

Copyright in Multiscale Cancer Modelling

I. V. Lishchuk, M. S. Stauch

Institut für Rechtsinformatik
 Leibniz Universität Hannover
 Hannover, Germany
 e-mail: lishchuk@iri.uni-hannover.de

Abstract — *In silico* hyper-modelling is a complex process which requires interdisciplinary effort. Hyper-modelling is a challenge for scientists, and determining applicable legal protection is a challenge for lawyers. Insofar as a computer model is defined as a computer program, software copyright comes into play. At this stage several questions arise: What elements in computer modelling are copyrightable? Is the modelling work protected? What about copyright in a hyper-model design? In this paper, we seek to suggest answers to these questions. In particular, we investigate the scope of copyright, the requirements for protection, and what elements may be subject to copyright in the context of cancer modelling.

Keywords- *in silico* oncology; cancer modelling; computer models; software copyright; copyright in compilations.

I. INTRODUCTION

In silico oncology aims to improve cancer knowledge and treatment by creating reliable computer predictions. Simulation of cancer progression in space and time requires the use of multiscale cancer modelling. “A model is considered to be “multiscale” if it spans two or more different spatial scales and/or includes processes that occur at two or more temporal scales.” [1]. Multiscale is realized in silico by constructing elementary models – the ones which correspond to elementary biological processes, and relation models – the ones which reflect relations across them, into the hyper-models [2].

In the medical research domain the term hyper-model first appeared in 2008 in relation to Virtual Physiological Human [3]; and in 2011 in the context of computer science, where it was defined as “a concrete instance of an integrative model, built as the orchestration of multiple computer models that might run on different computers at different locations using different simulation stacks.” [4]. The first implementation, based on web services, was tested on biochemical models [5].

The research on multiscale cancer modeling is ongoing. In the project CHIC [6], single-scale models (from molecular to compartment models) created by different research groups are linked into integrated multiscale hyper-models. Linking and inter-play between the models is shown in Figure 1[7].

The technical research on cancer modelling is motivated by the perspective of using multiscale cancer models as a clinical tool [7]. Meanwhile, the legal research in the project seeks to investigate solutions for protecting cancer models and hyper-models in the European legal framework. The goals of legal research is to determine the type of protection

applicable to cancer models, to identify protectable elements and specify what requirements such elements must fulfil to be protected, and to filter out elements which are not protectable.

The results which we have reached so far are presented in this paper. Section II looks at the substance of modelling, and identifies protectable elements in terms of copyright law. The applicability of copyright to cancer models, modelling work and hyper-models is then assessed in Section III, IV and V, respectively. The limits of copyright and elements which are not subject to copyright are presented in Section VI, and the key conclusions of the paper summarized in Section VII.

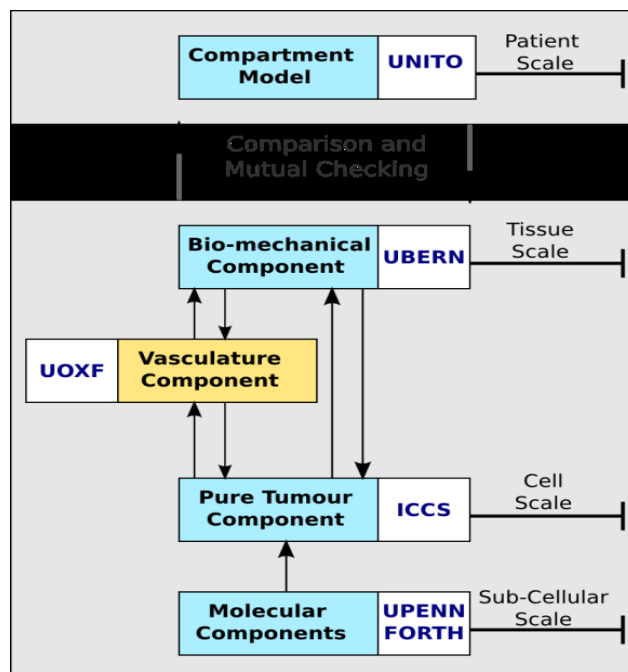


Figure 1. Schematic of the planned modelling framework for the CHIC project with the angiogenesis/vascular component highlighted.

II. CANCER MODELS IN TERMS OF COPYRIGHT

We may begin by distinguishing two types of models: scientific models and computer models. Scientific models are defined as: “finalized cognitive constructs of finite complexity that idealize an infinitely complex portion of reality through idealizations that contribute to the achievement of knowledge on that portion of reality that is objective, shareable, reliable and verifiable.” [8]. These

scientific models are implemented in silico via computer models. In the context of cancer modelling, a computer model is defined as: “*a computer program that implements a scientific model, so that when executed according to a given set of control instructions (control inputs) computes certain quantities (data outputs) on the basis of a set of initial quantities (data inputs), and a set of execution logs (control outputs).*” [9]. For implementation in silico, a scientific model is transformed into its executable form, i.e., encoded in a computer program.

When we consider that cancer models are implemented by computer models, which are written and executed by computer programs, protection of cancer models by copyright comes into question first. Computer programs are recognized as an object of copyright at the level of international law [10].

In addition, a cancer model itself emerges as the outcome of foregoing modelling work. The modelling materials which are recorded and lead to the reconstruction of a computer model for simulation may also constitute object of protection along with the model code [11].

Moreover, hyper-models, organized as choreographies, may also be an object of protection by copyright, provided the selection and arrangement of models in a hyper-model results from free intellectual creation of the modeler [12].

III. COPYRIGHT IN COMPUTER MODELS

To be protected by software copyright, a computer model must stand the criteria for protection. The requirements for copyright protection we consider below.

A. Protection of Software under Copyright

Copyright is a traditional type of protection, which both the European and the international law grant to software. Article 4 WIPO Copyright Treaty [13] and Article 10 TRIPS Agreement [10] protect computer programs as literary works within the meaning of Berne Convention (1886). The same principle is followed by European copyright law. Article 1 of Directive 2009/24/EC on the legal protection of computer programs (the “Software Directive”) recognizes computer programs as object of copyright protection in the EU [14].

B. Prerequisites for Copyright Protection

To be protected by the Software Directive, a cancer model must be “*original in the sense that it is the author’s own intellectual creation.*” No other criteria for protecting programs by copyright apply [15]. As interpreted by the CJEU, a work is original when the author in its own work expresses his creativity in an original manner [16].

Hence, to qualify as a copyright work, a model code must be developed (not copied) by the modeller in an original creative way. For this, the code must be written by a human being capable of intellectual activity. Code generated automatically will lack the level of creativity necessary for the copyright. Second, a model code must count as its author’s own intellectual creation. Third, the code must be expressed in a way which constitutes expression of a computer program for the purposes of software copyright.

How these criteria apply to the cancer models is described below.

C. Intellectual Creation in Computer Models

As stated, original creation is essential for copyright. However, if we consider that models are written in the era of powerful computing with the use of standard programming techniques and software development toolkits, like libraries, what counts as original creation in the model code might be questionable.

Model codes, as computer programs, are written in a programming language. The models, embodied in source code, are mostly written for interpreted languages, such as Python, Perl or MATLAB; models provided in object code are usually binary C or C++ compiled executables [9]. According to the CJEU SAS Institute decision, isolated symbols, figures, keywords, mathematical concepts, etc., which constitute the material of any programming language, do not constitute the intellectual creation of the author of a computer program and are not subject to copyright per se [17]. However, it is not these symbols that need to be protected. It is the model code which deserves protection most. The model code constitutes a script of symbols arranged into a specific sequence according to the syntax of the programming language so that a functional executable is achieved. And insofar as “*the choice, sequence and combination of those words...*” in the model code reflects original creation of the programmer such intellectual result is potentially subject to copyright protection [17].

At the same time, such protection does not mean that any copying of the code would automatically amount to copyright infringement. According to the CJEU approach, to prove copyright infringement the copying must, first, reflect intellectual expression of the author of the protected code and, second, be substantial [16]. “*Substantiality*” of the copying may be measured by different factors. Here, different jurisdictions operate varied criteria. Under the Australian case law, “*any part of a program is substantial if without it the program would either not work at all or not work as desired.*” [18]. In the US case law, substantiality is measured by the proportion of copying to the entire code. Copying 9 lines of 3,179 lines of code qualifies as “*... an innocent and inconsequential instance of copying in the context of a massive number of lines of code*” and does not count as copyright infringement [19]. Under the UK law substantiality is judged by the degree of skill and labor which the author deployed in the design and coding of the code alleged to be copied [18].

Therefore, before claiming copyright infringement substantiality of copying and intellectual creation reflected in the piece of code being copied first need to be evaluated. Another aspect which matters for the purposes of software copyright is the form of program expression.

D. Program Expression for the Purposes of Copyright

According to Article 10 TRIPS Agreement and case law of the CJEU, the program source code and object code constitute program expression for the purposes of software copyright. And, in line with the criteria established by the

CJEU, only such program expression which “*permits reproduction in different computer languages, such as the source code and the object code*” is subject to protection under Software Directive [20].

Hence, irrespective of whether the code is released as an executable, i.e. in the object code, or in its source code, the scope of copyright protection would be the same. By contrast, neither model codes generated by technical means, nor the models residing in the human mind or described in publications would count as program expression for the purposes of software copyright.

IV. COPYRIGHT IN PREPARATORY DESIGN MATERIAL

Copyright applicable to a model code may also extend to the preparatory design material, once such material stands the criteria for copyright.

A. Modelling Work as Preparatory Design Material

In addition to the computer model, the underlying modelling work also has the potential to be covered by software copyright. For this, the modelling work would need to qualify as preparatory design material (which is included into the object of protection by Article 1 of the Software Directive), and lead to the development of a computer model. The nature of modelling work must enable a computer model to be recreated from it at a later stage [11].

In assessing whether the modelling materials qualify as preparatory design work it is helpful to consider the modelling process by reference to the overall process of software development.

Thus, modelling of cancer models comprises various stages. These include identification of elementary processes for simulation (e.g., cell cycling, the angiogenesis process, declination of a cell to apoptosis after a particular treatment, etc.), the definition of modelling techniques - discrete, continuum, or hybrid [1], and development of computer codes for simulating those biological processes in silico.

These stages of modelling can be compared with the stages of software development. First, the problem to be solved by a computer is analyzed, then methods of solving the problem are adopted and stages of running the program are identified. Subsequently, detailed instructions for a computer to perform operations necessary for the execution of those stages are developed [21].

With this comparison in mind, insofar as the modelling materials lead to the recreation of a computer model they will have a good chance of qualifying as preparatory design material, and be subject to copyright in the model code.

In contrast to computer programs, there are no specific requirements to the form of expression of preparatory design work. However, the work must be expressed. The development documentation set down in writing, such as: data flow plans, designs of commands and information cycles, exhibits of scientific or technical art, expressed in any form, including mathematical, technical or graphic symbols, should suffice [22].

V. COPYRIGHT IN HYPER-MODELS

As noted in Section II, a further potential candidate for copyright protection in cancer modelling is the creative design by which simple models are integrated into a hyper-model. A hyper-model is defined as “*a model that emerges from the composition and orchestration of multiple hypomodels, each one of which is capable of simulating a specific entity or phenomenon. The hyper-model can simulate an entity or phenomenon that may be more complex than the ones simulated by each separate simpler model.*” [9].

In order to be protected, the overall design must embody intellectual creation. This is reflected in both Article 5 WIPO Copyright Treaty and Article 10 Paragraph 2 TRIPS Agreement, according to which (only): “*Compilations of data or other material, in any form, which by reason of the selection or arrangement of their contents constitute intellectual creations*” are subject to the law of copyright.

Although hyper-modelling is facilitated and semi-automated by technical infrastructure [23], nonetheless, an inter-play between the laws of biology and computer engineering make intellectual input in hyper-modelling indispensable. Accordingly, protection would be quite likely.

Copyright in the constituent models and copyright in the integrated hyper-model can subsist together. It allows that one and the same model may be re-assembled into different hyper-models. Any new hyper-model compiled in an original and creative way may be protected in its own right.

However, applicability of copyright to cancer models depends on multiple factors, and copyright also has its limitations. The limitations of copyright relevant to modelling are considered below.

VI. THE LIMITS OF COPYRIGHT PROTECTION

Copyright also has its limits and not all program elements are covered by copyright. The limits of copyright in relation to computer models we consider below.

A. Elements not Subject to Software Copyright

Above we established that copyright protects model codes. Nonetheless, not all aspects of the modelling will enjoy such protection. Most notably, the ideas and principles which lie behind the computer code of a cancer model are not subject to copyright. Consequently, whereas the model code as the end-product of modelling is copyrightable, the modeling background comprising techniques, algorithms, principles of computer science and bio-engineering on which a computer model rests will remain outside the boundaries of copyright. The non-copyrightability of such general modelling techniques means they can be used without restriction in future research for the benefit of the modelling community.

Non-copyrightability of ideas and principles is recognized by the law. Article 2 WIPO Copyright Treaty, Article 9 TRIPS Agreement provide: “*Copyright protection shall extend to expressions and not to ideas, procedures, methods of operation or mathematical concepts as such.*” Software Directive also explicitly says that ideas and

principles which underlie any element of a computer program, including its interfaces, are not protected by copyright [24]. Following this logic, also algorithms and programming languages which comprise those ideas are not copyrightable. Similarly, US copyright law also places “*ideas, plans, methods, systems, or devices, as distinguished from the particular manner in which they are expressed*” beyond the scope of copyright [25]. A code which is too abstract or straightforward and resembles an idea, rather than a functional executable, may lack the creativity to be protected by copyright in the US [26]. Whereas the computer code which implements some method or function may well be copyrightable, the method or functionality being implemented is not. In a case where the right holder tried to assert copyright in program functionality, the CJEU has adverted to the following argument to reject this: “... *to accept that the functionality of a computer program can be protected by copyright would amount to making it possible to monopolise ideas, to the detriment of technological progress and industrial development.*” [27].

On the other hand, some protection of innovative ideas, either processes or methods, may be obtainable under patent law. “*Unlike copyright, a valid patent does not protect the expression of an idea but the underlying substance of it.*” [28]. Further discussion of the scope of protection under patent law (which contains numerous complexities and uncertainties of its own) is outside the scope of the present paper.

B. Idea-Expression Dichotomy

Where, then, is the line between an idea and expression of an idea, or method of operation and implementation of the method by the code? What program elements are copyrightable expressions and what elements should remain free for the benefit of the public and for interoperability? The courts have sought to give answers to these questions by articulating various tests for segregating ideas and technical elements from copyrightable expression of a work.

Thus, the UK courts have regard to the expression - idea dichotomy and identify circumstances in which ideas “as such” are not subject to copyright [29]. For instance, if a work describes an idea, which has no connection with the literary or artistic nature of the work, such an idea is not covered by copyright in the work itself. For instance, the author of a paper describing a cancer model has a right to prevent copying of his paper, but he may not forbid others from implementing a model which he described. Secondly, commonplace ideas and principles which are used in a work, but do not constitute any substantial part of its nature are not-copyrightable [24]. In the context of computer modelling, such ideas may be common programming or modelling techniques. However, if we consider that such notions as “*judgment, skill and labour*” deployed by the author in a work matter for UK copyright, it may be argued that even if ideas as such are not copyrightable, the efforts a modeller deployed in expressing his ideas may be protected: “...*[UK law] cannot prevent the copying of a mere idea but can protect the copying of a detailed ‘idea’.* It is a question of degree where a good guide is the notion of over-

borrowing of the skill, labor and judgment which went into the copyright work.” [30].

A somewhat different test for distinguishing expression of an idea implemented by computer software from an idea per se has been adopted in the US and is known as the “*abstract-filtration-comparison*” test [31]. According to the test, the program is broken down into its structural components, the non-copyrightable elements are filtered out, and the copyrightable expression remains. For this purpose three elements precluded from copyright have been identified. First, copyright does not subsist in structures dictated by efficiency. To determine this, the court must inquire “*whether the use of this particular set of modules [is] necessary efficiently to implement that part of the program’s process being implemented.*” [31]. If the answer is positive, then the implementation of an idea by the programmer has merged with the underlying idea and is non-copyrightable. Second, copyright does not extend to elements dictated by external factors, such as standards, compatibility, program specifications, etc. Third, copyright in a program does not cover structures already found in the public domain, i.e., program elements which have been freely accessible.

The test may also be useful for filtering non-copyrightable elements in computer models. In this connection, an interesting question, at which we are looking at below, is how it may apply to an application programming interface (shortly known as “API”) – an element which constitutes part of a program, but is dictated by the requirements of interoperability.

C. Copyrightability of Program Interfaces

Interfaces are program elements by means of which a program communicates and interacts with the users and other software and hardware as it is intended to function [32]. For instance, a word processor lets a printer print a .doc file through its API.

Copyrightability of APIs has been tested under the US law. The case in question is C 10-03561 WHA, Oracle America, Inc. v. Google Inc. [19]. The case dealt with copyrightability and copyright infringement in interfaces of the programming language Java. The central question related to the “*extent to which if at all, certain replicated elements of the structure, sequence and organization of the Java application programming interface are protected by copyright*” [33].

Java is a powerful object oriented programming language, developed by Sun Microsystems, first released in 1996, and acquired by Oracle in 2010. Java has a number of pre-written programs, called “methods”, which invoke different functions, such as retrieving the cosine of an angle. These methods are grouped into “classes” and organised into “packages”. Software developers get access to those classes through the Java APIs [34]. In 2008 Java APIs had 166 “packages”, split into more than six hundred “classes”, all divided into six thousand “methods”.

Google built its Android platform on Java language and used 37 Java APIs which were the core for the smartphones. Google wrote its own implementations of the methods and classes which it needed. The only one substantial element

which Google copied from Java into Android was the names and headers of 37 API packages in question. Such copying of the headers amounted to replication of the structure, sequence and organisation of Java APIs. Oracle claimed copyright infringement, and Google defended with fair use arguing that Java is an open solution (as Oracle did not dispute) and there was no literal copying of the Java code.

Unlike code itself, the structure, sequence and organization of a program constitutes an element of non-literal expression of a program. As discussed in Section V, the protectability of such elements under the law of copyright will depend on whether the structure, sequence and organization of a program in question qualifies as an expression of an idea or an idea itself [35].

The District Court trying the case qualified the headers and method names in Java APIs as non-copyrightable. According to the interpretation criteria of the US Copyright Office, “*Even if a name, title, or short phrase is novel or distinctive or lends itself to a play on words, it cannot be protected by copyright.*” [25]. This lends support to non-protectability of isolated code items by copyright, as recognised by the CJEU [17].

As regards copying of the declarations and duplicating the command structure of Java APIs, the judge found that the command structure of Java APIs amounts to a method of operation – a material not subject to copyright in the US.

In Java programming, the specific declarations in the Java APIs designate a method. A method can be implemented in different ways, but is invoked by that specific declaration only. In Java, each symbol in a command structure is more than a simple name - each symbol carries a task to invoke a pre-assigned function:

java.package.Class.method()

Considering that for using Java class methods software developers need to replicate the Java declarations, the judge qualified the command structure of Java APIs as a method of operation – a functional element essential for interoperability not subject to the US Copyright Act. This position was based on the merger doctrine and non-copyrightability of structures dictated by efficiency: “... *When there is only one way to express an idea or function, then everyone is free to do so and no one can monopolize that expression.*” [36].

The decision of the District Court to keep Java APIs free of copyright was welcomed by the software industry as essential for interoperability. Freedom to implement or decompile program APIs would allow the developers to write compatible software and let people switch platforms and services without the need for a license [37].

However, on appeal, the Federal Circuit Court reversed that ruling. The court found the declaring code and the structure, sequence and organisation of packages in Java APIs to be protected by copyright, since Java programmers were not restrained by the factor of efficiency to arrange the declarations in a different way [38]. The implications of this appellate decision may be rather dangerous for the software industry, because copyright in APIs would deny the potential to extend program functionality by compatible apps without a license [37]. The case is now under

consideration by the US Supreme Court and the final ruling is expected soon.

As regards the copyrightability of APIs from the perspective of EU law, it is possible, following the approach of the CJEU in its BSA decision [20], to characterize API as an interactive interface which provides connectivity to a program, but which does not enable reproduction of a program in another computer language. Accordingly, a key element, which would count as a program expression in the meaning of Article 1 Software Directive, would be missing, and the API fall outside of protection. However, as long as the CJEU has not decided on copyrightability of APIs, it is a hypothetical approach only.

Copyrightability of program APIs is important for computer models as well. Such models, in common with any other computer programs, communicate with each other through APIs. In the context of CHIC, in order for the models in the project to be interoperable and communicate with each other as designed, the model interfaces are standardized. However, for expanding the cancer models beyond CHIC and for making the CHIC models interoperable with external models, the release of the models’ APIs will be required. The conditions for such release may well depend on their copyrightability and the intentions of the modelers (commercialization or release open source).

VII. CONCLUSIONS

From the above observations follows that copyright may reside in different elements of cancer models and hyper-models. First, the code of a computer cancer model may well qualify as a computer program and be protected as such. Second, copyright in the model code may extend to the modelling work, provided it is recorded in writing and leads to reproduction or subsequent recreation of the computer model. Third, hyper-models which by reason of the selection or arrangement of the models in them constitute intellectual creations may qualify as compilations and be protected as such.

At the same time, not all elements of the models and modelling efforts are protectable. Thus, copyright does not extend to ideas, principles, methods of operation, algorithms, mathematical formulae, etc. Such pieces of the modelling toolkit may not be monopolized. Further, the copyrightability of program and model interfaces remains questionable both in the EU and the US. However, in view of the open source commitment followed by the modelling community, copyrightability of interfaces plays a secondary role.

The research on cancer modelling is ongoing. Further legal research will focus on the semantic linking of in silico cancer models and applicability of copyright to hyper-models.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement No 600841.

REFERENCES

- [1] T. Deisboeck, Z. Wang, P. Macklin, V. Cristini, “Multiscale Cancer Modelling”, Annual Review of Biomedical Engineering, vol. 13, pp. 127-155, August 2011, doi: 10.1146/annurev-bioeng-071910-124729.
- [2] “Computational Horizons In Cancer (CHIC): Developing Meta- and Hyper-Multiscale Models and Repositories for In Silico Oncology”, CHIC, Project, Objectives, <http://chic-vph.eu/project/objectives/>2015.05.07.
- [3] M. Viceconti, “The skeleton of the Virtual Physiological Human”, presented at: “The Living Human Project: building the musculoskeletal physiome” mini-symposium, MBEC2008 Antwerpen, Belgium, 2008.
- [4] P. A. Fishwick, “Hyperm modelling: an intergrated approach to dynamic system modelling”, Journal of simulation, vol. 6, pp. 2-8, February 2012.
- [5] R. Guha, “Flexible Web Service Infrastructure,for the Development and Deployment of Predictive Models”, Journal of Chemical Information and Modelling, vol. 48 (2), pp. 456-464, 2008.
- [6] “Computational Horizons In Cancer (CHIC): Developing Meta- and Hyper-Multiscale Models and Repositories for In Silico Oncology”, CHIC, Project Summary, <http://chic-vph.eu/project/> 2015.05.07.
- [7] J. A. Grogan, P. K. Maini, J. Pitt-Francis and H. M. Byrne, “Simulating Tumour Vasculature at Multiple Scales”, Proc. 2014 6th Int. Adv. Res. Workshop on In Silico Oncology and Cancer Investigation (IARWISOCI), Athens, Greece, Nov.3-4, 2014 (www.6thiarwisoci.iccs.ntua.gr), pp. 23-26, (open-access version), ISBN: 978-618-80348-1-5.
- [8] M. Viceconti, “A tentative taxonomy for predictive models in relation to their falsifiability”, Philos. Transact A Math Phys Eng Sci, vol. 369, pp. 4149-61, 1954.
- [9] CHIC, Deliverable No. 7.1, “Hyperm modelling specifications”, CHIC, Downloads, Deliverables, Public Deliverables, http://chic-vph.eu/uploads/media/D7-1_Hyperm modelling_specifications.pdf [accessed: 2015-05-07].
- [10] Agreement on Trade-Related Aspects of Intellectual Property Rights, the TRIPS Agreement, Annex 1C of the Marrakesh Agreement Establishing the World Trade Organization, Marrakesh, Morocco, 15 April 1994.
- [11] Article 1 Paragraph 1, Recital 7, Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, OJEU, L 111/16, 5 May 2009.
- [12] Article 5, WIPO Copyright Treaty, Geneva, 20 December 1996; Article 10 Paragraph 2 TRIPS Agreement.
- [13] WIPO Copyright Treaty, Geneva, 20 December 1996.
- [14] Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, OJEU, L 111/16, 5 May 2009.
- [15] Article 1 Paragraph 3, Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, OJEU, L 111/16, 5 May 2009.
- [16] CJEU, Judgment of 16 July 2009, Case C 5/08, Infopaq International A/S v Danske Dagblades Forening, Recital 45.
- [17] CJEU, Judgment of 2 May 2012, Case C 406/10, SAS Institute Inc v World Programming Ltd.
- [18] S. Stokes, “Digital Copyright, Law and Practice”, 4. Edition, Hart Publishing, Oxford and Portland, Oregon, 2014, p. 121.
- [19] U.S. District Court for the Northern District of California, Ruling of 31 May 2012, Case C 10-03561 WHA, Oracle America, Inc., v. Google Inc., p. 14, lines 1- 4.
- [20] CJEU, Judgment of 22 December 2010, Case C 393/09, Bezpečnostní softwarová asociace – Svaz softwarové ochrany v Ministerstvo kultury, Recital 35.
- [21] Report by the International Bureau, “Measures to enhance International Cooperation in the field of Legal Protection of Computer Software”, WIPO expert group on the legal protection of computer software, First Session, Geneva, 27-30 November 1979, LPCS/I/2, 30 September 1979.
- [22] Federal Court of Justice of Germany, Judgment of 09 May 1985, Case I ZR 52/83, BGHZ 94, pp. 276 – 292.
- [23] D. Tartarini, K. Duan, N. Gruel, D. Testi, D. Walker, and M. Viceconti, “The VPH Hyperm modelling Framework for Cancer Multiscale Models in the Clinical Practice”, Proc. 2014 6th Int. Adv. Res. Workshop on In Silico Oncology and Cancer Investigation (IARWISOCI), Athens, Greece, Nov.3-4, 2014 (www.6thiarwisoci.iccs.ntua.gr), pp. 61-64, (open-access version), ISBN: 978-618-80348-1-5.
- [24] Article 1 Paragraph 2, Recital 11, Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, OJEU, L 111/16, 5 May 2009.
- [25] U.S. Copyright Office, Circular 34; “Copyright Protection ‘Not Available for Names, Titles or Short Phrases”, rev. January 2012.
- [26] R. T. Nimmer, “Legal Issues in Open Source and Free Software Distribution”, adapted from Chapter 11 in Raymond T. Nimmer, The Law of Computer Technology, 1997, 2005 Supp.
- [27] CJEU, Judgment of 02 May 2012, Case C 406/10, SAS Institute Inc v World Programming Ltd., Recital 40.
- [28] A. M. St. Laurent, “Understanding open source & free software licensing”, O’Reilly, 1. Edition, 2004, p. 2.
- [29] S. Stokes, “Digital Copyright, Law and Practice”, 4. Edition, Hart Publishing, Oxford and Portland, Oregon, 2014, p.117.
- [30] Lord Bingham of Cornill, Opinions of the Lords of Appeal for Judgment in the case Designer Guild Ltd. v. Russel Williams (Textiles) Ltd. (Trading as Washington DC), 23 Nov. 2000, 1 WLR 2416,2422D-2423E.
- [31] U.S. Court of Appeals for the Second Circuit, Computer Associates International, Inc. v. Altai, Inc., 982 F.2d 693, 2d Cir.1992.
- [32] Recital 10, Directive 2009/24/EC of the European Parliament and of the Council of 23 April 2009 on the legal protection of computer programs, Official Journal of the European Union, L 111/16, 5 May 2009.
- [33] U.S. District Court for the Northern District of California, Ruling of 31 May 2012, Case C 10-03561 WHA, p. 1, lines 17-22.
- [34] Oracle, Java Platform, Standard Edition 6, API Specification, Overview, <http://docs.oracle.com/javase/6/docs/api/> 2015.05.08.
- [35] U.S. District Court for the Northern District of California, Ruling of 31 May 2012, Case C 10-03561 WHA, p. 27, lines 4-8.
- [36] U.S. District Court for the Northern District of California, Ruling of 31 May 2012, Case C 10-03561 WHA, p. 3, lines 20-21.
- [37] C. McSherry, “Dangerous Decision in Oracle v. Google: Federal Circuit Reverses Sensible Lower Court Ruling on APIs”, Electronic Frontier Foundation, Deeplinks Blog, <https://www.eff.org/deeplinks/2014/05/dangerous-ruling-oracle-v-google-federal-circuit-reverses-sensible-lower-court> 2015.02.18.
- [38] U.S. Court of Appeals for the Federal Circuit, Ruling of 09 May 2014, Appeals from the United States District Court for the Northern District of California in No. 10-CV-3561.