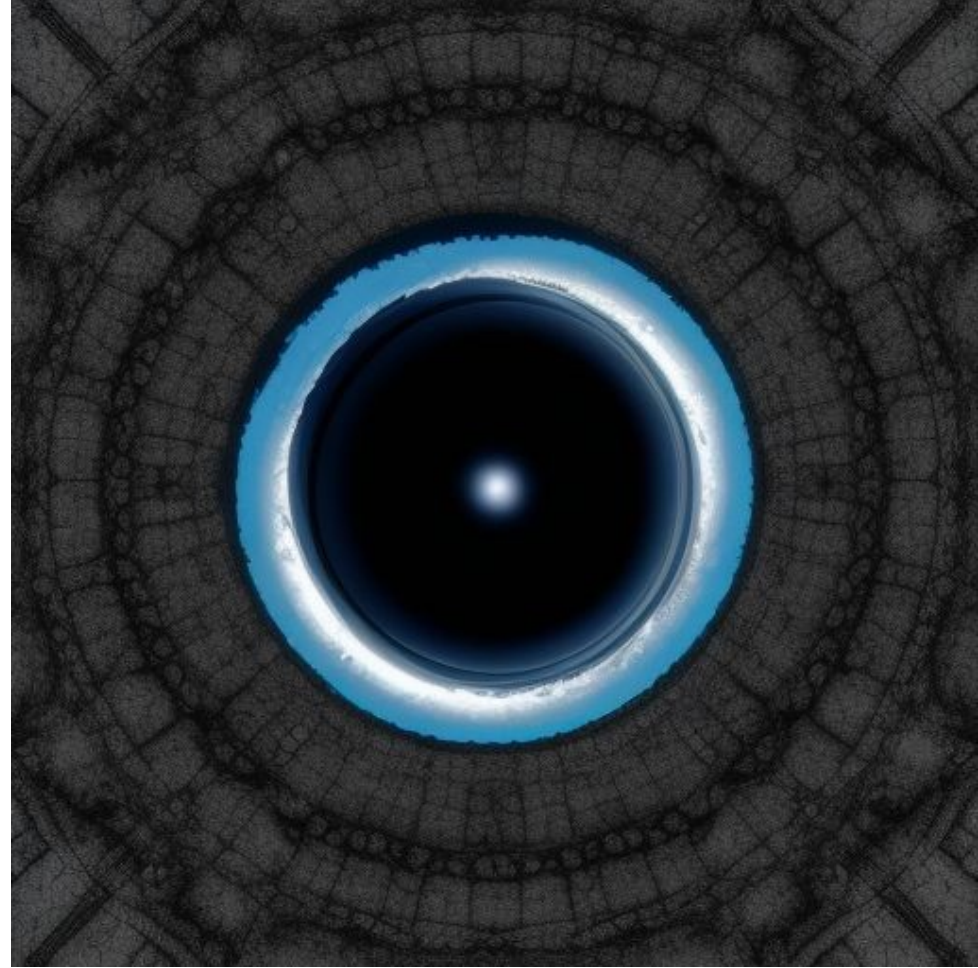


MUNI LAW

[GAZING INTO AN ABYSS: DEFINING THE NON-PROTECTED ABSTRACT MATTERS IN EU COPYRIGHT LAW]



Matěj Myška

Institute of Law and Technology, Faculty of Law, Masaryk University

www.cyber.law.muni.cz

matej.myska@law.muni.cz

@matejmyska

Acknowledgment / Affiliation

- This presentation was supported by the Czech Science Foundation (Grantová agentura České republiky) – Project **Díla chráněná autorským právem a požadavek dostatečné přesnosti a objektivity (22-22517S)**

Outline

1. Problem statement and course of the examination
2. Sufficient precision and objectivity as the new protectability requirement
3. Defining the non-protected abstract matters in EU copyright law with focus on computer programs
4. Excluded abstract matters in patent law
5. Conclusions

1. Problem statement and course of the examination

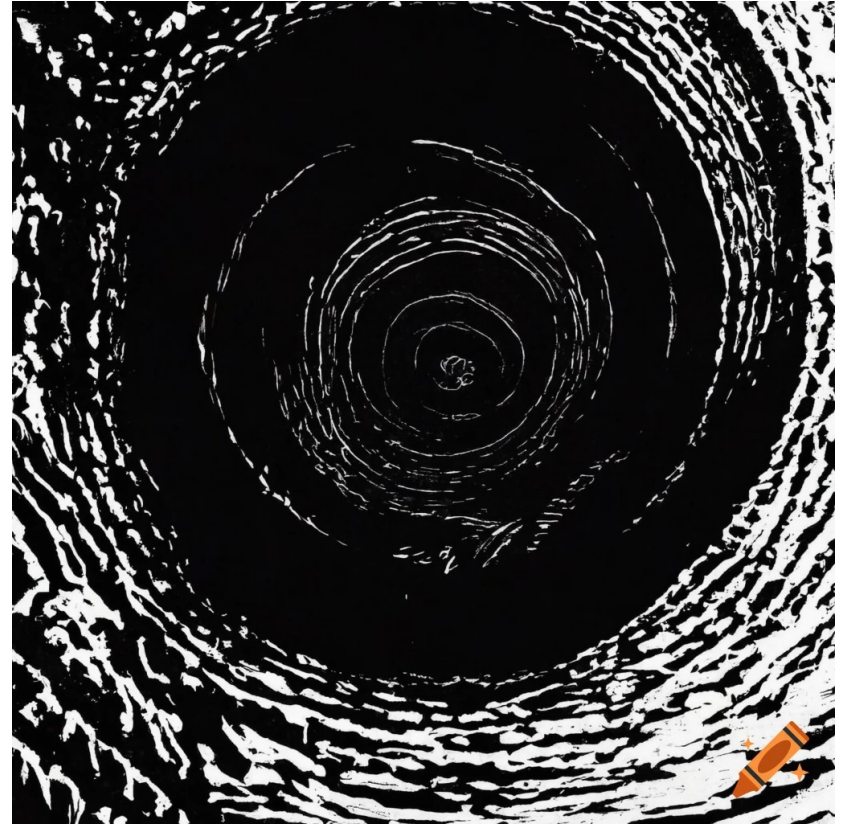
The [happy place] of EU copyright law

- *“Ohne Werk kein Urheber und kein Urheberrecht”*
- Autonomous term
- fully interpreted by the (CJEU)
 - Levola Hengelo, Cofemel and Brompton Bicycle
- *“original, in the sense that it is the author’s own intellectual creation”*
- OR MAYBE NOT



Searching for limits and outskirts of protection

- Focus: computer programs
- Sufficient precision and objectivity
- “[void]”/“abstractness”
- Patent law
- Widła [2023, 16] *“any attempt to pinpoint the exact boundary between protected and unprotected elements is perilous”*



2. Sufficient precision and objectivity as the new protectability requirement

Why?

- Serves to identify, clearly and precisely, the subject matter”
- *“bounded expressive objects that have a certain unity and stability of expressive form”* [Pila, 2021, 67]
- „Attributable connection“ between creative process and expression
- The practical part > needs to be proved in infringement proceeding [Peukert, 2023, 65]



For computer programs?

- ❑ Stable and expressive form
- ❑ BUT graphic user interface; programming languages, data formats, functionality >>> do not constitute an “expression”
- ❑ Laskowska-Litak [2019, 767] logical counter-conclusion:
 - ❑ only such an expression of the computer program is to be protected when its *“reproduction would engender the reproduction of the computer program itself, thus enabling the computer to perform its task”* (C-393/09, para. 38)
 - ❑ **only the [“functional expressions”], i.e., the one realizing the tasks of the computer program, are generally to be regarded as an expression of a computer program**



3. Defining the non-protected abstract matters in EU copyright law with focus on computer programs

Non-protected matters

- “the more abstract the to-be-protected matter, the more likely it is to be regarded as not eligible for copyright protection” [Peukert, 2023, 66]
- “only expressions” Peukert [2023, 65] > (“konkret-persönliche Form”) for that matter, and not “ideas, procedures, methods of operation or mathematical concepts as such” A9P2 TRIPS (also WCT, recital 11 CPD *“logic, algorithms and programming languages comprise ideas and principles, those ideas and principles are not protected”*)
- THE LEVEL OF ABSTRACTION – MOST IMPORTANT – BUT FLUID [Grützmacher, 2022, § 69a, marg. n. 28]

Non-protected matters

- Copyright law protects: only the concretely expressed original idea (“Ausdrucksform”; “konkret ausgedrückte Ideen”) and not the concretized mental concepts (“concretisierte gedankliche Konzepte”) and not at all abstract ideas [Wiebe et al., 2022, 204]
- “not the abstract functionality (idea, working method), i.e., not the technical problem, that the software is solving” [Peukert, 2023, 77; similarly, Wiebe, 2019, marg. n. 21].
- Paradoxically > the solution of the informational problem is to be regarded as the “most important contribution” : Blocher/Walter [2010, 104]

Example: Algorithm

- basically a description of a process/solving of a problem [Blocher/ Walter, 2010, 103; Janssens, 2021, 80]
- >>> description on a high level of abstraction [Blocher/Walter, 2010, 103] excluded from copyright protection to the level of comprising ideas and principles
- however a specific expression > a “structured solution” to a problem might be protected [Blocher/Walter, 2010, 104] or *“the way in which the algorithms are implemented and assigned to each other”* [Grü tzmacher, 2022, marg. n. 29]

Trying to find out

- Abstraction-Filtration-Comparison test \Leftrightarrow „tissue theory” (“Gewebetheorie”) I ZR 139/89
- Wiebe [2019, marg. n. 22]:
- *„different levels of abstraction are differentiated, from coding to the task of the programme as a whole and determination of a level of the idea.*
- *the expression is determined by the idea, i.e. whether the programmer had any freedom of design*
- *restrictive factors such as efficiency and functional constraints, standardisation, compatibility and the general spread of programming techniques are also taken into account*
- *...the infringement examination, the remaining elements are compared with the infringing programme with regard to essential similarities”*

4. Excluded abstract matters in patent law

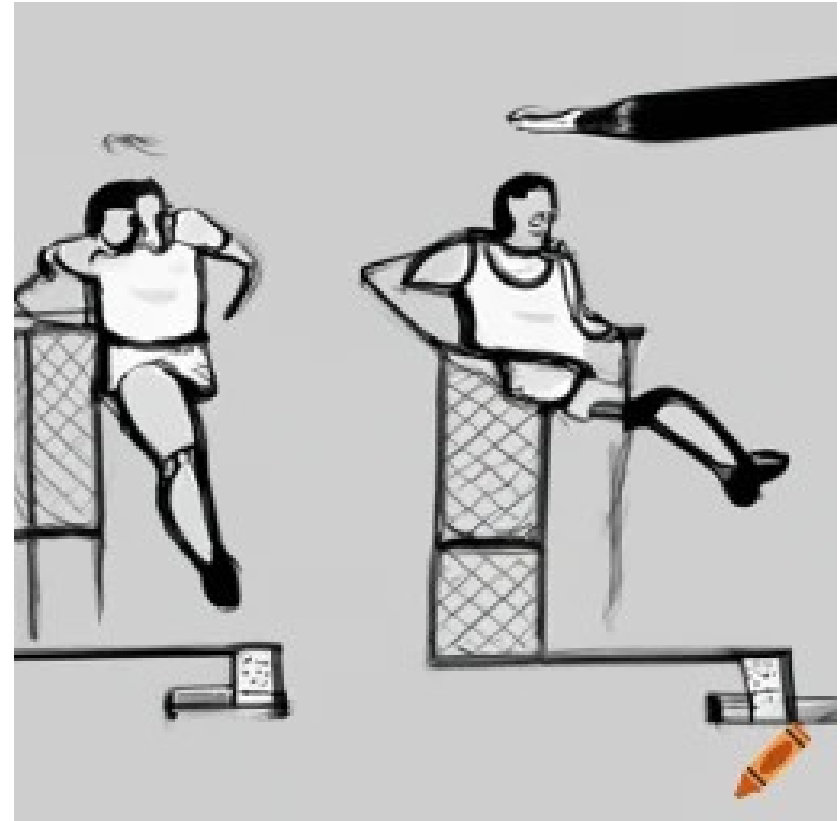
Rejecting patent law

- Tendency not to protect „abstract matters“
- Patentability requirements (e.g. A52 EPC): novelty, inventive step, industrial application AND invention > [technical solution] to technical means
- EXCLUDED:
 - (a) discoveries, scientific theories and mathematical methods;
 - (b) aesthetic creations;
 - (c) schemes, rules and methods for performing mental acts, playing games or doing business, and programs for computers;
 - (d) presentations of information
- = Abstract and intellectual matters as such



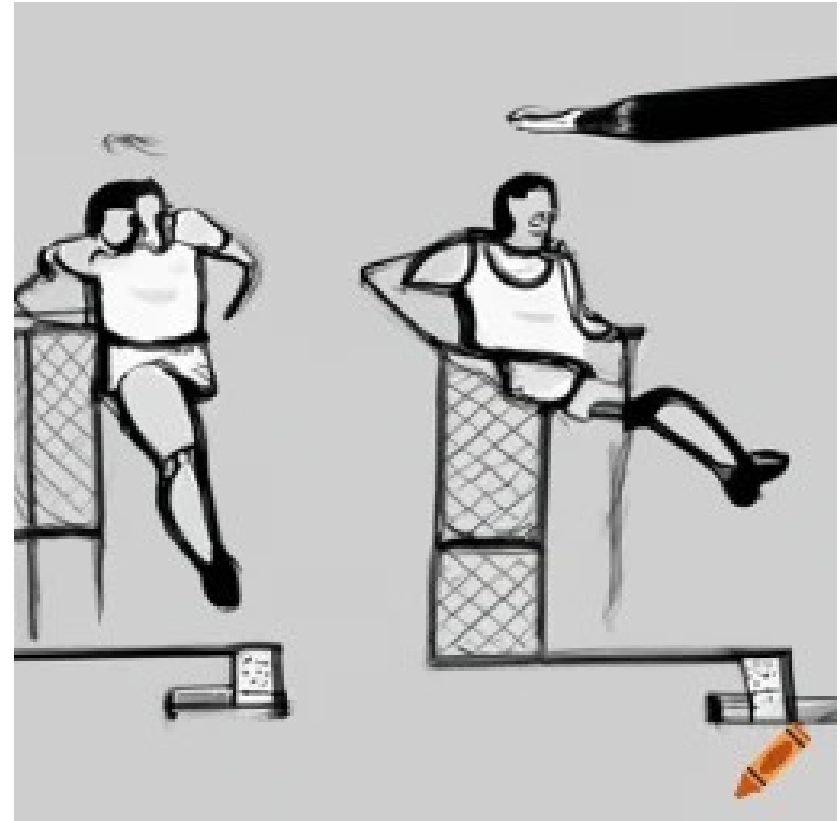
Accepting patent law

- Abstract matters/activities > still part of a patentable subject matter if the basis invention has technical character
- Cornerstone of debates of patentability of computer-implemented inventions
- [two-hurdle] approach
- Invention + patentability > inventive step



Accepting patent law

- ONLY features making technical contribution are assessed
- Algorithms not technical, but might be IF interacting with other parts of the claimed inventiond (Steinbrener/ Chandler et al., 2019)
- Technical effect T 1173/97-3.5.01



Merging the concepts

- Functional [“form of expression”] for computer programs protected (Laskowska-Litak [2019, 767])
- Purely abstract or theoretical concepts not passing the first hurdle >>> automatically excluded from copyrightability > lacking in the functional expression, despite the fact that the functionality as such is not protected by copyright law



[Unmergable concepts]

- Passing of the second hurdle, i.e., the technical contribution/technical effect
- does not help much with identifying the “void spaces” >
- basic teleological differences between the two protection regimes
- the difference between the protection of the (technical) function and its expression



5. Conclusions

Conclusions

- „Subjective precision and objectivity“ criterion is not to be regarded as fundamental problem (legal transplant)
- „Abstract matters“ delimitation is
- Patent abstract matters are also copyright abstract matters
- *Sony Computer Entertainment Europe (Datel case)* > immutability of variables content that are being used by the underlying program in the working memory but changed by another independent program = infringement?
- *Mio and Others* originality requirement

MUNI LAW

**Thank you
for your
attention
and your
questions!**

matej.myska@law.muni.cz
[@matejmyska](#)



Stable Diffusion prompt: sleepy academic

Illustrations / References

- Created by Craiyon <https://www.craiyon.com/>
- Prompt in [brackets]
- Please refer to the published version of the contribution