017: Belgrade, Serbia*
  Creating Values – Impact & Exploitation in H2020 projects for the EEN Belgrade, in cooperation with the Center for Technology Transfer, University of Belgrade
- *10 October 2017: Brussels, Belgium*
  IP and Coffee: Joint webinar session
- *12 October 2017: Berlin, Germany*
  IP strategy for SMEs, in cooperation with Berlin Partner (EEN) and Fit4Health2.0
- *17 October 2017: Bilbao, Spain*
  IPR in general and Technology Transfer, in cooperation with the Foundation for Health Research and Innovation
- *21 October 2017: Sibiu, Romania*
  Creating Values – Impact & Exploitation in H2020 projects, at the International conference on Engineering

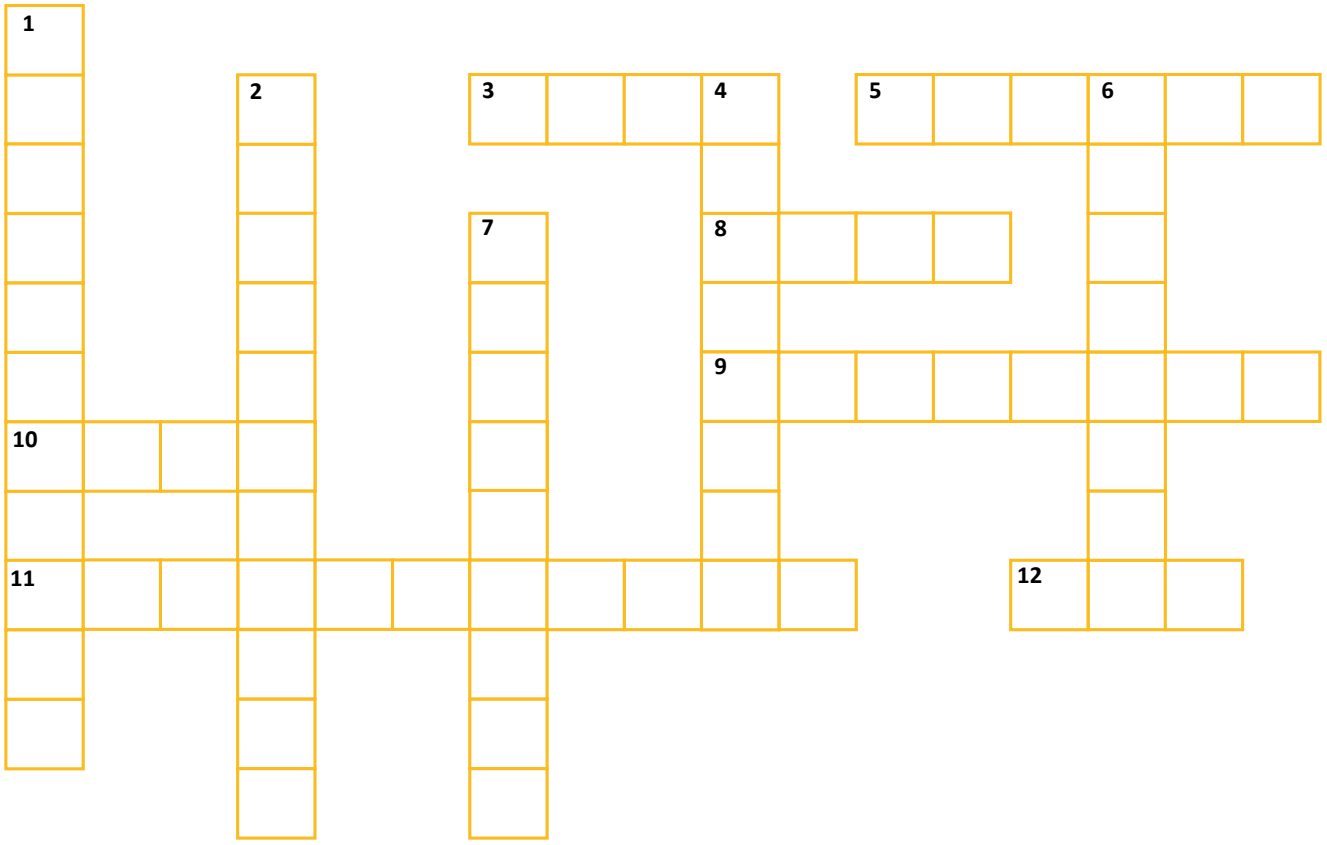
Upcoming webinars

- *14 September 2017:* Introduction to IP
- *27 September 2017:* Technology Transfer
- *10 October 2017:* How to properly manage IP and turn research results into standardisation
- *18 October 2017:* IP Commercialisation and Licensing

For further information, please have a look at our online event calendar.

SOFTWARE CROSSWORD

How about making a final recap of this Bulletin issue with a crossword puzzle? The answers are hidden in the articles!



Across

3. Many members of the software development community prefer not to adopt one or the other philosophy, this is the reason why the term ____ is widely used. *(abbreviation)*
5. The European Union ____ Licence is the first European F/OSS licence created on the initiative of the European Commission.
8. A program is ____ software if the program's users have the four essential freedoms.
9. "Non ____ clause" indicates that the software is provided "as is", and the developers have no liability with respect to the (bad) use of the software.
10. The distribution terms of ____ source software must comply with ten different criteria.
11. In order to be patentable, the software needs to be a computer ____ invention.
12. A movement that establishes other ways to deal with software, based on the user's liberty concept.

Down

1. In the "____ clause", the licensor asks the licensee to respect some conditions.
2. Is an essential requirement of copyright law.
4. A set of different objects that can include source code, compiled code or executables, interfaces, documentation and even preparatory design work.
6. Software is distributed under _____ .
7. The GNU _____ was written by Richard Stallman in 1985.

SOLUTION PREVIOUS COPYRIGHT QUIZ

1. The term of copyright protection is 70 years after the publication of a work. **FALSE**
2. Copyright registration is not required to obtain protection. **TRUE**
3. Shakespeare's "Romeo and Juliet" is in the public domain. So, can I put the image of the last edition's book cover published by the publishing company "Runo" on a T-shirt and sell it in my shop? **FALSE**
4. The decisions of the Court of Justice of the European Union (CJEU) have over the years clarified the meaning of what constitutes the author's own intellectual creation. **TRUE**
5. An artist who performs a musical work in the public domain owns no rights over his performance. **FALSE**
6. Moral rights are the only rights owned by authors. **FALSE**
7. National copyright laws have only been partially harmonised through EU Law. **TRUE**

Fancy a Little Quiz?

As you know in every issue we include a quiz to help you develop your patent searching skills using Espacenet. Why don't you try using Espacenet today?

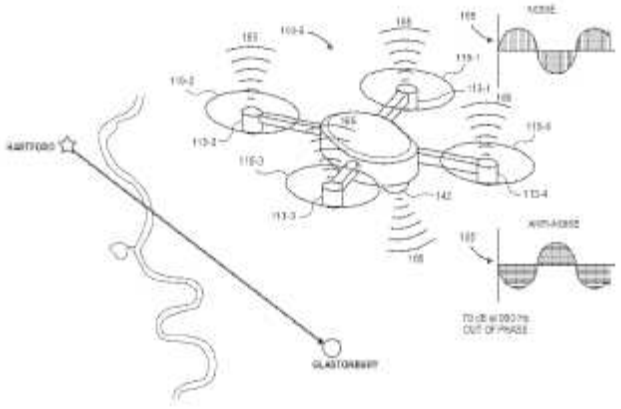
PATENT QUIZ

Silence the drones

Drones, like their name indicates, emit a low buzz that will be a problem when fleets of delivery drones start operating in urban airspace. Drones can also be used for filming, performing surveillance, reconnaissance, and exploration tasks for military and civilian applications. The noise that a drone emits can be quite annoying.

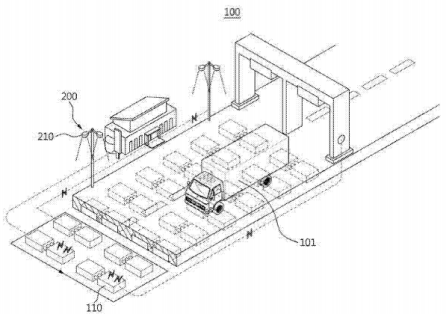
It could be interesting to develop noise cancelling systems to reduce the noise emitted by those unmanned aerial vehicles.

Using ESPACENET try finding patents covering such devices.



SOLUTION PREVIOUS PATENT QUIZ

Harvesting Energy from roads



Energy can be harvested on roads and highways using a piezoelectric generator.

Piezoelectric generators are embedded in a road or airport pavement and produce electrical power when a vehicle crosses their location.

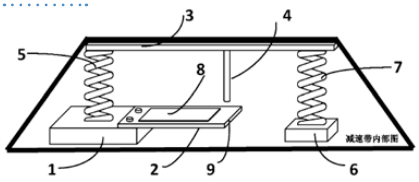
Using ESPACENET try finding patents covering such devices.

Step one: To find similar patents, identify the most pertinent aspects of the invention – common technical features that may be found in related patents – and for each aspect, define a comprehensive set of synonyms. To perform the search, the following concepts – groups of synonyms covering the different aspects of the invention – can be defined:

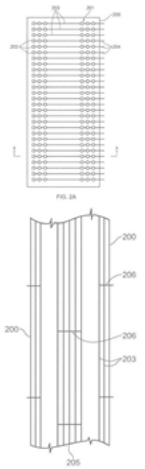
- piezo*
- road
- gener*
- Energy, Electricity

Several combinations can be tried. The combination **piezo* road gener*** yields the following list of documents out of which you can find:

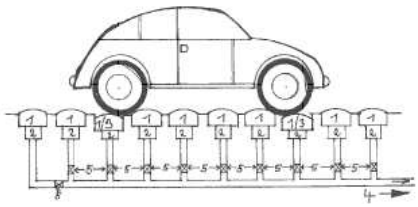
CN106100441 (A) - Piezoelectric energy collector applied to road deceleration strip and collection circuit of piezoelectric energy collector



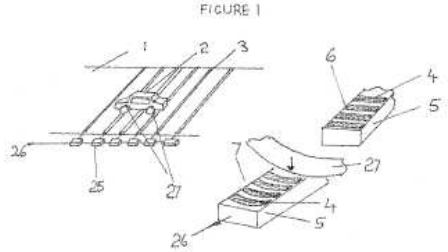
GB2498215 (A) - Piezoelectric Energy Harvesting Layer for Pavement



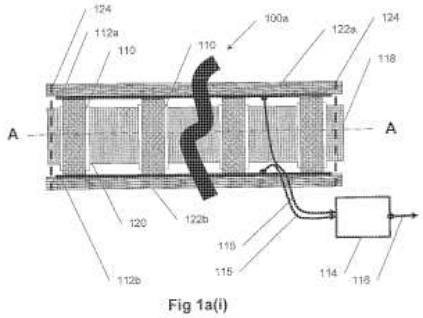
DE102011116180 (A1) - Piezoelectric power plant for use under e.g. road surfaces, has piezoelectric materials arranged under road surfaces, railway tracks, or pedestrians, such that short-term pressure is exerted on piezoelectric materials



GB2484953 (A) - Electricity generation methods utilising moving vehicles



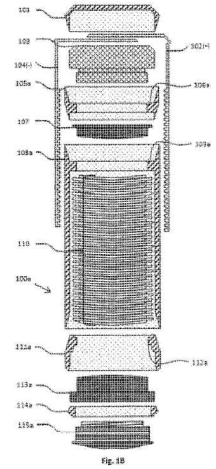
WO2010116348 (A1) - MODULAR PIEZOELECTRIC GENERATORS



The search can be performed by combining similar concepts using classification symbols. There is one broadly covering the concepts of a piezoelectric generator: H02N18, another one those of roads: E01.

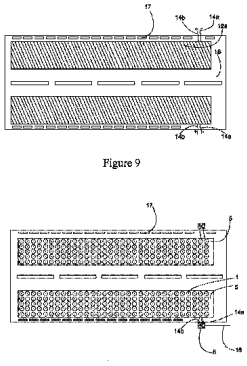
Step two: Combining H02N18 with road results in this list, where you will find some additional documents like:

WO2015142152 (A1) - APPARATUS FOR GENERATING ELECTRIC ENERGY FROM THE MECHANICAL COMPRESSION OF PIEZOELECTRIC TRANSDUCERS



Combining E01 with piezoelectric you will obtain this list, out of which you will find:

US2014300250 (A1) - SYSTEM FOR GENERATING AND DISTRIBUTING ENERGY FROM PIEZOELECTRIC MATERIALS



This search can be completed by multiplying the combinations of keywords and classification. It very clearly demonstrates that this field is heavily patented and even if not many of those products have been put on the market, the industrial interest in this type of energy generator is considerable. China is very present in this field.

GLOSSARY

Software or computer programmes are a set of different objects that can include source code, compiled code or executables, interfaces, documentation and even preparatory design work. Ideas and algorithms, as well as interfaces, are not protected.

Free/Open Source Software (F/OSS) is computer software that can be classified as both free and open-source software. Most of the licences used in this kind of software make it both "free" and "open source" so there is no real legal distinction between these two concepts, despite the philosophical aspects. Such licences must comply with four specific unrestricted criteria on the use to be made of the software -freedom to run, copy, distribute, study, change and improve the program-, which have to be granted by the licensor to any licensee.

Free Software Foundation (FSF) is a non-profit movement to promote computer user freedom, which believes that any user should have the right to study the source code, modify it, and share the program.

European Union Public Licence (EUPL) is a licence, meaning a contract between a licensor (the author of the software) and a licensee (the user of the software, who can then use it according to the licence terms). Such licence is compulsory to authorise the widest possible use of the software: communication, copy, change or distribution, in full respect of the applicable law. The EUPL, which is the first European Free/Open Source (F/OSS) licence created on the initiative of the European Commission, is "Open Source" ensuring freedoms to use, analyse, adapt and redistribute the software. It is a unique legal instrument developed in 22 European languages and can be used by anyone for software distribution.

GET IN TOUCH

European IPR Helpdesk
c/o infeurope S.A.
62, rue Charles Martel
L-2134, Luxembourg

Phone: +352 25 22 33 - 333 (Helpline)
Fax: +352 25 22 33 - 334 (Helpline)
Email: service@iprhelpdesk.eu
www.iprhelpdesk.eu

Or find us on Twitter or LinkedIn:
https://twitter.com/IPRHelpdesk
http://www.linkedin.com/in/european-ipr-helpdesk

DISCLAIMER / LEGAL NOTICE

The European IPR Helpdesk project receives funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No 641474. It is managed by the European Commission's Executive Agency for Small and Medium-sized Enterprises (EASME), with policy guidance provided by the European Commission's Internal Market, Industry, Entrepreneurship and SMEs Directorate-General. Even though this Bulletin has been developed with the financial support of the EU, its content is not and shall not be considered as the official position of the EASME or the European Commission.

EDITORIAL BOARD

GUEST AUTHORS

Stefano Gentile
European Commission's Central IP Service

Teresa Gomez-Diaz
CNRS/LIGM

Sigmar Lampe
Université du Luxembourg, Luxembourg

SUBSCRIPTION

The Bulletin is published three-monthly by the European IPR Helpdesk and it is distributed free of charge.

All issues of the Bulletin are available at www.iprhelpdesk.eu/library/bulletins.